STATA META-ANALYSIS REFERENCE MANUAL

RELEASE 18



A Stata Press Publication
StataCorp LLC
College Station, Texas



Copyright © 1985–2023 StataCorp LLC
All rights reserved
Version 18

Published by Stata Press, 4905 Lakeway Drive, College Station, Texas 77845

ISBN-10: 1-59718-390-3
ISBN-13: 978-1-59718-390-1

This manual is protected by copyright. All rights are reserved. No part of this manual may be reproduced, stored in a retrieval system, or transcribed, in any form or by any means—electronic, mechanical, photocopy, recording, or otherwise—without the prior written permission of StataCorp LLC unless permitted subject to the terms and conditions of a license granted to you by StataCorp LLC to use the software and documentation. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document.

StataCorp provides this manual "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. StataCorp may make improvements and/or changes in the product(s) and the program(s) described in this manual at any time and without notice.

The software described in this manual is furnished under a license agreement or nondisclosure agreement. The software may be copied only in accordance with the terms of the agreement. It is against the law to copy the software onto DVD, CD, disk, diskette, tape, or any other medium for any purpose other than backup or archival purposes.

The automobile dataset appearing on the accompanying media is Copyright © 1979 by Consumers Union of U.S., Inc., Yonkers, NY 10703-1057 and is reproduced by permission from CONSUMER REPORTS, April 1979.

Stata, STATA, Stata Press, Mata, MATA, and NetCourse are registered trademarks of StataCorp LLC.

Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations.

StataNow and NetCourseNow are trademarks of StataCorp LLC.

Other brand and product names are registered trademarks or trademarks of their respective companies.

For copyright information about the software, type `help copyright` within Stata.

The suggested citation for this software is

> StataCorp. 2023. *Stata 18*. Statistical software. StataCorp LLC.

The suggested citation for this manual is

> StataCorp. 2023. *Stata 18 Meta-Analysis Reference Manual* . College Station, TX: Stata Press.

Contents

[+] This manual includes features that are part of StataNow.

Intro . . . . . Introduction to meta-analysis 1
meta . . . . . Introduction to meta 19

meta data . . . . . Declare meta-analysis data 57
meta esize . . . . . Compute effect sizes and declare meta-analysis data[+] 81
meta set . . . . . Declare meta-analysis data using generic effect sizes 114
meta update . . . . . Update, describe, and clear meta-analysis settings 129

meta forestplot . . . . . Forest plots[+] 133
meta summarize . . . . . Summarize meta-analysis data[+] 172

meta galbraithplot . . . . . Galbraith plots 213
meta labbeplot . . . . . L'Abbé plots 221
meta regress . . . . . Meta-analysis regression 230
meta regress postestimation . . . . . Postestimation tools for meta regress 250
estat bubbleplot . . . . . Bubble plots after meta regress 260

meta funnelplot . . . . . Funnel plots 269
meta bias . . . . . Tests for small-study effects in meta-analysis 287
meta trimfill . . . . . Nonparametric trim-and-fill analysis of publication bias 302

meta meregress . . . . . Multilevel mixed-effects meta-regression 315
meta multilevel . . . . . Multilevel random-intercepts meta-regression 345
meta me postestimation . . . . . Postestimation tools for multilevel mixed-effects meta-analysis 356

meta mvregress . . . . . Multivariate meta-regression 364
meta mvregress postestimation . . . . . Postestimation tools for meta mvregress 401

estat group . . . . . Summarize the composition of the nested groups 412
estat heterogeneity (me) . . . . . Compute multilevel heterogeneity statistics 413
estat heterogeneity (mv) . . . . . Compute multivariate heterogeneity statistics 416
estat recovariance . . . . . Display estimated random-effects covariance matrices 422
estat sd . . . . . Display variance components as standard deviations and correlations 424

Glossary 426

Subject and author index 435

Cross-referencing the documentation

When reading this manual, you will find references to other Stata manuals, for example, **[U] 27 Overview of Stata estimation commands**; [R] **regress**; and [D] **reshape**. The first example is a reference to chapter 27, *Overview of Stata estimation commands*, in the *User's Guide*; the second is a reference to the `regress` entry in the *Base Reference Manual*; and the third is a reference to the `reshape` entry in the *Data Management Reference Manual*.

All the manuals in the Stata Documentation have a shorthand notation:

[GSM]	*Getting Started with Stata for Mac*
[GSU]	*Getting Started with Stata for Unix*
[GSW]	*Getting Started with Stata for Windows*
[U]	*Stata User's Guide*
[R]	*Stata Base Reference Manual*
[ADAPT]	*Stata Adaptive Designs: Group Sequential Trials Reference Manual*
[BAYES]	*Stata Bayesian Analysis Reference Manual*
[BMA]	*Stata Bayesian Model Averaging Reference Manual*
[CAUSAL]	*Stata Causal Inference and Treatment-Effects Estimation Reference Manual*
[CM]	*Stata Choice Models Reference Manual*
[D]	*Stata Data Management Reference Manual*
[DSGE]	*Stata Dynamic Stochastic General Equilibrium Models Reference Manual*
[ERM]	*Stata Extended Regression Models Reference Manual*
[FMM]	*Stata Finite Mixture Models Reference Manual*
[FN]	*Stata Functions Reference Manual*
[G]	*Stata Graphics Reference Manual*
[H2OML]	*Machine Learning in Stata Using H2O: Ensemble Decision Trees Reference Manual*
[IRT]	*Stata Item Response Theory Reference Manual*
[LASSO]	*Stata Lasso Reference Manual*
[XT]	*Stata Longitudinal-Data/Panel-Data Reference Manual*
[META]	*Stata Meta-Analysis Reference Manual*
[ME]	*Stata Multilevel Mixed-Effects Reference Manual*
[MI]	*Stata Multiple-Imputation Reference Manual*
[MV]	*Stata Multivariate Statistics Reference Manual*
[PSS]	*Stata Power, Precision, and Sample-Size Reference Manual*
[P]	*Stata Programming Reference Manual*
[RPT]	*Stata Reporting Reference Manual*
[SP]	*Stata Spatial Autoregressive Models Reference Manual*
[SEM]	*Stata Structural Equation Modeling Reference Manual*
[SVY]	*Stata Survey Data Reference Manual*
[ST]	*Stata Survival Analysis Reference Manual*
[TABLES]	*Stata Customizable Tables and Collected Results Reference Manual*
[TS]	*Stata Time-Series Reference Manual*
[I]	*Stata Index*
[M]	*Mata Reference Manual*

Description Remarks and examples References Also see

Description

Meta-analysis (Glass 1976) is a statistical technique for combining the results from several similar studies. The results of multiple studies that answer similar research questions are often available in the literature. It is natural to want to compare their results and, if sensible, provide one unified conclusion. This is precisely the goal of the meta-analysis, which provides a single estimate of the effect of interest computed as the weighted average of the study-specific effect estimates. When these estimates vary substantially between the studies, meta-analysis may be used to investigate various causes for this variation.

Another important focus of the meta-analysis may be the exploration and impact of small-study effects, which occur when the results of smaller studies differ systematically from the results of larger studies. One of the common reasons for the presence of small-study effects is publication bias, which arises when the results of published studies differ systematically from all the relevant research results.

Comprehensive overview of meta-analysis may be found in Sutton and Higgins (2008); Cooper, Hedges, and Valentine (2019); Borenstein et al. (2009); Higgins and Green (2017); Hedges and Olkin (1985); Sutton et al. (2000a); and Palmer and Sterne (2016). A book dedicated to addressing publication bias was written by Rothstein, Sutton, and Borenstein (2005).

This entry presents a general introduction to meta-analysis and describes relevant statistical terminology used throughout the manual. For how to perform meta-analysis in Stata, see [META] **meta**.

Remarks and examples

Remarks are presented under the following headings:

Brief overview of meta-analysis
Meta-analysis models
  Common-effect ("fixed-effect") model
  Fixed-effects model
  Random-effects model
  Comparison between the models and interpretation of their results
  Meta-analysis estimation methods
Forest plots
Heterogeneity
  Assessing heterogeneity
  Addressing heterogeneity
  Subgroup meta-analysis
  Meta-regression
Publication bias
  Funnel plots
  Tests for funnel-plot asymmetry
  The trim-and-fill method
Cumulative meta-analysis
Leave-one-out meta-analysis
Multivariate meta-regression
Multilevel meta-regression

Brief overview of meta-analysis

The term meta-analysis refers to the analysis of the data obtained from a collection of studies that answer similar research questions. These studies are known as *primary studies*. Meta-analysis uses statistical methods to produce an overall estimate of an effect, explore between-study heterogeneity, and investigate the impact of publication bias or, more generally, small-study effects on the final results. Pearson (1904) provides the earliest example of what we now call meta-analysis. In that reference, the average of study-specific correlation coefficients was used to estimate an overall effect of vaccination against smallpox on subjects' survival.

There is a lot of information reported by a myriad of studies, which can be intimidating and difficult to absorb. Additionally, these studies may report conflicting results in terms of the magnitudes and even direction of the effects of interest. For example, many studies that investigated the effect of taking aspirin for preventing heart attacks reported contradictory results. Meta-analysis provides a principled approach for consolidating all of this overwhelming information to provide an overall conclusion or reasons for why such a conclusion cannot be reached.

Meta-analysis has been used in many fields of research. See the Cochrane Collaboration (https://us.cochrane.org/) for a collection of results from meta-analysis that address various treatments from all areas of healthcare. Meta-analysis has also been used in econometrics (for example, Dalhuisen et al. [2003]; Woodward and Wui [2001]; Hay, Knechel, and Wang [2006]; Card, Kluve, and Weber [2010]); education (for example, Bernard et al. [2004]; Fan and Chen [2001]); psychology (for example, Sin and Lyubomirsky [2009]; Barrick and Mount [1991]; Harter, Schmidt, and Hayes [2002]); psychiatry (for example, Hanji 2017); criminology (for example, Gendreau, Little, and Goggin [1996]; Pratt and Cullen [2000]); and ecology (for example, Hedges, Gurevitch, and Curtis [1999]; Gurevitch, Curtis, and Jones [2001]; Winfree et al. [2009]; Arnqvist and Wooster [1995]).

Meta-analysis is the statistical-analysis step of a systematic review. The term systematic review refers to the entire process of integrating the empirical research to achieve unified and potentially more general conclusions. Meta-analysis provides the theoretical underpinning of a systematic review and sets it apart from a narrative review; in the latter, an area expert summarizes the study-specific results and provides final conclusions, which could lead to potentially subjective and difficult-to-replicate findings. The theoretical soundness of meta-analysis made systematic reviews the method of choice for integrating empirical evidence from multiple studies. See Cooper, Hedges, and Valentine (2019) for more information as well as for various stages of a systematic review.

In what follows, we briefly describe the main components of meta-analysis: effect sizes, forest plots, heterogeneity, and publication bias.

Effect sizes. Effect sizes (or various measures of outcome) and their standard errors are the two most important components of a meta-analysis. They are obtained from each of the primary studies prior to the meta-analysis. Effect sizes of interest depend on the research objective and type of study. For example, in a meta-analysis comparing two groups, odds ratios and risk ratios are commonly used for binary outcomes and Hedges's g and Cohen's d measures for continuous outcomes. For meta-analysis estimating a single proportion (prevalence), the Freeman–Tukey-transformed proportions are typically used. When you deal with correlation data, the Fisher's z-transformed correlations are often used as the effect size. An overall effect size is computed as a weighted average of study-specific effect sizes, with more precise (larger) studies having larger weights. The weights are determined by the chosen meta-analysis model; see *Meta-analysis models*. Also see [META] **meta esize** for how to compute various effect sizes in a meta-analysis.

Meta-analysis models. Another important consideration for meta-analysis is that of the underlying model. Three commonly used models are a common-effect, fixed-effects, and random-effects models. The models differ in how they estimate and interpret parameters. See *Meta-analysis models* for details.

Meta-analysis summary—forest plots. The results of meta-analysis are typically summarized on a forest plot, which plots the study-specific effect sizes and their corresponding confidence intervals, the combined estimate of the effect size and its confidence interval, and other summary measures such as heterogeneity statistics. See *Forest plots* for details.

Heterogeneity. The estimates of effect sizes from individual studies will inherently vary from one study to another. This variation is known as a study heterogeneity. Two types of heterogeneity described by Deeks, Higgins, and Altman (2017) are methodological, when the studies differ in design and conduct, and clinical, when the studies differ in participants, treatments, and exposures or outcomes. The authors also define statistical heterogeneity, which exists when the observed effects differ between the studies. It is typically a result of clinical heterogeneity, methodological heterogeneity, or both. There are methods for assessing and addressing heterogeneity that we discuss in detail in *Heterogeneity*.

Publication bias. The selection of studies in a meta-analysis is an important step. Ideally, all studies that meet prespecified selection criteria must be included in the analysis. This is rarely achievable in practice. For instance, it may not be possible to have access to some unpublished results. So some of the relevant studies may be omitted from the meta-analysis. This may lead to what is known in statistics as a sample-selection problem. In the context of meta-analysis, this problem is known as publication bias or, more generally, reporting bias. Reporting bias arises when the omitted studies are systematically different from the studies selected in the meta-analysis. For details, see *Publication bias*.

Finally, you may ask, Does it make sense to combine different studies? According to Borenstein et al. (2009, chap. 40), "in the early days of meta-analysis, Robert Rosenthal was asked whether it makes sense to perform a meta-analysis, given that the studies differ in various ways and that the analysis amounts to combining apples and oranges. Rosenthal answered that *combining apples and oranges makes sense if your goal is to produce a fruit salad*."

Meta-analysis would be of limited use if it could combine the results of identical studies only. The appeal of meta-analysis is that it actually provides a principled way of combining a broader set of studies and can answer broader questions than those originally posed by the included primary studies. The specific goals of the considered meta-analysis should determine which studies can be combined and, more generally, whether a meta-analysis is even applicable.

Meta-analysis models

The role of a meta-analysis model is important for the computation and interpretation of the meta-analysis results. Different meta-analysis models make different assumptions and, as a result, estimate different parameters of interest. In this section, we describe the available meta-analysis models and point out the differences between them.

Suppose that there are K independent studies. Each study reports an estimate, $\widehat{\theta}_j$, of the unknown true effect size θ_j and an estimate, $\widehat{\sigma}_j$, of its standard error, $j = 1, 2, \ldots, K$. The goal of a meta-analysis is to combine these estimates in a single result to obtain valid inference about the population parameter of interest, θ_{pop}.

Depending on the research objective and assumptions about studies, three approaches are available to model the effect sizes: a common-effect model (historically known as a fixed-effect model—notice the singular "effect"), a fixed-effects model (notice the plural "effects"), and a random-effects model. We briefly define the three models next and describe them in more detail later.

Consider the model

$$\hat{\theta}_j = \theta_j + \epsilon_j \qquad j = 1, 2, \ldots, K \tag{1}$$

where ϵ_j's are sampling errors and $\epsilon_j \sim N(0, \sigma_j^2)$. Although σ_j^2's are unknown, meta-analysis does not estimate them. Instead, it treats the estimated values, $\hat{\sigma}_j^2$'s, of these variances as known and uses them during estimation. In what follows, we will thus write $\epsilon_j \sim N(0, \hat{\sigma}_j^2)$.

A common-effect model, as suggested by its name, assumes that all study effect sizes in (1) are the same and equal to the true effect size θ; that is, $\theta_j = \theta_{j'} = \theta$ for $j \neq j'$. The research questions and inference relies heavily on this assumption, which is often violated in practice.

A fixed-effects model assumes that the study effect sizes in (1) are different, $\theta_j \neq \theta_{j'}$ for $j \neq j'$, and "fixed". That is, the studies included in the meta-analysis define the entire population of interest. So the research questions and inference concern only the specific K studies included in the meta-analysis.

A random-effects model also assumes that the study effect sizes in (1) are different, $\theta_j \neq \theta_{j'}$ for $j \neq j'$, but that they are "random". That is, the studies in the meta-analysis represent a sample from a population of interest. The research questions and inference extend beyond the K studies included in the meta-analysis to the entire population of interest.

The models differ in the population parameter, θ_{pop}, they estimate; see *Comparison between the models and interpretation of their results*. Nevertheless, they all use the weighted average as the estimator for θ_{pop}:

$$\hat{\theta}_{\text{pop}} = \frac{\sum_{j=1}^{K} w_j \hat{\theta}_j}{\sum_{j=1}^{K} w_j} \tag{2}$$

However, they differ in how they define the weights w_j.

We describe each model and the parameter they estimate in more detail below.

Common-effect ("fixed-effect") model

As we mentioned earlier, a common-effect (CE) meta-analysis model (Hedges 1982; Rosenthal and Rubin 1982) is historically known as a fixed-effect model. The term "fixed-effect model" is easy to confuse with the "fixed-effects model" (plural), so we avoid it in our documentation. The term "common-effect", as suggested by Rice, Higgins, and Lumley (2018), is also more descriptive of the underlying model assumption. A CE model assumes a common (one true) effect for all studies in (1):

$$\hat{\theta}_j = \theta + \epsilon_j \qquad j = 1, 2, \ldots, K$$

The target of interest in a CE model is an estimate of a common effect size, $\theta_{\text{pop}} = \theta$. The CE model generally uses the weights $w_j = 1/\hat{\sigma}_j^2$ in (2) to estimate θ.

CE models are applicable only when the assumption that the same parameter underlies each study is reasonable, such as with pure replicate studies.

Fixed-effects model

A fixed-effects (FE) meta-analysis model (Hedges and Vevea 1998; Rice, Higgins, and Lumley 2018) is defined by (1); it assumes that different studies have different effect sizes ($\theta_1 \neq \theta_2 \neq \cdots \neq \theta_K$) and that the effect sizes are fixed quantities. By fixed quantities, we mean that the studies included in the meta-analysis define the entire population of interest. FE models are typically used whenever the analyst wants to make inferences only about the included studies.

The target of interest in an FE model is an estimate of the weighted average of true study-specific effect sizes,

$$\theta_{\text{pop}} = \text{Ave}(\theta_j) = \frac{\sum_{j=1}^{K} W_j\theta_j}{\sum_{j=1}^{K} W_j}$$

where W_j's represent true, unknown weights, which are defined in Rice, Higgins, and Lumley (2018, eq. 3). The estimated weights, $w_j = 1/\hat{\sigma}_j^2$, are generally used in (2) to estimate θ_{pop}.

Based on Rice, Higgins, and Lumley (2018), an FE model answers the question, "What is the magnitude of the average true effects in the set of K studies included in the meta-analysis?" It is appropriate when the true effects sizes are different across studies and the research interest lies in their average estimate.

Random-effects model

A random-effects (RE) meta-analysis model (Hedges 1983; DerSimonian and Laird 1986) assumes that the study effect sizes are different and that the collected studies represent a random sample from a larger population of studies. (The viewpoint of random effect sizes is further explored by Bayesian meta-analysis; see, for example, *Random-effects meta-analysis of clinical trials* in [BAYES] **bayesmh**.) The goal of RE meta-analysis is to provide inference for the population of studies based on the sample of studies used in the meta-analysis.

The RE model may be described as

$$\hat{\theta}_j = \theta_j + \epsilon_j = \theta + u_j + \epsilon_j$$

where $u_j \sim N(0, \tau^2)$ and, as before, $\epsilon_j \sim N(0, \hat{\sigma}_j^2)$. Parameter τ^2 represents the between-study variability and is often referred to as the heterogeneity parameter. It estimates the variability among the studies, beyond the sampling variability. When $\tau^2 = 0$, the RE model reduces to the CE model.

Here the target of inference is $\theta_{\text{pop}} = E(\theta_j)$, the mean of the distribution of effect sizes θ_j's. θ_{pop} is estimated from (2) with $w_j = 1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$.

Comparison between the models and interpretation of their results

CE and FE models are computationally identical but conceptually different. They differ in their target of inference and the interpretation of the overall effect size. In fact, all three models have important conceptual and interpretation differences. table 1 summarizes the different interpretations of θ_{pop} under the three models.

Table 1. Interpretation of θ_{pop} under various meta-analysis models

Model	Interpretation of θ_{pop}
common-effect	common effect ($\theta_1 = \theta_2 = \cdots = \theta_K = \theta$)
fixed-effects	weighted average of the K true study effects
random-effects	mean of the distribution of $\theta_j = \theta + u_j$

A CE meta-analysis model estimates the true effect size under the strong assumption that all studies share the same effect and thus all the variability between the studies is captured by the sampling errors. Under that assumption, the weighted average estimator indeed estimates the true common effect size, θ.

In the presence of additional variability unexplained by sampling variations, the interpretation of the results depends on how this variability is accounted for in the analysis.

An FE meta-analysis model uses the same weighted average estimator as a CE model, but the latter now estimates the weighted average of the K true study-specific effect sizes, Ave(θ_j).

An RE meta-analysis model assumes that the study contributions, u_j's, are random. It decomposes the variability of the effect sizes into the between-study and within-study components. The within-study variances, $\hat{\sigma}_j^2$'s, are assumed known by design. The between-study variance, τ^2, is estimated from the sample of the effect sizes. Thus, the extra variability attributed to τ^2 is accounted for during the estimation of the mean effect size, $E(\theta_j)$.

So which model should you choose? The literature recommends to start with a random-effects model, which is Stata's default for most meta-analyses. If you are willing to assume that the studies have different true effect sizes and you are interested only in providing inferences about these specific studies, then the FE model is appropriate. If the assumption of study homogeneity is reasonable for your data, a CE model may be considered.

Meta-analysis estimation methods

Depending on the chosen meta-analysis model, various methods are available to estimate the weights w_j in (2). The meta-analysis models from the previous sections assumed the inverse-variance estimation method (Whitehead and Whitehead 1991) under which the weights are inversely related to the variance. The inverse-variance estimation method is applicable to all meta-analysis models and all types of effect sizes. Thus, it can be viewed as the most general approach.

For a two-group comparison of binary outcomes, CE and FE models also support the Mantel–Haenszel estimation method, which can be used to combine odds ratios, risk ratios, and risk differences. The classical Mantel–Haenszel method (Mantel and Haenszel 1959) is used for odds ratios, and its extension by Greenland and Robins (1985) is used for risk ratios and risk differences. The Mantel–Haenszel method is recommended with sparse data. Fleiss, Levin, and Paik (2003) also suggests that it be used with small studies provided that there are many.

In RE models, the weights are inversely related to the total variance, $w_j = 1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$. Different methods are proposed for estimating the between-study variability, τ^2, which is used in the expression for the weights. These include the restricted maximum likelihood (REML), maximum likelihood (ML), empirical Bayes (EB), DerSimonian–Laird (DL), Hedges (HE), Sidik–Jonkman (SJ), and Hunter–Schmidt (HS).

REML, ML, and EB are iterative methods, whereas other methods are noniterative (have closed-form expressions). The former estimators produce nonnegative estimates of τ^2. The other estimators, except SJ, may produce negative estimates and are thus truncated at zero when this happens. The SJ estimator always produces a positive estimate of τ^2.

REML, ML, and EB assume that the distribution of random effects is normal. The other estimators make no distributional assumptions about random effects. Below, we briefly describe the properties of each method. See Sidik and Jonkman (2007), Viechtbauer (2005), and Veroniki et al. (2016) for a detailed discussion and the merit of each estimation method.

The REML method (Raudenbush 2009) produces an unbiased, nonnegative estimate of τ^2 and is commonly used in practice. (It is the default estimation method in Stata because it performs well in most scenarios.)

When the number of studies is large, the ML method (Hardy and Thompson 1998; Thompson and Sharp 1999) is more efficient than the REML method but may produce biased estimates when the number of studies is small, which is a common case in meta-analysis.

The EB estimator (Berkey et al. 1995), also known as the Paule–Mandel estimator (Paule and Mandel 1982), tends to be less biased than other RE methods, but it is also less efficient than REML or DL (Knapp and Hartung 2003).

The DL method (DerSimonian and Laird 1986), historically, is one of the most popular estimation methods because it does not make any assumptions about the distribution of the random effects and does not require iteration. But it may underestimate τ^2, especially when the variability is large and the number of studies is small. However, when the variability is not too large and the studies are of similar sizes, this estimator is more efficient than other noniterative estimators HE and SJ. See Veroniki et al. (2016) for details and relevant references.

The SJ estimator (Sidik and Jonkman 2005), along with the EB estimator, is the best estimator in terms of bias for large τ^2 (Sidik and Jonkman 2007). This method always produces a positive estimate of τ^2 and thus does not need truncating at 0, unlike the other noniterative methods.

Like DL, the HE estimator (Hedges 1983) is a method of moments estimator, but, unlike DL, it does not weight effect-size variance estimates (DerSimonian and Laird 1986). Veroniki et al. (2016) note, however, that this method is not widely used in practice.

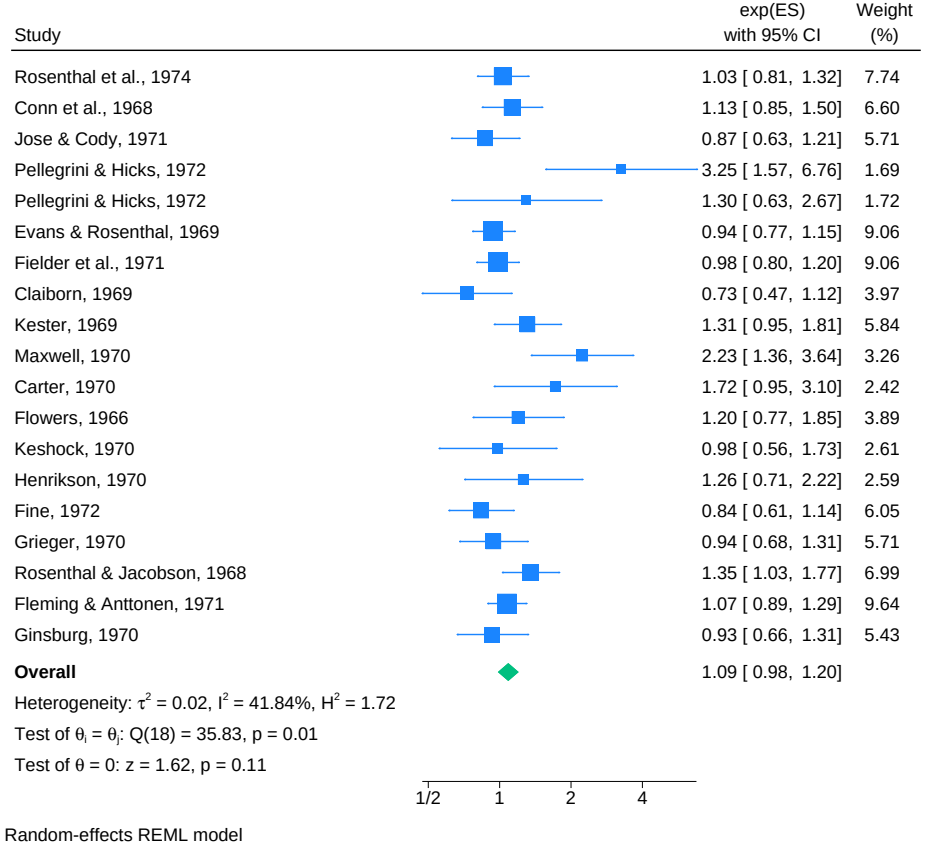
The HS estimator (Schmidt and Hunter 2015) is negatively biased and thus not recommended when unbiasedness is important (Viechtbauer 2005). Otherwise, the mean squared error of HS is similar to that of ML and is smaller than those of HE, DL, and REML.

Forest plots

Meta-analysis results are often presented using a forest plot (for example, Lewis and Ellis [1982]). A forest plot shows study-specific effect sizes and an overall effect size with their respective confidence intervals. The information about study heterogeneity and the significance of the overall effect size are also

typically presented. This plot provides a convenient way to visually compare the study effect sizes, which can be any summary estimates available from primary studies, such as standardized and unstandardized mean differences, (log) odds ratios, (log) risk ratios, and (log) hazard ratios.

Below is an example of a forest plot.



A blue square is plotted for each study, with the size of the square being proportional to the study weight; that is, larger squares correspond to larger (more precise) studies. Studies' CIs are plotted as whiskers extending from each side of the square and spanning the width of the CI. The estimate of the overall effect size, depicted here by a green diamond, is typically plotted following the individual effect sizes. The diamond is centered at the estimate of the overall effect size and the width of the diamond represents the corresponding CI width. Heterogeneity measures such as the I^2 and H^2 statistics, homogeneity test, and the significance test of the overall effect sizes are also commonly reported.

Three further variations of forest plots are for cumulative, subgroup, and leave-one-out meta-analyses; see *Cumulative meta-analysis*, *Subgroup meta-analysis*, and *Leave-one-out meta-analysis*.

For further details about forest plots, see [META] **meta forestplot**.

Heterogeneity

The exposition below is based on Deeks, Higgins, and Altman (2017) and references therein.

It is natural for effect sizes of studies collected in a meta-analysis to vary between the studies because of sampling variability. However, when this variation exceeds the levels that could be explained by sampling variation, it is referred to as the between-study heterogeneity. Between-study heterogeneity may arise for different reasons and is generally divided into two types: clinical and methodological (Thompson 1994; Deeks, Higgins, and Altman 2017). Clinical heterogeneity is the variability in the intervention strategies, outcomes, and study participants. Methodological heterogeneity is the variability in the study design and conduct. Statistical heterogeneity refers to the cases when the variability between the observed effects cannot be explained by sampling variability alone. It arises when the true effects in each study are different and may be the result of clinical heterogeneity, methodological heterogeneity, or both. In what follows, we refer to statistical heterogeneity simply as heterogeneity.

Assessing heterogeneity

Forest plots are useful for visual examination of heterogeneity. Its presence can be evaluated by looking at the plotted CIs, which are represented as horizontal lines on the plot. Heterogeneity is suspect if there is a lack of overlap between the CIs.

For many studies, Galbraith plots may be a more visually appealing alternative to forest plots for assessing heterogeneity and presenting meta-analysis results. These plots graph standardized effect sizes against precision for each study with a regression line through the origin with the overall effect size as its slope. Excess variation of the scatter points around the regression line may suggest the presence of heterogeneity. See [META] **meta galbraithplot**.

For a two-group comparison of binary outcomes, L'Abbé plots may be used to assess heterogeneity and compare study-specific event rates in the two groups; see [META] **meta labbeplot**.

You can also test for heterogeneity more formally by using Cochran's homogeneity test. Additionally, various heterogeneity measures such as the I^2 statistic, which estimates the percentage of the between-study variability, are available to quantify heterogeneity.

See [META] **meta summarize** for details.

Addressing heterogeneity

There are several strategies to address heterogeneity when it is present. Below, we summarize some of the recommendations from Deeks, Higgins, and Altman (2017):

1. "*Explore heterogeneity*". Subgroup analyses and meta-regression are commonly used to explore heterogeneity. For such analyses to be proper, you must prespecify upfront (before your meta-analysis) the study attributes you would like to explore. Often, meta-analysts are already familiar with the studies, so the genuine prestudy specification may not be possible. In that case, you should use caution when interpreting the results. Once heterogeneity is established, its exploration after the fact is viewed as data snooping and should be avoided.
2. "*Perform an RE meta-analysis*". After careful consideration of subgroup analysis and meta-regression, you may consider an RE meta-analysis to account for the remaining unexplained between-study heterogeneity. See Deeks, Higgins, and Altman (2017, sec. 9.5.4) for details.

3. *"Exclude studies"*. Generally, you should avoid excluding studies from a meta-analysis because this may lead to bias. You may consider doing this in the presence of a few outlying studies when the reasons for the outlying results are well understood and are unlikely to interfere with your research objectives. Even then, you still need to perform sensitivity analysis and report both the results with and without the outlying studies.

4. *"Do not perform a meta-analysis"*. In the presence of substantial variation that cannot be explained, you may have to abandon the meta-analysis altogether. In this case, it will be misleading to report a single overall estimate of an effect, especially if there is a disagreement among the studies about the direction of the effect.

Below, we discuss ways of exploring heterogeneity via subgroup meta-analysis and meta-regression.

Subgroup meta-analysis

It is not uncommon for the studies in a meta-analysis to report varying effect-size estimates. But it is important to understand and account for such variation during the meta-analysis to obtain reliable results (Thompson 1994; Berlin 1995). In the presence of substantial between-study variability, meta-analysis may be used to explore the relationship between the effect sizes and study-level covariates of interest, known in the meta-analysis literature as moderators. For example, the effect of a particular vaccine may depend on a study location, the effect of a particular drug may depend on the studies' dosages, and so on.

Depending on the type of covariates, subgroup meta-analysis or meta-regression may be used to explore the between-study heterogeneity. Subgroup meta-analysis is commonly used with categorical covariates, whereas meta-regression is used when at least one of the covariates is continuous.

In subgroup meta-analysis or simply subgroup analysis, the studies are grouped based on study or participants' characteristics, and an overall effect-size estimate is computed for each group. The goal of subgroup analysis is to compare these overall estimates across groups and determine whether the considered grouping helps explain some of the observed between-study heterogeneity. Note that subgroup analysis can be viewed as a special case of a meta-regression with only one categorical moderator.

For more details about subgroup analysis, see the `subgroup()` option in [META] **meta summarize** and [META] **meta forestplot**.

Meta-regression

Meta-regression explores a relationship between the study-specific effect sizes and the study-level covariates, such as a latitude of a study location or a dosage of a drug. These covariates are often referred to as moderators. See, for instance, Greenland (1987), Berkey et al. (1995), Thompson and Sharp (1999), Thompson and Higgins (2002), and Viechtbauer et al. (2015) for more information about meta-regression.

Two types of meta-regression are commonly considered in the meta-analysis literature: fixed-effects meta-regression and random-effects meta-regression.

An FE meta-regression (Greenland 1987) assumes that all heterogeneity between the study outcomes can be accounted for by the specified moderators. Let $\mathbf{x}_j$ be a $1 \times p$ vector of moderators with the corresponding unknown $p \times 1$ coefficient vector $\boldsymbol{\beta}$. An FE meta-regression is given by

$$\hat{\theta}_j = \mathbf{x}_j\boldsymbol{\beta} + \epsilon_j \qquad \text{weighted by } w_j = \frac{1}{\hat{\sigma}_j^2}, \text{ where } \epsilon_j \sim N(0, \hat{\sigma}_j^2)$$

A traditional FE meta-regression does not model residual heterogeneity, but it can be incorporated by multiplying each of the variances, $\widehat{\sigma}^2_j$, by a common factor. This model is known as an FE meta-regression with a multiplicative dispersion parameter or a multiplicative FE meta-regression (Thompson and Sharp 1999).

An RE meta-regression (Berkey et al. 1995) can be viewed as a meta-regression that incorporates the residual heterogeneity via an additive error term, which is represented in a model by a study-specific random effect. These random effects are assumed to be normal with mean zero and variance τ^2, which estimates the remaining between-study heterogeneity that is unexplained by the considered moderators. An RE meta-regression is

$$\widehat{\theta}_j = \mathbf{x}_j\boldsymbol{\beta} + u_j + \epsilon_j \qquad \text{weighted by } w^*_j = \frac{1}{\widehat{\sigma}^2_j + \widehat{\tau}^2}, \text{ where } u_j \sim N(0,\tau^2) \text{ and } \epsilon_j \sim N(0,\widehat{\sigma}^2_j)$$

For more details about meta-regression, see [META] **meta regress** and [META] **meta regress postestimation**.

Publication bias

Publication bias or, more generally, reporting bias occurs when the studies selected for a scientific review are systematically different from all available relevant studies. Specifically, publication bias is known in the meta-analysis literature as an association between the likelihood of a publication and the statistical significance of a study result. The rise of systematic reviews for summarizing the results of scientific studies elevated the importance of acknowledging and addressing publication bias in research. Publication bias typically arises when nonsignificant results are being underreported in the literature (for example, Rosenthal [1979]; Iyengar and Greenhouse [1988]; Begg and Berlin [1988]; Hedges [1992]; Stern and Simes [1997]; Givens, Smith, and Tweedie [1997]; Sutton et al. [2000b]; and Kicinski, Springate, and Kontopantelis [2015]).

Suppose that we are missing some of the studies in our meta-analysis. If these studies are simply a random sample of all the studies that are relevant to our research question, our meta-analytic results will remain valid but will not be as precise. That is, we will likely obtain wider confidence intervals and less powerful tests. However, if the missing studies differ systematically from our observed studies, such as when smaller studies with nonsignificant findings are suppressed from publication, our meta-analytic results will be biased toward a significant result. Any health-policy or clinical decisions based on them will be invalid.

Dickersin (2005) notes that to avoid potentially serious consequences of publication bias, many researchers (for example, Simes [1986]; Dickersin [1988]; Hetherington et al. [1989]; Dickersin and Rennie [2003]; Antes and Chalmers [2003]; and Krakovsky [2004]) called for the registration of clinical trials worldwide at the outset to keep track of the findings, whether or not significant, from all trials. Although this may not necessarily eradicate the problem of publication bias, this will make it more difficult for the results of smaller trials to go undetected. Generally, when one selects the studies for meta-analysis, the review of the literature should be as comprehensive as possible, including searching the grey literature to uncover the relevant unpublished studies.

See Borenstein et al. (2009, chap. 30) for the summary of other factors for publication bias such as language bias and cost bias.

Funnel plots

The funnel plot (Light and Pillemer 1984) is commonly used to explore publication bias (Sterne, Becker, and Egger 2005). It is a scatterplot of the study-specific effect sizes versus measures of study precision. In the absence of publication bias, the shape of the scatterplot should resemble a symmetric inverted funnel. The funnel-plot asymmetry, however, may be caused by factors other than publication bias such as a presence of a moderator correlated with the study effect and study size or, more generally, the presence of substantial between-study heterogeneity (Egger et al. 1997 ; Peters et al. 2008 ; Sterne et al. 2011). The so-called contour-enhanced funnel plots have been proposed to help discriminate between the funnel-plot asymmetry because of publication bias versus other reasons.

See [META] **meta funnelplot** for details.

Tests for funnel-plot asymmetry

Graphical evaluation of funnel plots is useful for data exploration but may be subjective when detecting the asymmetry. Statistical tests provide a more formal evaluation of funnel-plot asymmetry. These tests are also known as tests for small-study effects (Sterne, Gavaghan, and Egger 2000) and, historically, as tests for publication bias. The tests are no longer referred to as "tests for publication bias" because, as we commented earlier, the presence of the funnel-plot asymmetry may not necessarily be attributed to publication bias, particularly in the presence of substantial between-study variability. See Harbord, Harris, and Sterne (2016) for a summary of these tests.

Two types of tests for funnel-plot asymmetry are considered in the literature: regression-based tests (Egger et al. 1997 ; Harbord, Egger, and Sterne 2006; and Peters et al. 2006) and a nonparametric rank-based test (Begg and Mazumdar 1994). These tests explore the relationship between the study-specific effect sizes and study precision. The presence of the funnel-plot asymmetry is declared when the association between the two measures is greater than what would have been observed by chance.

For more details regarding the tests of funnel-plot asymmetry, see [META] **meta bias**.

The trim-and-fill method

Tests for funnel-plot asymmetry are useful for detecting publication bias but are not able to estimate the impact of this bias on the final meta-analysis results. The nonparametric trim-and-fill method of Duval and Tweedie (2000a, 2000b) provides a way to assess the impact of missing studies because of publication bias on the meta-analysis. It evaluates the amount of potential bias present in meta-analysis and its impact on the final conclusion. This method is typically used as a sensitivity analysis to the presence of publication bias.

See [META] **meta trimfill** for more information about the trim-and-fill method.

Cumulative meta-analysis

Cumulative meta-analysis performs multiple meta-analyses, where each analysis is produced by adding one study at a time. It is useful to identify various trends in the overall effect sizes. For example, when the studies are ordered chronologically, one can determine the point in time of the potential change in the direction or significance of the effect size. A well-known example of a cumulative meta-analysis is presented in *Cumulative meta-analysis* of [META] **meta** for the study of the efficacy of streptokinase after a myocardial infarction (Lau et al. 1992). Also see the `cumulative()` option in [META] **meta summarize** and [META] **meta forestplot**.

Leave-one-out meta-analysis

Just like cumulative meta-analysis, the leave-one-out meta-analysis also performs multiple meta-analyses; however, in this case, each analysis is produced by excluding a single study. It is quite common that studies yield effect sizes that are relatively exaggerated. Their presence in the meta-analysis may distort the overall results, and it is of great importance to identify such studies for further examination. The leave-one-out meta-analysis is a useful tool to investigate the influence of each study on the overall effect size estimate. See the `leaveoneout` option in [META] **meta summarize** and [META] **meta forestplot** for more information.

Multivariate meta-regression

Multivariate meta-analysis combines results from studies where multiple dependent effect sizes (outcomes) are reported by each study. Let $\hat{\boldsymbol{\theta}}_j$ be a $d\times 1$ vector of estimates of the true population multivariate effect size $\boldsymbol{\theta}_j$ for study j. Let $\mathbf{x}_j$ be a $1 \times p$ vector of moderators with the corresponding unknown $p \times 1$ regression coefficient vector $\boldsymbol{\beta}_i$ for $i = 1, \ldots, d$.

An FE multivariate meta-regression (Raudenbush, Becker, and Kalaian 1988) is given by

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \boldsymbol{\epsilon}_j,\ \boldsymbol{\epsilon}_j \sim N_d\left(\mathbf{0},\ \boldsymbol{\Lambda}_j\right)$$

where $\mathbf{X}_j = \mathbf{x}_j \otimes I_d$ is a $d\times dp$ matrix and $\boldsymbol{\beta} = \left(\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \ldots, \boldsymbol{\beta}_d'\right)'$ is a $dp\times 1$ vector of unknown regression coefficients; $\otimes$ is the Kronecker product. The within-study covariance matrices $\boldsymbol{\Lambda}_j$'s are assumed known and thus do not require estimation.

The RE multivariate meta-regression (Berkey et al. 1998) can be expressed as

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \boldsymbol{\epsilon}_j^* = \mathbf{X}_j\boldsymbol{\beta} + \mathbf{u}_j + \boldsymbol{\epsilon}_j, \text{ where } \boldsymbol{\epsilon}_j^* \sim N_d\left(\mathbf{0}, \boldsymbol{\Lambda}_j + \boldsymbol{\Sigma}\right)$$

where $\mathbf{u}_j$ is a $d \times 1$ vector of random effects corresponding to the d outcomes.

`meta mvregress` fits multivariate meta-regression; see [META] **meta mvregress**. By default, a random-effects model is assumed. The goal of multivariate meta-regression is to estimate the regression coefficients $\boldsymbol{\beta}$ and the random-effects covariance matrix $\boldsymbol{\Sigma}$, also known as the between-study covariance matrix. Three estimation methods are available to fit the RE multivariate meta-regression model and multiple covariance structures can be specified to model the between-study covariance $\boldsymbol{\Sigma}$. After fitting the multivariate meta-regression model, you can assess heterogeneity; see [META] **estat heterogeneity (mv)**. Various postestimation tools are available such as predicting random effects, computing the linear predictor, residuals, standardized residuals, and more; see [META] **meta mvregress postestimation**.

Multilevel meta-regression

Multilevel meta-analysis synthesizes the results from potentially dependent effect sizes that exhibit a hierarchical or nested structure. For example, studies and their corresponding effect sizes may be nested within higher-level groupings such as geographical locations (for example, states or countries) or administrative units (for example, school districts).

When a hierarchical structure is present in the data, the multilevel meta-analysis is preferred over the classical meta-analysis. By properly accounting for the hierarchical structure among the effect sizes, we can obtain more accurate estimates of the overall effect size and better overall statistical inference. We can also decompose the heterogeneity present among the effect sizes across the different hierarchical levels, which can provide valuable insights into the factors that affect our outcome of interest.

The three-level meta-regression model (for example, Goldstein et al. [2000]; Thompson, Turner, and Warn [2001]; and Konstantopoulos [2011]) can be expressed as

$$\hat{\theta}_{jkr} = \mathbf{x}_{jkr}\boldsymbol{\beta} + \mathbf{z}^{(3)}_{jkr}\mathbf{u}^{(3)}_{j} + \mathbf{z}^{(2)}_{jkr}\mathbf{u}^{(2)}_{jk} + \epsilon_{jkr}$$

where $j = 1, 2, \ldots, M$, $k = 1, 2, \ldots, m_j$, and $r = 1, 2, \ldots, m_{jk}$. In this case, $\mathbf{x}_{jkr} = (1, x_{1,jkr}, \ldots, x_{p-1,jkr})$ is a $1 \times p$ vector of moderators and $\boldsymbol{\beta}$ is the corresponding $p \times 1$ vector of unknown fixed-effects parameters. $\mathbf{z}^{(3)}_{jkr}$ is a $1 \times q_3$ vector of moderators associated with the level-3 $q_3 \times 1$ vector of random effects $\mathbf{u}^{(3)}_{j}$ (1 intercept and $q_3 - 1$ slopes), where $\mathbf{u}^{(3)}_{j} \sim N(\mathbf{0}, \boldsymbol{\Sigma}^{(3)})$. Similarly, $\mathbf{z}^{(2)}_{jkr}$ is a $1 \times q_2$ vector of moderators associated with the level-2 (within-level-3) $q_2 \times 1$ vector of random effects $\mathbf{u}^{(2)}_{jk}$, where $\mathbf{u}^{(2)}_{jk} \sim N(\mathbf{0}, \boldsymbol{\Sigma}^{(2)})$. $\epsilon_{jkr} \sim N(0, \hat{\sigma}^2_{jkr})$, where $\hat{\sigma}^2_{jkr}$'s are known sampling (effect-size) variances.

`meta meregress` fits multilevel meta-regression; see [META] **meta meregress**. If your model contains only random intercepts (no random slopes), you may use the `meta multilevel` command, which has a simpler syntax geared toward random-intercepts multilevel models; see [META] **meta multilevel**. The goal of multilevel meta-regression is to estimate the regression coefficients $\boldsymbol{\beta}$ and the random-effects covariance matrices $\boldsymbol{\Sigma}^{(l)}$ for each level $l > 1$. By default, the REML estimation method is assumed, but the MLE method is also supported. Multiple covariance structures can be specified to model the $\boldsymbol{\Sigma}^{(l)}$ matrices. After fitting the multilevel meta-regression model, you can assess multilevel heterogeneity; see [META] **estat heterogeneity (me)**. Various postestimation tools are available, such as predicting random effects and computing the linear predictor, residuals, standardized residuals, and more; see [META] **meta me postestimation**.

References

Antes, G., and I. Chalmers. 2003. Under-reporting of clinical trials is unethical. *Lancet* 361: 978–979. https://doi.org/10.1016/S0140-6736(03)12838-3.

Arnqvist, G., and D. Wooster. 1995. Meta-analysis: Synthesizing research findings in ecology and evolution. *Trends in Ecology and Evolution* 10: 236–240. https://doi.org/10.1016/S0169-5347(00)89073-4.

Barrick, M. R., and M. K. Mount. 1991. The big five personality dimensions and job performance: A meta-analysis. *Personnel Psychology* 44: 1–26. https://doi.org/10.1111/j.1744-6570.1991.tb00688.x.

Begg, C. B., and J. A. Berlin. 1988. Publication bias: A problem in interpreting medical data. *Journal of the Royal Statistical Society*, A ser., 151: 419–463. https://doi.org/10.2307/2982993.

Begg, C. B., and M. Mazumdar. 1994. Operating characteristics of a rank correlation test for publication bias. *Biometrics* 50: 1088–1101. https://doi.org/10.2307/2533446.

Berkey, C. S., D. C. Hoaglin, F. Mosteller, and G. A. Colditz. 1995. A random-effects regression model for meta-analysis. *Statistics in Medicine* 14: 395–411. https://doi.org/10.1002/sim.4780140406.

———. 1998. Meta-analysis of multiple outcomes by regression with random effects. *Statistics in Medicine* 17: 2537–2550. https://doi.org/10.1002/(SICI)1097-0258(19981130)17:22%3C2537::AID-SIM953%3E3.0.CO;2-C.

Berlin, J. A. 1995. Invited commentary: Benefits of heterogeneity in meta-analysis of data from epidemiologic studies. *American Journal of Epidemiology* 142: 383–387. https://doi.org/10.1093/oxfordjournals.aje.a117645.

Bernard, R. M., P. C. Abrami, Y. Lou, E. Borokhovski, A. Wade, L. Wozney, P. A. Wallet, M. Fiset, and B. Huang. 2004. How does distance education compare with classroom instruction? A meta-analysis of the empirical literature. *Review of Educational Research* 74: 379–439. https://doi.org/10.3102/00346543074003379.

Borenstein, M., L. V. Hedges, J. P. T. Higgins, and H. R. Rothstein. 2009. *Introduction to Meta-Analysis*. Chichester, UK: Wiley.

Card, D., J. Kluve, and A. Weber. 2010. Active labour market policy evaluations: A meta-analysis. *Economic Journal* 120: F452–F477. https://doi.org/10.1111/j.1468-0297.2010.02387.x.

Cooper, H., L. V. Hedges, and J. C. Valentine, eds. 2019. *The Handbook of Research Synthesis and Meta-Analysis*. 3rd ed. New York: Russell Sage Foundation.

Dalhuisen, J. M., R. J. G. M. Florax, H. L. F. de Groot, and P. Nijkamp. 2003. Price and income elasticities of residential water demand: A meta-analysis. *Land Economics* 79: 292–308. https://doi.org/10.2307/3146872.

Deeks, J. J., J. P. T. Higgins, and D. G. Altman. 2017. "Analysing data and undertaking meta-analyses". In *Cochrane Handbook for Systematic Reviews of Interventions Version 5.2.0*, edited by J. P. T. Higgins and S. Green. London: The Cochrane Collaboration. https://training.cochrane.org/handbook.

DerSimonian, R., and N. M. Laird. 1986. Meta-analysis in clinical trials. *Controlled Clinical Trials* 7: 177–188. https://doi.org/10.1016/0197-2456(86)90046-2.

Dickersin, K. 1988. Report from the panel on the case for registers of clinical trials at the eighth annual meeting of the society for clinical trials. *Controlled Clinical Trials* 9: 76–80. https://doi.org/10.1016/0197-2456(88)90010-4.

———. 2005. "Publication bias: Recognizing the problem, understanding its origins and scope, and preventing harm". In *Publication Bias in Meta-Analysis: Prevention, Assessment and Adjustments*, edited by H. R. Rothstein, A. J. Sutton, and M. Borenstein, 9–33. Chichester, UK: Wiley. https://doi.org/10.1002/0470870168.ch2.

Dickersin, K., and D. Rennie. 2003. Registering clinical trials. *Journal of the American Medical Association* 290: 516–523. https://doi.org/10.1001/jama.290.4.516.

Duval, S., and R. L. Tweedie. 2000a. Trim and fill: A simple funnel-plot–based method of testing and adjusting for publication bias in meta-analysis. *Biometrics* 56: 455–463. https://doi.org/10.1111/j.0006-341X.2000.00455.x.

———. 2000b. A nonparametric "trim and fill" method of accounting for publication bias in meta-analysis. *Journal of the American Statistical Association* 95: 89–98. https://doi.org/10.1080/01621459.2000.10473905.

Egger, M., G. Davey Smith, M. Schneider, and C. Minder. 1997. Bias in meta-analysis detected by a simple, graphical test. *BMJ* 315: 629–634. https://doi.org/10.1136/bmj.315.7109.629.

Fan, X., and M. Chen. 2001. Parental involvement and students' academic achievement: A meta-analysis. *Educational Psychology Review* 13: 1–22. https://doi.org/10.1023/A:1009048817385.

Fleiss, J. L., B. Levin, and M. C. Paik. 2003. *Statistical Methods for Rates and Proportions*. 3rd ed. New York: Wiley. https://doi.org/10.1002/0471445428.

Gendreau, P., T. Little, and C. Goggin. 1996. A meta-analysis of the predictors of adult offender recidivism: What works! *Criminology* 34: 575–608. https://doi.org/10.1111/j.1745-9125.1996.tb01220.x.

Givens, G. H., D. D. Smith, and R. L. Tweedie. 1997. Publication bias in meta-analysis: A Bayesian data-augmentation approach to account for issues exemplified in the passive smoking debate. *Statistical Science* 12: 221–240. https://doi.org/10.1214/ss/1030037958.

Glass, G. V. 1976. Primary, secondary, and meta-analysis of research. *Educational Researcher* 5: 3–8. https://doi.org/10.2307/1174772.

Goldstein, H., M. Yang, R. Z. Omar, R. M. Turner, and S. G. Thompson. 2000. Meta-analysis using multilevel models with an application to the study of class size effects. *Journal of the Royal Statistical Society*, C ser., 49: 399–412. https://doi.org/10.1111/1467-9876.00200.

Greenland, S. 1987. Quantitative methods in the review of epidemiologic literature. *Epidemiologic Reviews* 9: 1–30. https://doi.org/10.1093/oxfordjournals.epirev.a036298.

Greenland, S., and J. M. Robins. 1985. Estimation of a common effect parameter from sparse follow-up data. *Biometrics* 41: 55–68. https://doi.org/10.2307/2530643.

Gurevitch, J., P. S. Curtis, and M. H. Jones. 2001. Meta-analysis in ecology. *Advances in Ecology* 32: 199–247. https://doi.org/10.1016/S0065-2504(01)32013-5.

Hanji, M. B. 2017. *Meta-Analysis in Psychiatry Research: Fundamental and Advanced Methods*. Waretown, NJ: Apple Academic Press. https://doi.org/10.1201/9781315366234.

Harbord, R. M., M. Egger, and J. A. C. Sterne. 2006. A modified test for small-study effects in meta-analyses of controlled trials with binary endpoints. *Statistics in Medicine* 25: 3443–3457. https://doi.org/10.1002/sim.2380.

Harbord, R. M., R. J. Harris, and J. A. C. Sterne. 2016. "Updated tests for small-study effects in meta-analyses". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 153–165. 2nd ed. College Station, TX: Stata Press.

Hardy, R. J., and S. G. Thompson. 1998. Detecting and describing heterogeneity in meta-analysis. *Statistics in Medicine* 17: 841–856. https://doi.org/10.1002/(sici)1097-0258(19980430)17:8<841::aid-sim781>3.0.co;2-d.

Harter, J. K., F. L. Schmidt, and T. L. Hayes. 2002. Business-unit-level relationship between employee satisfaction, employee engagement, and business outcomes: A meta-analysis. *Applied Psychology* 87: 268–279. https://doi.org/10.1037/0021-9010.87.2.268.

Hay, D. C., W. R. Knechel, and N. Wang. 2006. Audit fees: A meta-analysis of the effect of supply and demand attributes. *Contemporary Accounting Research* 23: 141–191. https://doi.org/10.1506/4XR4-KT5V-E8CN-91GX.

Hedges, L. V. 1982. Estimation of effect size from a series of independent experiments. *Psychological Bulletin* 92: 490–499. https://doi.org/10.1037/0033-2909.92.2.490.

———. 1983. A random effects model for effect sizes. *Psychological Bulletin* 93: 388–395. https://doi.org/10.1037/0033-2909.93.2.388.

———. 1992. Modeling publication selection effects in meta-analysis. *Statistical Science* 7: 246–255. https://doi.org/10.1214/ss/1177011364.

Hedges, L. V., J. Gurevitch, and P. S. Curtis. 1999. The meta-analysis of response ratios in experimental ecology. *Ecology* 80: 1150–1156. https://doi.org/10.2307/177062.

Hedges, L. V., and I. Olkin. 1985. *Statistical Methods for Meta-Analysis*. Orlando, FL: Academic Press.

Hedges, L. V., and J. L. Vevea. 1998. Fixed- and random-effects models in meta-analysis. *Psychological Methods* 3: 486–504. https://doi.org/10.1037/1082-989X.3.4.486.

Hetherington, J., K. Dickersin, I. Chalmers, and C. L. Meinert. 1989. Retrospective and prospective identification of unpublished controlled trials: Lessons from a survey of obstetricians and pediatricians. *Pediatrics* 84: 374–380.

Higgins, J. P. T., and S. Green, eds. 2017. *Cochrane Handbook for Systematic Reviews of Interventions Version 5.2.0*. London: The Cochrane Collaboration. https://training.cochrane.org/handbook.

Iyengar, S., and J. B. Greenhouse. 1988. Selection models and the file drawer problem. *Statistical Science* 3: 109–117. https://doi.org/10.1214/ss/1177013012.

Kicinski, M., D. A. Springate, and E. Kontopantelis. 2015. Publication bias in meta-analyses from the Cochrane Database of Systematic Reviews. *Statistics in Medicine* 34: 2781–2793. https://doi.org/10.1002/sim.6525.

Knapp, G., and J. Hartung. 2003. Improved tests for a random effects meta-regression with a single covariate. *Statistics in Medicine* 22: 2693–2710. https://doi.org/10.1002/sim.1482.

Konstantopoulos, S. 2011. Fixed effects and variance components estimation in three-level meta-analysis. *Research Synthesis Methods* 2: 61–76. https://doi.org/10.1002/jrsm.35.

Kontopantelis, E., and D. Reeves. 2020. Pairwise meta-analysis of aggregate data using metaan in Stata. *Stata Journal* 20: 680–705.

Krakovsky, M. 2004. Register or perish: Looking to make the downside of therapies known. *Scientific American* 6: 18–20.

Lau, J., E. M. Antman, J. Jimenez-Silva, B. Kupelnick, F. Mosteller, and T. C. Chalmers. 1992. Cumulative meta-analysis of therapeutic trials for myocardial infarction. *New England Journal of Medicine* 327: 248–254. https://doi.org/10.1056/NEJM199207233270406.

Lewis, J. A., and S. H. Ellis. 1982. A statistical appraisal of post-infarction beta-blocker trials. *Primary Cardiology* Suppl. 1: 31–37.

Light, R. J., and D. B. Pillemer. 1984. *Summing Up: The Science of Reviewing Research*. Cambridge, MA: Harvard University Press. https://doi.org/10.2307/j.ctvk12px9.

Linden, A. 2022. Computing the fragility index for randomized trials and meta-analyses using Stata. *Stata Journal* 22: 77–88.

Mantel, N., and W. Haenszel. 1959. Statistical aspects of the analysis of data from retrospective studies of disease. *Journal of the National Cancer Institute* 22: 719–748. Reprinted in *Evolution of Epidemiologic Ideas: Annotated Readings on Concepts and Methods*, ed. S. Greenland, pp. 112–141. Newton Lower Falls, MA: Epidemiology Resources.

Palmer, T. M., and J. A. C. Sterne, eds. 2016. *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*. 2nd ed. College Station, TX: Stata Press.

Paule, R. C., and J. Mandel. 1982. Consensus values and weighting factors. *Journal of Research of the National Bureau of Standards* 87: 377–385. https://doi.org/10.6028/jres.087.022.

Pearson, K. 1904. Report on certain enteric fever inoculation statistics. *BMJ* 2: 1243–1246. https://doi.org/10.1136/bmj.2.2288.1243.

Peters, J. L., A. J. Sutton, D. R. Jones, K. R. Abrams, and L. Rushton. 2006. Comparison of two methods to detect publication bias in meta-analysis. *Journal of the American Medical Association* 295: 676–680. https://doi.org/10.1001/jama.295.6.676.

———. 2008. Contour-enhanced meta-analysis funnel plots help distinguish publication bias from other causes of asymmetry. *Journal of Clinical Epidemiology* 61: 991–996. https://doi.org/10.1016/j.jclinepi.2007.11.010.

Pratt, T. C., and F. T. Cullen. 2000. The empirical status of Gottfredson and Hirschi's general theory of crime: A meta-analysis. *Criminology* 38: 931–964. https://doi.org/10.1111/j.1745-9125.2000.tb00911.x.

Raudenbush, S. W. 2009. "Analyzing effect sizes: Random-effects models". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 295–316. 2nd ed. New York: Russell Sage Foundation.

Raudenbush, S. W., B. J. Becker, and H. Kalaian. 1988. Modeling multivariate effect sizes. *Psychological Bulletin* 103: 111–120. https://doi.org/10.1037/0033-2909.103.1.111.

Rice, K., J. P. T. Higgins, and T. S. Lumley. 2018. A re-evaluation of fixed effect(s) meta-analysis. *Journal of the Royal Statistical Society*, A ser., 181: 205–227. https://doi.org/10.1111/rssa.12275.

Rosenthal, R. 1979. The file drawer problem and tolerance for null results. *Psychological Bulletin* 86: 638–641. https://doi.org/10.1037/0033-2909.86.3.638.

Rosenthal, R., and D. B. Rubin. 1982. Comparing effect sizes of independent studies. *Psychological Bulletin* 92: 500–504. https://doi.org/10.1037/0033-2909.92.2.500.

Rothstein, H. R., A. J. Sutton, and M. Borenstein, eds. 2005. *Publication Bias in Meta-Analysis: Prevention, Assessment and Adjustments*. Chichester, UK: Wiley. https://doi.org/10.1002/0470870168.

Schmidt, F. L., and J. E. Hunter. 2015. *Methods of Meta-Analysis: Correcting Error and Bias in Research Findings*. 3rd ed. Thousand Oaks, CA: Sage. https://doi.org/10.4135/9781483398105.

Sidik, K., and J. N. Jonkman. 2005. A note on variance estimation in random effects meta-regression. *Journal of Biopharmaceutical Statistics* 15: 823–838. https://doi.org/10.1081/BIP-200067915.

———. 2007. A comparison of heterogeneity variance estimators in combining results of studies. *Statistics in Medicine* 26: 1964–1981. https://doi.org/10.1002/sim.2688.

Simes, R. J. 1986. Publication bias: The case for an international registry of clinical trials. *Journal of Clinical Oncology* 4: 1529–1541. https://doi.org/10.1200/JCO.1986.4.10.1529.

Sin, N. L., and S. Lyubomirsky. 2009. Enhancing well-being and alleviating depressive symptoms with positive psychology interventions: A practice-friendly meta-analysis. *Journal of Clinical Psychology* 65: 467–487. https://doi.org/10.1002/jclp.20593.

Stern, J. M., and R. J. Simes. 1997. Publication bias: Evidence of delayed publication in a cohort study of clinical research projects. *BMJ* 315: 640–645. https://doi.org/10.1136/bmj.315.7109.640.

Sterne, J. A. C., B. J. Becker, and M. Egger. 2005. "The funnel plot". In *Publication Bias in Meta-Analysis: Prevention, Assessment and Adjustments*, edited by H. R. Rothstein, A. J. Sutton, and M. Borenstein, 73–98. Chichester, UK: Wiley. https://doi.org/10.1002/0470870168.ch5.

Sterne, J. A. C., D. Gavaghan, and M. Egger. 2000. Publication and related bias in meta-analysis: Power of statistical tests and prevalence in the literature. *Journal of Clinical Epidemiology* 53: 1119–1129. https://doi.org/10.1016/S0895-4356(00)00242-0.

Sterne, J. A. C., A. J. Sutton, J. P. A. Ioannidis, N. Terrin, D. R. Jones, J. Lau, J. R. Carpenter, G. Rücker, R. M. Harbord, C. H. Schmid, J. Tetzlaff, J. J. Deeks, J. L. Peters, P. Macaskill, G. Schwarzer, S. Duval, D. G. Altman, D. Moher, and J. P. T. Higgins. 2011. Recommendations for examining and interpreting funnel plot asymmetry in meta-analyses of randomised controlled trials. *BMJ* 343: d4002. https://doi.org/10.1136/bmj.d4002.

Sutton, A. J., K. R. Abrams, D. R. Jones, T. A. Sheldon, and F. Song. 2000a. *Methods for Meta-Analysis in Medical Research*. New York: Wiley.

Sutton, A. J., S. Duval, R. L. Tweedie, K. R. Abrams, and D. R. Jones. 2000b. Empirical assessment of effect of publication bias on meta-analyses. *BMJ* 320: 1574–1577. https://doi.org/10.1136/bmj.320.7249.1574.

Sutton, A. J., and J. P. T. Higgins. 2008. Recent developments in meta-analysis. *Statistics in Medicine* 27: 625–650. https://doi.org/10.1002/sim.2934.

Thompson, S. G. 1994. Systematic review: Why sources of heterogeneity in meta-analysis should be investigated. *BMJ* 309: 1351–1355. https://doi.org/10.1136/bmj.309.6965.1351.

Thompson, S. G., and J. P. T. Higgins. 2002. How should meta-regression analyses be undertaken and interpreted? *Statistics in Medicine* 21: 1559–1573. https://doi.org/10.1002/sim.1187.

Thompson, S. G., and S. J. Sharp. 1999. Explaining heterogeneity in meta-analysis: A comparison of methods. *Statistics in Medicine* 18: 2693–2708. https://doi.org/10.1002/(sici)1097-0258(19991030)18:20<2693::aid-sim235>3.0.co;2-v.

Thompson, S. G., R. M. Turner, and D. E. Warn. 2001. Multilevel models for meta-analysis, and their application to absolute risk differences. *Statistical Methods in Medical Research* 10: 375–392. https://doi.org/10.1177/096228020101000602.

Veroniki, A. A., D. Jackson, W. Viechtbauer, R. Bender, J. Bowden, G. Knapp, O. Kuss, J. P. T. Higgins, D. Langan, and G. Salanti. 2016. Methods to estimate the between-study variance and its uncertainty in meta-analysis. *Research Synthesis Methods* 7: 55–79. https://doi.org/10.1002/jrsm.1164.

Viechtbauer, W. 2005. Bias and efficiency of meta-analytic variance estimators in the random-effects model. *Journal of Educational and Behavioral Statistics* 30: 261–293. https://doi.org/10.3102/10769986030003261.

Viechtbauer, W., J. A. López-López, J. Sánchez-Meca, and F. Marín-Martínez. 2015. A comparison of procedures to test for moderators in mixed-effects meta-regression models. *Psychological Methods* 20: 360–374. https://doi.org/10.1037/met0000023.

Whitehead, A., and J. Whitehead. 1991. A general parametric approach to the meta-analysis of randomized clinical trials. *Statistics in Medicine* 10: 1665–1677. https://doi.org/10.1002/sim.4780101105.

Winfree, R., R. Aguilar, D. P. Vázquez, G. LeBuhn, and M. A. Aizen. 2009. A meta-analysis of bees' responses to anthropogenic disturbance. *Ecology* 90: 2068–2076. https://doi.org/10.1890/08-1245.1.

Woodward, R. T., and Y.-S. Wui. 2001. The economic value of wetland services: A meta-analysis. *Ecological Economics* 37: 257–270. https://doi.org/10.1016/S0921-8009(00)00276-7.

Also see

[META] **meta** — Introduction to meta

[META] **Glossary**

meta — Introduction to meta

Description Remarks and examples Acknowledgments References
Also see

Description

The `meta` command performs meta-analysis. In a nutshell, you can do the following:

1. Compute or specify effect sizes; see [META] **meta esize** and [META] **meta set**.
2. Summarize meta-analysis data; see [META] **meta summarize** and [META] **meta forestplot**.
3. Examine heterogeneity and perform meta-regression; see [META] **meta galbraithplot**, [META] **meta labbeplot**, and [META] **meta regress**.
4. Explore small-study effects and publication bias; see [META] **meta funnelplot**, [META] **meta bias**, and [META] **meta trimfill**.
5. Perform multivariate meta-regression; see [META] **meta mvregress**.
6. Perform multilevel meta-regression; see [META] **meta meregress** and [META] **meta multilevel**.

For software-free introduction to meta-analysis, see [META] **Intro**.

Declare, update, and describe `meta` data

`meta data`	Declare meta-analysis data
`meta esize`	Compute effect sizes and declare `meta` data
`meta set`	Declare `meta` data using precalculated effect sizes
`meta update`	Update current settings of `meta` data
`meta query`	Describe current settings of `meta` data
`meta clear`	Clear current settings of `meta` data

Summarize `meta` data by using a table

`meta summarize`	Summarize meta-analysis data
`meta summarize, subgroup()`	Perform subgroup meta-analysis
`meta summarize, cumulative()`	Perform cumulative meta-analysis
`meta summarize, leaveoneout`	Perform leave-one-out meta-analysis

Summarize meta data by using a forest plot

meta forestplot	Produce meta-analysis forest plots
meta forestplot, subgroup()	Produce subgroup meta-analysis forest plots
meta forestplot, cumulative()	Produce cumulative meta-analysis forest plots
meta forestplot, leaveoneout	Produce leave-one-out meta-analysis forest plots

Explore heterogeneity and perform meta-regression

meta galbraithplot	Produce Galbraith plots
meta labbeplot	Produce L'Abbé plots for two-group comparison of binary outcomes
meta regress	Fit meta-regression
estat bubbleplot	Produce bubble plots after meta-regression

Explore and address small-study effects (funnel-plot asymmetry, publication bias)

meta funnelplot	Produce funnel plots
meta funnelplot, contours()	Produce contour-enhanced funnel plots
meta bias	Test for small-study effects or funnel-plot asymmetry
meta trimfill	Perform trim-and-fill analysis of publication bias

Perform multivariate meta-regression

meta mvregress	Fit multivariate meta-regression
estat heterogeneity (mv)	Assess heterogeneity in multivariate meta-regression

meta mvregress does not require your dataset to be meta set.

Perform multilevel meta-regression

meta meregress	Fit multilevel meta-regression
meta multilevel	Fit random-intercepts multilevel meta-regression
estat heterogeneity (me)	Assess heterogeneity in multilevel meta-regression

meta meregress and meta multilevel do not require your dataset to be meta set.

Remarks and examples

This entry describes Stata's suite of commands, `meta`, for performing meta-analysis. For a software-free introduction to meta-analysis, see [META] **Intro**.

Remarks are presented under the following headings:

Introduction to meta-analysis using Stata
Example datasets
  Effects of teacher expectancy on pupil IQ (pupiliq.dta)
  Effect of streptokinase after a myocardial infarction (strepto.dta)
  Efficacy of BCG vaccine against tuberculosis (bcg.dta)
  Effectiveness of nonsteroidal anti-inflammatory drugs (nsaids.dta)
  Treatment of moderate periodontal disease (periodontal.dta)
Tour of meta-analysis commands
  Prepare your data for meta-analysis in Stata
  Basic meta-analysis summary
  Subgroup meta-analysis
  Cumulative meta-analysis
  Heterogeneity: Galbraith plot, meta-regression, and bubble plot
  Funnel plots for exploring small-study effects
  Testing for small-study effects
  Trim-and-fill analysis for addressing publication bias
  Multivariate meta-regression
  Multilevel meta-regression

Introduction to meta-analysis using Stata

Stata's `meta` command offers full support for meta-analysis from computing various effect sizes and producing basic meta-analytic summary and forest plots to accounting for between-study heterogeneity and potential publication bias. Random-effects, common-effect, and fixed-effects meta-analyses are supported.

Standard effect sizes, such as the log odds-ratio for a two-group comparison of binary outcomes, Hedges's g for a two-group comparison of continuous outcomes, the Fisher's z-transformed correlation for correlation data, or the Freeman–Tukey-transformed proportion for estimating a single proportion (prevalence), may be computed using the `meta esize` command; see [META] **meta esize**. Generic (pre-calculated) effect sizes may be specified by using the `meta set` command; see [META] **meta set**.

`meta esize` and `meta set` are part of the meta-analysis declaration step, which is the first step of meta-analysis in Stata. During this step, you specify the main information about your meta-analysis such as the study-specific effect sizes and their corresponding standard errors and the meta-analysis model and method. This information is then automatically used by all subsequent `meta` commands for the duration of your meta-analysis session. You can use `meta update` to easily update some of the specified information during the session; see [META] **meta update**. And you can use `meta query` to remind yourself about the current meta settings at any point of your meta-analysis; see [META] **meta update**. For more information about the declaration step, see [META] **meta data**. Also see *Prepare your data for meta-analysis in Stata*.

Random-effects, common-effect, and fixed-effects meta-analysis models are supported. You can specify them during the declaration step and use the same model throughout your meta-analysis or you can specify a different model temporarily with any of the `meta` commands. You can also switch to a different model for the rest of your meta-analysis by using `meta update`. See *Declaring a meta-analysis model* in [META] **meta data** for details.

Traditionally, meta-analysis literature and software used the term "fixed-effect model" (notice singular effect) to refer to the model that assumes a common effect for all studies. To avoid potential confusion with the term "fixed-effects model" (notice plural effects), which is commonly used in various disciplines to refer to the model whose effects vary from one group to another, we adopted the terminology from Rice, Higgins, and Lumley (2018) of the "common-effect model". This terminology is also reflected in the option names for specifying the corresponding models with `meta` commands: `common` specifies a common-effect model and `fixed` specifies a fixed-effects model. (Similarly, `random` specifies a random-effects model.) Although the overall effect-size estimates from the common-effect and fixed-effects models are computationally identical, their interpretation is different. We provide the two options to emphasize this difference and to encourage proper interpretation of the final results given the specified model. See common-effect versus fixed-effects models in [META] **meta data** and *Meta-analysis models* in [META] **Intro** for more information.

Depending on the chosen meta-analysis model, various estimation methods are available: inverse-variance and Mantel–Haenszel for the common-effect and fixed-effects models and seven different estimators for the between-study variance parameter for the random-effects model. See *Declaring a meta-analysis estimation method* in [META] **meta data**.

Also see *Default meta-analysis model and method* in [META] **meta data** for the default model and method used by the `meta` commands.

Results of a basic meta-analysis can be summarized numerically in a table by using `meta summarize` (see [META] **meta summarize**) or graphically by using forest plots; see [META] **meta forestplot**. See *Basic meta-analysis summary*.

To evaluate the trends in the estimates of the overall effect sizes, you can use the `cumulative()` option with `meta summarize` or `meta forestplot` to perform cumulative meta-analysis. See *Cumulative meta-analysis*.

In the presence of subgroup heterogeneity, you can use the `subgroup()` option with `meta summarize` or `meta forestplot` to perform single or multiple subgroup analyses. See *Subgroup meta-analysis*.

Heterogeneity can also be explored by fitting meta-regression using the `meta regress` command; see [META] **meta regress**. After meta-regression, you can produce bubble plots (see [META] **estat bubbleplot**) and perform other postestimation analysis (see [META] **meta regress postestimation**). Also see *Heterogeneity: Galbraith plot, meta-regression, and bubble plot*.

In addition to forest plots, you can also visually explore heterogeneity using `meta galbraithplot`, which works with any type of data (see [META] **meta galbraithplot**), and `meta labbeplot`, which works with a two-group comparison of binary outcomes (see [META] **meta labbeplot**).

Publication bias, or more accurately, small-study effects or funnel-plot asymmetry, may be explored graphically via standard or contour-enhanced funnel plots (see [META] **meta funnelplot**). Regression-based and other tests for detecting small-study effects are available with the `meta bias` command; see [META] **meta bias**. The trim-and-fill method for assessing the potential impact of publication bias on the meta-analysis results is implemented in the `meta trimfill` command; see [META] **meta trimfill**. See *Funnel plots for exploring small-study effects*, *Testing for small-study effects*, and *Trim-and-fill analysis for addressing publication bias*.

Multivariate meta-regression can be fit via `meta mvregress` (see [META] **meta mvregress**). After multivariate meta-regression, you can explore heterogeneity using `estat heterogeneity` (see [META] **estat heterogeneity (mv)**) and conduct other postestimation analysis (see [META] **meta mvregress postestimation**).

Multilevel meta-regression can be fit via `meta meregress` (see [META] **meta meregress**) or `meta multilevel` ([META] **meta multilevel**). After multilevel meta-regression, you can explore multilevel heterogeneity using `estat heterogeneity` (see [META] **estat heterogeneity (me)**) and conduct other postestimation analysis (see [META] **meta me postestimation**).

Example datasets

We present several datasets that we will use throughout the documentation to demonstrate the `meta` suite. Feel free to skip over this section to *Tour of meta-analysis commands* and come back to it later for specific examples.

Example datasets are presented under the following headings:

Effects of teacher expectancy on pupil IQ (pupiliq.dta)
Effect of streptokinase after a myocardial infarction (strepto.dta)
Efficacy of BCG vaccine against tuberculosis (bcg.dta)
Effectiveness of nonsteroidal anti-inflammatory drugs (nsaids.dta)
Treatment of moderate periodontal disease (periodontal.dta)

Effects of teacher expectancy on pupil IQ (pupiliq.dta)

This example describes a well-known study of Rosenthal and Jacobson (1968) that found the so-called Pygmalion effect, in which expectations of teachers affected outcomes of their students. A group of students was tested and then divided randomly into experimentals and controls. The division may have been random, but the teachers were told that the students identified as experimentals were likely to show dramatic intellectual growth. A few months later, a test was administered again to the entire group of students. The experimentals outperformed the controls.

Subsequent researchers attempted to replicate the results, but many did not find the hypothesized effect.

Raudenbush (1984) did a meta-analysis of 19 studies and hypothesized that the Pygmalion effect might be mitigated by how long the teachers had worked with the students before being told about the nonexistent higher expectations for the randomly selected subsample of students. We explore this hypothesis in *Subgroup meta-analysis*.

The data are saved in pupiliq.dta. Below, we describe some of the variables that will be used in later analyses.

```
. use https://www.stata-press.com/data/r18/pupiliq
(Effects of teacher expectancy on pupil IQ)
. describe
Contains data from https://www.stata-press.com/data/r18/pupiliq.dta
 Observations:            19                  Effects of teacher expectancy
                                                on pupil IQ
    Variables:            14                  24 Apr 2022 08:28
                                              (_dta has notes)
------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
study           byte    %9.0g                 Study number
author          str20   %20s                  Author
year            int     %9.0g                 Publication year
nexper          int     %9.0g                 Sample size in experimental group
ncontrol        int     %9.0g                 Sample size in control group
stdmdiff        double  %9.0g                 Standardized difference in means
weeks           byte    %9.0g                 Weeks of prior teacher-student
                                                contact
catweek         byte    %9.0g      catwk      Weeks of prior contact
                                                (categorical)
week1           byte    %9.0g      catweek1   Prior teacher-student contact > 1
                                                week
se              double  %10.0g                Standard error of stdmdiff
se_c            float   %9.0g                 se from Pubbias book, p.322
setting         byte    %8.0g      testtype   Test setting
tester          byte    %8.0g      tester     Tester (blind or aware)
studylbl        str26   %26s                  Study label
------------------------------------------------------------------------------
Sorted by:
```

Variables stdmdiff and se contain the effect sizes (standardized mean differences between the experimental and control groups) and their standard errors, respectively. Variable weeks records the number of weeks of prior contact between the teacher and the students. Its dichotomized version, week1, records whether the teachers spent more than one week with the students (high-contact group, week1 = 1) or one week and less (low-contact group, week1 = 0) prior to the experiment.

We perform basic meta-analysis summary of this dataset in *Basic meta-analysis summary* and explore the between-study heterogeneity of the results with respect to the amount of the teacher–student contact in *Subgroup meta-analysis*.

This dataset is also used in *Examples of using meta summarize* of [META] **meta summarize**, example 5 of [META] **meta forestplot**, example 8 of [META] **meta funnelplot**, and *Examples of using meta bias* of [META] **meta bias**.

See example 1 for the declaration of the pupiliq.dta. You can also use its predeclared version, pupiliqset.dta.

Effect of streptokinase after a myocardial infarction (strepto.dta)

Streptokinase is a medication used to break down clots. In the case of myocardial infarction (heart attack), breaking down clots reduces damage to the heart muscle.

Lau et al. (1992) conducted a meta-analysis of 33 studies performed between 1959 and 1988. These studies were of heart attack patients who were randomly treated with streptokinase or a placebo. Lau et al. (1992) introduced cumulative meta-analysis to investigate the time when the effect of streptokinase became statistically significant. Studies were ordered by time, and as each was added to the analysis, standard meta-analysis was performed. See *Cumulative meta-analysis* for details.

The data are saved in `strepto.dta`.

```
. use https://www.stata-press.com/data/r18/strepto
(Effect of streptokinase after a myocardial infarction)

. describe

Contains data from https://www.stata-press.com/data/r18/strepto.dta
 Observations:            33                  Effect of streptokinase after a
                                                myocardial infarction
    Variables:             7                  14 May 2022 18:24
                                              (_dta has notes)
```

Variable name	Storage type	Display format	Value label	Variable label
study	str12	%12s		Study name
year	int	%10.0g		Publication year
ndeadt	int	%10.0g		Number of deaths in treatment group
nsurvt	int	%9.0g		Number of survivors in treatment group
ndeadc	int	%10.0g		Number of deaths in control group
nsurvc	int	%9.0g		Number of survivors in control group
studyplus	str13	%13s		Study label for cumulative MA

```
Sorted by:
```

The outcome of interest was death from myocardial infarction. Variables `ndeadt` and `nsurvt` contain the numbers of deaths and survivals, respectively, in the treatment group and `ndeadc` and `nsurvc` contain those in the control (placebo) group.

See example 5 for the declaration of the `strepto.dta`. You can also use its predeclared version, `streptoset.dta`.

Efficacy of BCG vaccine against tuberculosis (bcg.dta)

BCG vaccine is a vaccine used to prevent tuberculosis (TB). The vaccine is used worldwide. Efficacy has been reported to vary. Colditz et al. (1994) performed meta-analysis on the efficacy using 13 studies—all randomized trials—published between 1948 and 1980. The dataset, shown below, has been studied by, among others, Berkey et al. (1995), who hypothesized that the latitude of the study location might explain the variations in efficacy. We explore this via meta-regression in *Heterogeneity: Galbraith plot, meta-regression, and bubble plot*.

The data are saved in bcg.dta. Below, we describe some of the variables we will use in future analyses.

```
. use https://www.stata-press.com/data/r18/bcg
(Efficacy of BCG vaccine against tuberculosis)
. describe
Contains data from https://www.stata-press.com/data/r18/bcg.dta
 Observations:            13                  Efficacy of BCG vaccine against
                                                tuberculosis
    Variables:            11                  1 May 2022 14:40
                                              (_dta has notes)
------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
trial           byte    %9.0g                 Trial number
trialloc        str14   %14s                  Trial location
author          str21   %21s                  Author
year            int     %9.0g                 Publication year
npost           int     %9.0g                 Number of TB positive cases in
                                                treated group
nnegt           long    %9.0g                 Number of TB negative cases in
                                                treated group
nposc           int     %9.0g                 Number of TB positive cases in
                                                control group
nnegc           long    %9.0g                 Number of TB negative cases in
                                                control group
latitude        byte    %9.0g                 Absolute latitude of the study
                                                location (in degrees)
alloc           byte    %10.0g      alloc     Method of treatment allocation
studylbl        str27   %27s                  Study label
------------------------------------------------------------------------------
Sorted by: trial
```

Variables npost and nnegt contain the numbers of positive and negative TB cases, respectively, in the treatment group (vaccinated group) and nposc and nnegc contain those in the control group. Variable latitude records the latitude of the study location, which is a potential moderator for the vaccine efficacy. Studies are identified by studylbl, which records the names of the authors and the year of the publication for each study.

This dataset is also used in example 3 of [META] **meta data**, *Examples of using meta forestplot* of [META] **meta forestplot**, example 1 of [META] **meta galbraithplot**, example 1 of [META] **meta labbeplot**, *Examples of using meta regress* of [META] **meta regress**, *Remarks and examples* of [META] **meta regress postestimation**, and *Examples of using estat bubbleplot* of [META] **estat bubbleplot**.

See example 7 for the declaration of the bcg.dta. You can also use its predeclared version, bcgset.dta.

Effectiveness of nonsteroidal anti-inflammatory drugs (nsaids.dta)

Strains and sprains cause pain, and nonsteroidal anti-inflammatory drugs (NSAIDS) are used to treat it. How well do they work? People who study such things define success as a 50-plus percent reduction in pain. Moore et al. (1998) performed meta-analysis of 37 randomized trials that looked into successful pain reduction via NSAIDS. Following their lead, we will explore publication bias or, more generally, small-study effects in these data. See *Funnel plots for exploring small-study effects*, *Testing for small-study effects*, and *Trim-and-fill analysis for addressing publication bias*.

The data are saved in nsaids.dta.

```
. use https://www.stata-press.com/data/r18/nsaids
(Effectiveness of nonsteroidal anti-inflammatory drugs)
. describe
Contains data from https://www.stata-press.com/data/r18/nsaids.dta
 Observations:            37                  Effectiveness of nonsteroidal
                                                anti-inflammatory drugs
    Variables:             5                  24 Apr 2022 17:09
                                              (_dta has notes)
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
study           byte    %8.0g                 Study ID
nstreat         byte    %8.0g                 Number of successes in the
                                                treatment arm
nftreat         byte    %9.0g                 Number of failures in the
                                                treatment arm
nscontrol       byte    %8.0g                 Number of successes in the
                                                control arm
nfcontrol       byte    %9.0g                 Number of failures in the control
                                                arm
-------------------------------------------------------------------------------
Sorted by:
```

Variables nstreat and nftreat contain the numbers of successes and failures, respectively, in the experimental group and nscontrol and nfcontrol contain those in the control group.

This dataset is also used in *Examples of using meta funnelplot* of [META] **meta funnelplot** and example 3 of [META] **meta bias**.

See example 11 for the declaration of the nsaids.dta. You can also use its predeclared version, nsaidsset.dta.

Treatment of moderate periodontal disease (periodontal.dta)

Periodontal disease is the inflammation of the gum that may destroy the bone supporting the teeth. Antczak-Bouckoms et al. (1993) investigated five randomized controlled trials that explored the impact of two procedures, surgical and nonsurgical, on treating periodontal disease. This dataset was also analyzed by Berkey et al. (1998). Subjects' mouths were split into sections. These sections were randomly allocated to the two treatment procedures. For each patient, at least one section was treated surgically and at least one other section was treated nonsurgically. Two outcomes (effect sizes) of interest were mean improvements from baseline (pretreatment) in probing depth (y1) and attachment level (y2) around the teeth.

The data are saved in periodontal.dta.

```
. use https://www.stata-press.com/data/r18/periodontal
(Treatment of moderate periodontal disease)
. describe
Contains data from https://www.stata-press.com/data/r18/periodontal.dta
 Observations:             5                  Treatment of moderate
                                                periodontal disease
    Variables:             9                  13 Jan 2023 18:11
                                              (_dta has notes)
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
trial           str23   %23s                  Trial label
pubyear         byte    %9.0g                 Publication year centered at 1983
y1              float   %6.2f                 Mean improvement in probing depth
                                                (mm)
y2              float   %6.2f                 Mean improvement in attachment
                                                level (mm)
v11             float   %6.4f                 Variance of y1
v12             float   %6.4f                 Covariance of y1 and y2
v22             float   %6.4f                 Variance of y2
s1              double  %10.0g                Standard error of y1
s2              double  %10.0g                Standard error of y2
-------------------------------------------------------------------------------
Sorted by:
```

Other variables of interest that will be used in example 15 are the year of publication (pubyear) and three variables defining the within-study covariance matrix for each study: v11, v12, and v22.

This dataset is also used in *Examples of using meta mvregress* of [META] **meta mvregress**.

Tour of meta-analysis commands

In this section, we provide a tour of Stata's meta-analysis (meta) commands with applications to several real-world datasets. We demonstrate the basic meta-analysis summary and a forest plot and explore heterogeneity via subgroup analysis using the pupil IQ dataset. We then demonstrate cumulative meta-analysis using the streptokinase dataset. We continue with more heterogeneity analyses of the BCG dataset. Finally, we explore and address publication bias for the NSAIDS dataset.

Examples are presented under the following headings:

Prepare your data for meta-analysis in Stata
Basic meta-analysis summary
Subgroup meta-analysis
Cumulative meta-analysis
Heterogeneity: Galbraith plot, meta-regression, and bubble plot
Funnel plots for exploring small-study effects
Testing for small-study effects
Trim-and-fill analysis for addressing publication bias
Multivariate meta-regression
Multilevel meta-regression

Prepare your data for meta-analysis in Stata

The first step of meta-analysis in Stata is to declare your data as meta data. During this step, we specify the main information needed for meta-analysis such as effect sizes and their standard errors. We declare this information once by using either meta set or meta esize, and it is then used by all meta

commands. If needed, we can update our initial settings throughout the meta-analysis session by using `meta update`. The declaration step helps minimize potential mistakes and typing; see [META] **meta data** for details.

▷ Example 1: Set up your data for meta-analysis in Stata

Consider the pupil IQ dataset described in *Effects of teacher expectancy on pupil IQ (pupiliq.dta)*.

```
. use https://www.stata-press.com/data/r18/pupiliq
(Effects of teacher expectancy on pupil IQ)
. describe studylbl stdmdiff se week1
```

Variable name	Storage type	Display format	Value label	Variable label
studylbl	str26	%26s		Study label
stdmdiff	double	%9.0g		Standardized difference in means
se	double	%10.0g		Standard error of stdmdiff
week1	byte	%9.0g	catweek1	Prior teacher-student contact > 1 week

First, we prepare our data for use with `meta` commands. The dataset contains precomputed effect sizes, standardized mean differences stored in variable `stdmdiff`, and their standard errors stored in variable `se`. We will use `meta set` to declare these data. (If we needed to compute the individual effect sizes and their standard errors from the available summary data, we would have used [META] **meta esize**.)

We specify the effect sizes `stdmdiff` and their standard errors `se` with `meta set`. We also specify the variable that contains the study labels in the `studylabel()` option and the effect-size label in the `eslabel()` option. These are optional but useful for displaying the study and effect-size labels instead of generic study numbers and the generic label `Effect size`.

```
. meta set stdmdiff se, studylabel(studylbl) eslabel(Std. mean diff.)
Meta-analysis setting information
 Study information
    No. of studies: 19
       Study label: studylbl
        Study size: N/A
       Effect size
             Type: <generic>
            Label: Std. mean diff.
         Variable: stdmdiff
         Precision
         Std. err.: se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

The header reports that there are $K = 19$ studies in the meta-analysis and which variables contain the study labels, the effect sizes, and the standard errors. The output also shows that we will be using the random-effects model with the REML estimation method for our meta-analysis. This can be changed by specifying options with either `meta set` or the `meta` command of interest; see *Declaring a meta-analysis model* in [META] **meta data**.

`meta set` creates some system variables beginning with `_meta_` and stores some data characteristics. For example, the system variables `_meta_cil` and `_meta_ciu` store the lower and upper limits of the CIs for the effect sizes. See *System variables* in [META] **meta data** for details.

See [META] **meta set** for more information about the command. o

◁

Basic meta-analysis summary

In this section, we focus on basic meta-analysis summary by using [META] **meta summarize** and [META] **meta forestplot**. See *Introduction* of [META] **meta summarize** and *Overview* of [META] **meta forestplot** for an overview of the meta-analysis summary and forest plots.

▷ Example 2: Meta-analysis summary

Continuing with example 1, we use `meta summarize` to combine the studies and estimate the overall effect size.

```
. meta summarize
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
        Study label: studylbl

Meta-analysis summary                           Number of studies =     19
Random-effects model                            Heterogeneity:
Method: REML                                                tau2 =  0.0188
                                                         I2 (%) =   41.84
                                                             H2 =    1.72

                Effect size: Std. mean diff.
------------------------------------------------------------------------------
                       Study |     Effect size    [95% conf. interval]  % weight
-----------------------------+------------------------------------------------
    Rosenthal et al., 1974   |          0.030      -0.215       0.275      7.74
         Conn et al., 1968   |          0.120      -0.168       0.408      6.60
         Jose & Cody, 1971   |         -0.140      -0.467       0.187      5.71
  Pellegrini & Hicks, 1972   |          1.180       0.449       1.911      1.69
  Pellegrini & Hicks, 1972   |          0.260      -0.463       0.983      1.72
   Evans & Rosenthal, 1969   |         -0.060      -0.262       0.142      9.06
      Fielder et al., 1971   |         -0.020      -0.222       0.182      9.06
            Claiborn, 1969   |         -0.320      -0.751       0.111      3.97
              Kester, 1969   |          0.270      -0.051       0.591      5.84
             Maxwell, 1970   |          0.800       0.308       1.292      3.26
              Carter, 1970   |          0.540      -0.052       1.132      2.42
             Flowers, 1966   |          0.180      -0.257       0.617      3.89
             Keshock, 1970   |         -0.020      -0.586       0.546      2.61
           Henrikson, 1970   |          0.230      -0.338       0.798      2.59
               Fine, 1972    |         -0.180      -0.492       0.132      6.05
             Grieger, 1970   |         -0.060      -0.387       0.267      5.71
Rosenthal & Jacobson, 1968   |          0.300       0.028       0.572      6.99
  Fleming & Anttonen, 1971   |          0.070      -0.114       0.254      9.64
           Ginsburg, 1970    |         -0.070      -0.411       0.271      5.43
-----------------------------+------------------------------------------------
                       theta |          0.084      -0.018       0.185
------------------------------------------------------------------------------
Test of theta = 0: z = 1.62                     Prob > |z| = 0.1052
Test of homogeneity: Q = chi2(18) = 35.83         Prob > Q = 0.0074
```

The output from the standard meta-analysis summary includes heterogeneity statistics, the individual and overall effect sizes, and other information. The estimate of the overall effect size θ is reported at the bottom of the table and labeled as `theta`. It is computed as the weighted average of study-specific effect sizes (standardized mean differences in our example). For these data, the overall estimate is 0.084 with a 95% CI of $[-0.018, 0.185]$. The significance test of H_0: $\theta = 0$ is reported below the table and has a p-value of 0.1052, which suggests that the overall effect size is not statistically significantly different from zero.

We should be careful with our inferential conclusions about θ because of the presence of between-study heterogeneity, as indicated, for instance, by the homogeneity test of H_0: $\theta_1 = \theta_2 = \cdots = \theta_{19} = \theta$ reported following the significance test. Its Q test statistic is 35.83 with a p-value of 0.0074, from which we can infer that there is significant heterogeneity between the individual studies.

The presence of heterogeneity among studies can be inferred also from the heterogeneity statistics reported in the header. For instance, $I^2 = 41.84$ indicates that about 42% of the variability in the effect-size estimates is due to the differences between studies. The between-study heterogeneity must be addressed before final meta-analytic conclusions; see *Subgroup meta-analysis*.

The table also reports the study-specific effect-sizes and their corresponding 95% CIs, but this information can be suppressed, if desired, by specifying the `nostudies` option.

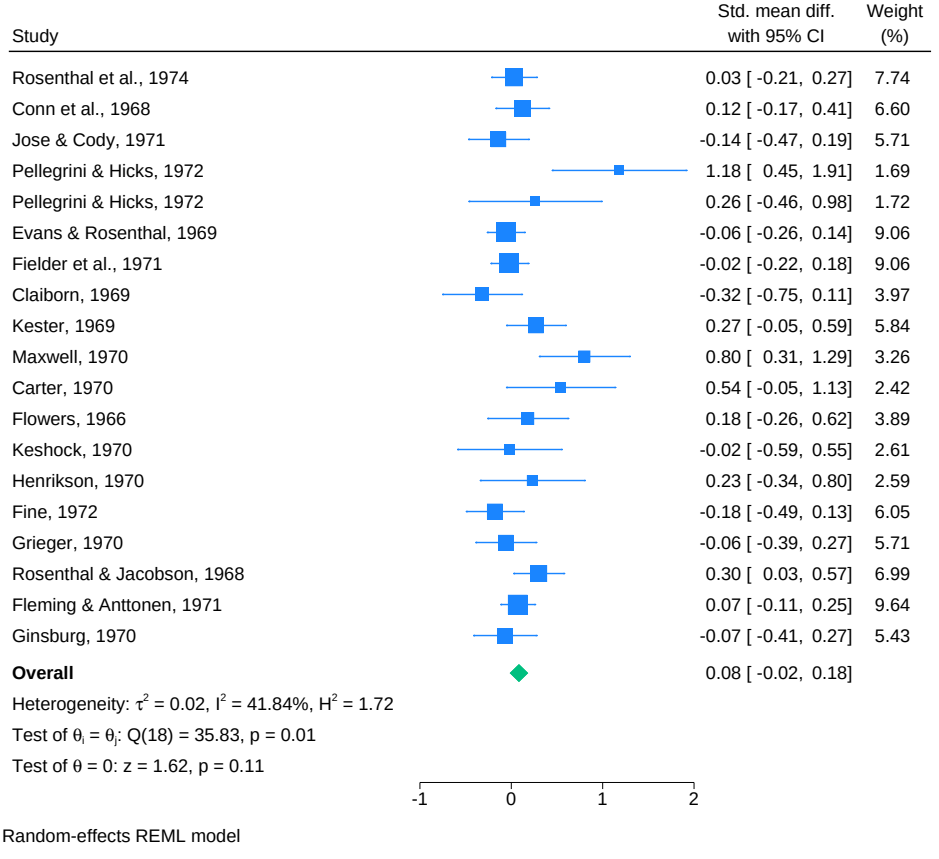
See [META] **meta summarize** for details.

◁

▷ Example 3: Forest plot

The results of meta-analysis are commonly displayed graphically using a forest plot. Continuing with example 2, we can use `meta forestplot` to produce a meta-analysis forest plot for the pupil IQ data.

```
. meta forestplot
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
        Study label: studylbl
```



We obtain the same meta-analysis summary as with `meta summarize` in example 2, but it is now displayed on a graph. In addition to the estimated values, the effect sizes are displayed graphically as blue squares centered at their estimates with areas proportional to the study weights and with horizontal lines or whiskers that represent the length of the corresponding CIs. The overall effect size is displayed as a green diamond with its width corresponding to the respective CI. (Notice that only the width and not the height of the diamond is relevant for the overall effect size.)

A forest plot provides an easy way to visually explore the agreement between the study-specific effect sizes and how close they are to the overall effect size. We can also spot the studies with large weights more easily by simply looking at the studies with large squares. In our example, the presence of between-study heterogeneity is evident—there are several studies whose effect-size estimates are very different from the overall estimate, and there are studies whose CIs do not even overlap.

See [META] **meta forestplot** for details.

Subgroup meta-analysis

In example 2 and example 3, we established the presence of between-study heterogeneity in the pupil IQ dataset. Sometimes, the differences between studies may be explained by study-level covariates available in the data. When these covariates are categorical, we can perform meta-analysis separately for each category, which is known as subgroup meta-analysis; see *Subgroup meta-analysis* of [META] **Intro**.

▷ Example 4: Subgroup meta-analysis

Raudenbush (1984) suspected that the amount of time the teachers spent with students before the experiment could impact their susceptibility to researchers' test results about children's intellectual abilities. If so, we would expect the effect sizes to be negatively associated with the amount of contact.

Continuing with example 2, we see that the dataset contains a binary variable `week1` that records whether the teachers spend more than one week with children (high-contact group) or one week and less (low-contact group). Let's perform meta-analysis separately for each group. Under Raudenbush's hypothesis, we should expect to see larger effect sizes in the low-contact group and smaller effect sizes in the high-contact group.

We use the `subgroup()` option with `meta summarize` to perform a separate analysis for each group of `week1`.

```
. meta summarize, subgroup(week1)

  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
        Study label: studylbl

Subgroup meta-analysis summary                    Number of studies =     19
Random-effects model
Method: REML
Group: week1
```

Effect size: Std. mean diff.

Study	Effect size	[95% conf.	interval]	% weight
Group: <= 1 week				
Pellegrini & Hicks, 1972	1.180	0.449	1.911	1.69
Pellegrini & Hicks, 1972	0.260	-0.463	0.983	1.72
Kester, 1969	0.270	-0.051	0.591	5.84
Maxwell, 1970	0.800	0.308	1.292	3.26
Carter, 1970	0.540	-0.052	1.132	2.42
Flowers, 1966	0.180	-0.257	0.617	3.89
Keshock, 1970	-0.020	-0.586	0.546	2.61
Rosenthal & Jacobson, 1968	0.300	0.028	0.572	6.99
theta	0.373	0.189	0.557	
Group: > 1 week				
Rosenthal et al., 1974	0.030	-0.215	0.275	7.74
Conn et al., 1968	0.120	-0.168	0.408	6.60
Jose & Cody, 1971	-0.140	-0.467	0.187	5.71
Evans & Rosenthal, 1969	-0.060	-0.262	0.142	9.06
Fielder et al., 1971	-0.020	-0.222	0.182	9.06
Claiborn, 1969	-0.320	-0.751	0.111	3.97
Henrikson, 1970	0.230	-0.338	0.798	2.59
Fine, 1972	-0.180	-0.492	0.132	6.05
Grieger, 1970	-0.060	-0.387	0.267	5.71
Fleming & Anttonen, 1971	0.070	-0.114	0.254	9.64
Ginsburg, 1970	-0.070	-0.411	0.271	5.43
theta	-0.021	-0.102	0.059	
Overall				
theta	0.084	-0.018	0.185	

Heterogeneity summary

Group	df	Q	P > Q	tau2	% I2	H2
<= 1 week	7	11.20	0.130	0.015	22.40	1.29
> 1 week	10	6.40	0.780	0.000	0.00	1.00
Overall	18	35.83	0.007	0.019	41.84	1.72

Test of group differences: Q_b = chi2(1) = 14.77 Prob > Q_b = 0.000

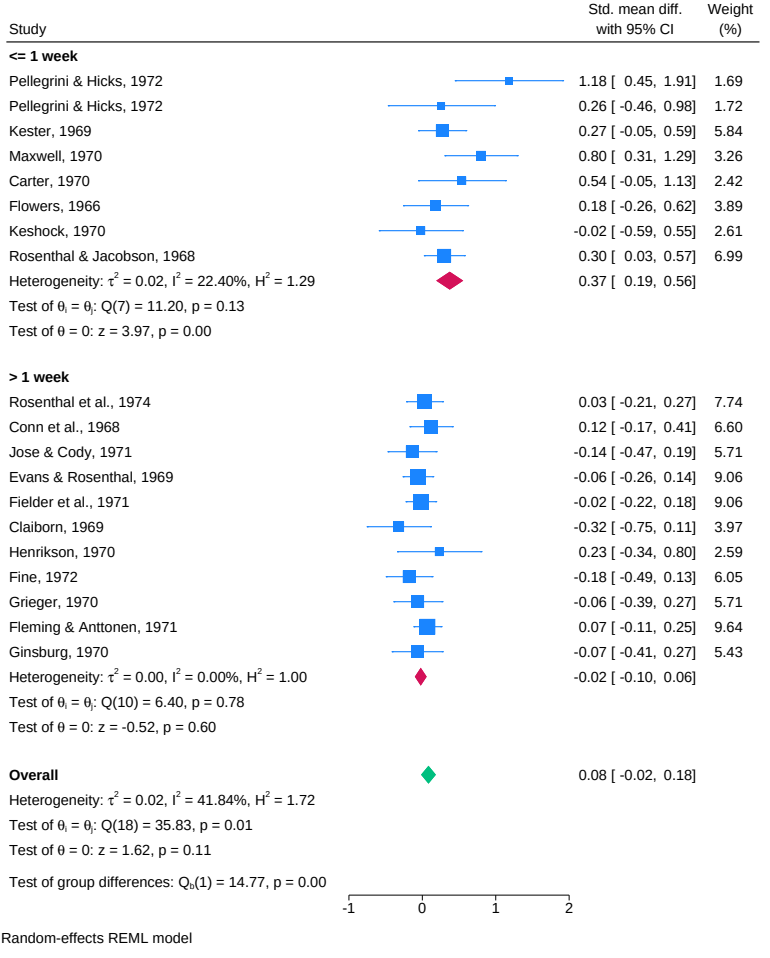
Indeed, if we look at the overall effect-size estimates for each group, the low-contact group has a larger estimate of 0.373 with a 95% CI of $[0.189, 0.557]$, which suggests a statistically significant effect in this group, whereas the high-contact group has a smaller estimate of -0.021 with a 95% CI of $[-0.102, 0.059]$, which suggests that the effect in this group is not different from 0 at a 5% significance level. Clearly, the amount of teacher contact with students has an impact on the meta-analysis results.

If we look at the heterogeneity summary reported following the main table, we will see that heterogeneity is reduced within each group. It is essentially nonexistent in the high-contact group and is much smaller (for instance, $I^2 = 22\%$ versus the earlier $I^2 = 42\%$) in the low-contact group.

The test of group differences (with $Q_b = 14.77$ and the corresponding p-value of 0.000) reported at the bottom of the output also indicates that the group-specific overall effect sizes are statistically different.

We can also present the results of our subgroup analysis graphically by using the `subgroup()` option with `meta forest`:

```
. meta forestplot, subgroup(week1)
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
        Study label: studylbl
```



It appears that stratifying our meta-analysis on the amount of prior contact between students and teachers explained most of the variability in the magnitudes of the effect sizes, at least in the high-contact group. ◁

When interpreting results from subgroup analysis, we should be mindful that the results are based on fewer studies and thus may not be as precise, in general.

See [META] **meta summarize** and [META] **meta forestplot**.

Cumulative meta-analysis

Cumulative meta-analysis performs multiple meta-analyses by accumulating studies one at a time after ordering them with respect to a variable of interest. This analysis is useful to monitor the trend in the estimates of the overall effect sizes with respect to some factor. For instance, it may be used to detect the time when the effect size of interest became significant.

▷ Example 5: Computing log odds-ratios using meta esize

Consider the streptokinase dataset described in *Effect of streptokinase after a myocardial infarction (strepto.dta)*.

```
. use https://www.stata-press.com/data/r18/strepto, clear
(Effect of streptokinase after a myocardial infarction)
. describe
Contains data from https://www.stata-press.com/data/r18/strepto.dta
 Observations:            33                  Effect of streptokinase after a
                                                myocardial infarction
    Variables:             7                  14 May 2022 18:24
                                              (_dta has notes)
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
study           str12   %12s                  Study name
year            int     %10.0g                Publication year
ndeadt          int     %10.0g                Number of deaths in treatment
                                                group
nsurvt          int     %9.0g                 Number of survivors in treatment
                                                group
ndeadc          int     %10.0g                Number of deaths in control group
nsurvc          int     %9.0g                 Number of survivors in control
                                                group
studyplus       str13   %13s                  Study label for cumulative MA
-------------------------------------------------------------------------------
Sorted by:
```

As in example 1, first we prepare our data for use with `meta` commands. Our dataset contains the summary data that represent the study-specific 2×2 tables. The variables `ndeadt`, `nsurvt`, `ndeadc`, and `nsurvc` record the numbers of deaths and survivors in the treatment and control groups.

Lau et al. (1992) considered an odds ratio as the effect size of interest for these data. For odds ratios, meta-analysis is performed in the log metric. We can use `meta esize` to compute study-specific log odds-ratios and their corresponding standard errors and declare them for the subsequent meta-analysis. To compute log odds-ratios, we specify the four variables containing table cell counts with `meta esize`. As with `meta set` in example 1, we specify the study labels in the `studylabel()` option with `meta esize`.

```
. meta esize ndeadt nsurvt ndeadc nsurvc, studylabel(studyplus) common
Meta-analysis setting information
 Study information
    No. of studies: 33
       Study label: studyplus
        Study size: _meta_studysize
      Summary data: ndeadt nsurvt ndeadc nsurvc
       Effect size
             Type: lnoratio
            Label: Log odds-ratio
         Variable: _meta_es
   Zero-cells adj.: 0.5, only0
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Common effect
           Method: Mantel-Haenszel
```

`meta esize` reports that there are 33 trials and that the computed effect size is log odds-ratio. This is the default effect size with a two-group comparison of binary outcomes. You can specify other effect sizes in the `esize()` option, which include a log risk-ratio, risk difference, and log Peto's odds-ratio. (After the declaration, you can use `meta update` to change the effect size more easily without having to respecify your summary data variables; see [META] **meta update**.)

Lau et al. (1992) used a common-effect model with the Mantel–Haenszel method to perform their cumulative meta-analysis. We will follow their approach. Thus, we also specified the `common` option with `meta esize`. The command reported that the assumed meta-analysis model is a common-effect model. The Mantel–Haenszel estimation method is the default method for log odds-ratios under a common-effect model.

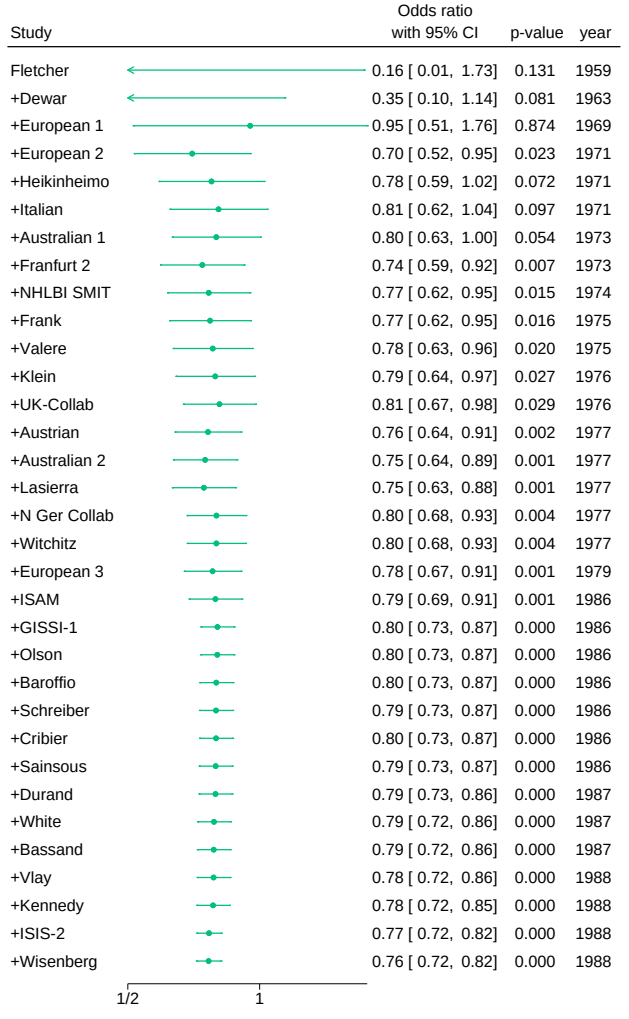
◁

▷ Example 6: Cumulative meta-analysis

After the data declaration in example 5, we are ready to perform the cumulative meta-analysis. Lau et al. (1992) used cumulative meta-analysis to investigate the trends in the effect of the streptokinase drug used to prevent death after a myocardial infarction. We replicate their analysis below by producing a cumulative meta-analysis plot over the years for these data. Also see Borenstein, Hedges, Higgins, and Rothstein (2009) for the analysis of these data.

We use the `meta forestplot` command with the `cumulative()` option. We use the `or` option to display odds ratios instead of the default log odds-ratios. To match figure 1 in Lau et al. (1992) more closely, we also specify the `crop(0.5 .)` option to crop the lower CI limits and log odds-ratios estimates that are smaller than 0.5.

```
. meta forestplot, cumulative(year) or crop(0.5 .)
  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
        Study label: studyplus
```



The cumulative meta-analysis forest plot displays the overall effect-size estimates and the corresponding CIs computed for the first study, for the first two studies, for the first three studies, and so on. The point estimates are represented by green circles, and the CIs are represented by the CI lines. The change in style and color of the plotted markers emphasizes that the (cumulative) overall effect sizes and not the study-specific effect sizes are being plotted.

The "+" sign in front of the study label we used for this analysis (variable `studyplus`) indicates that each subsequent study is being added to the previous ones for each analysis. In addition to the ordered values of the specified variable of interest (`year` in our example), the plot also displays the p-values corresponding to the tests of significance of the computed overall effect sizes.

For example, the cumulative odds ratio in the fourth row marked as `+European 2` is 0.70 with a 95% CI of $[0.52, 0.95]$ and a p-value of 0.023. So, based on the first four trials, the overall odds of death is roughly 30% less in the treatment group (treated with streptokinase) compared with the placebo group.

Notice that the first two odds-ratio estimates (and their lower CI limits) are smaller than 0.5. Because we used the `crop(0.5 .)` option, their values are not displayed on the graph. Instead, the arrowheads are displayed at the lower ends of the CI lines to indicate that the lower limits and the effect-size estimates are smaller than 0.5.

Borenstein, Hedges, Higgins, and Rothstein (2009) states that with the inclusion of additional trials in the cumulative meta-analysis, the overall effect sizes become more uniform because the chance of any new trial reporting a drastically different overall effect size is low. Also, the CIs become more narrow because the precision increases as more data become available.

If we look back at the plot, we will notice that starting from 1977, the overall effect size becomes (and stays) highly significant over the next decade of additional trials. Lau et al. (1992) and Borenstein et al. (2009, chap. 42) noted that if cumulative meta-analysis was used at that time to monitor the accumulated evidence from the trials, perhaps, the benefits from streptokinase could have been adopted in practice as early as 1977.

We can also obtain the same results as above but in a table by using `meta summarize`.

```
. meta summarize, cumulative(year) or
  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
        Study label: studyplus
Cumulative meta-analysis summary                    Number of studies =     33
Common-effect model
Method: Mantel-Haenszel
Order variable: year
```

Study	Odds ratio	[95% conf.	interval]	p-value	year
Fletcher	0.159	0.015	1.732	0.131	1959
+Dewar	0.345	0.104	1.141	0.081	1963
+European 1	0.952	0.514	1.760	0.874	1969
+European 2	0.702	0.517	0.951	0.023	1971
+Heikinheimo	0.776	0.589	1.023	0.072	1971
+Italian	0.806	0.624	1.040	0.097	1971
+Australian 1	0.796	0.632	1.004	0.054	1973
+Franfurt 2	0.740	0.594	0.921	0.007	1973
+NHLBI SMIT	0.765	0.616	0.950	0.015	1974
+Frank	0.770	0.623	0.953	0.016	1975
+Valere	0.781	0.635	0.962	0.020	1975
+Klein	0.792	0.644	0.974	0.027	1976
+UK-Collab	0.809	0.670	0.979	0.029	1976
+Austrian	0.762	0.641	0.906	0.002	1977
+Australian 2	0.751	0.636	0.887	0.001	1977
+Lasierra	0.746	0.632	0.881	0.001	1977
+N Ger Collab	0.797	0.683	0.930	0.004	1977
+Witchitz	0.797	0.683	0.928	0.004	1977
+European 3	0.781	0.673	0.906	0.001	1979
+ISAM	0.793	0.690	0.910	0.001	1986
+GISSI-1	0.801	0.734	0.874	0.000	1986
+Olson	0.800	0.733	0.873	0.000	1986
+Baroffio	0.796	0.730	0.869	0.000	1986
+Schreiber	0.795	0.729	0.867	0.000	1986
+Cribier	0.795	0.729	0.868	0.000	1986
+Sainsous	0.794	0.728	0.866	0.000	1986
+Durand	0.793	0.727	0.865	0.000	1987
+White	0.787	0.721	0.858	0.000	1987
+Bassand	0.785	0.721	0.856	0.000	1987
+Vlay	0.785	0.720	0.856	0.000	1988
+Kennedy	0.783	0.718	0.853	0.000	1988
+ISIS-2	0.766	0.718	0.817	0.000	1988
+Wisenberg	0.765	0.717	0.816	0.000	1988

See [META] **meta summarize** and [META] **meta forestplot**.

◁

Heterogeneity: Galbraith plot, meta-regression, and bubble plot

The Galbraith plot (Galbraith 1988) is mainly used to assess heterogeneity of the studies and detect potential outliers. It may also be an alternative to forest plots for summarizing meta-analysis results, especially when there are many studies. See [META] **meta galbraithplot**.

Meta-regression performs a weighted linear regression of effect sizes on moderators; see [META] **meta regress**. With one moderator, the relationship between the effect sizes and the moderator may be further explored via a bubble plot after meta-regression; see [META] **estat bubbleplot**.

In this section, we will demonstrate how to use Galbraith plots, meta-regression, and bubble plots to assess heterogeneity and examine relationships between effects sizes and moderators.

▷ Example 7: Computing log risk-ratios using meta esize

Consider the BCG dataset described in *Efficacy of BCG vaccine against tuberculosis (bcg.dta)*.

```
. use https://www.stata-press.com/data/r18/bcg, clear
(Efficacy of BCG vaccine against tuberculosis)
. describe studylbl npost nnegt nposc nnegc latitude
```

Variable name	Storage type	Display format	Value label	Variable label
studylbl	str27	%27s		Study label
npost	int	%9.0g		Number of TB positive cases in treated group
nnegt	long	%9.0g		Number of TB negative cases in treated group
nposc	int	%9.0g		Number of TB positive cases in control group
nnegc	long	%9.0g		Number of TB negative cases in control group
latitude	byte	%9.0g		Absolute latitude of the study location (in degrees)

As in example 5, this dataset also records summary data for a two-group comparison of binary outcomes, so we will again use `meta esize` to compute our effect sizes.

In this example, our effect size of interest is a risk ratio. Just like with odds ratios, the meta-analysis of risk ratios is performed in the log metric, so we will be computing log risk-ratios.

```
. meta esize npost nnegt nposc nnegc, esize(lnrratio) studylabel(studylbl)
Meta-analysis setting information
 Study information
    No. of studies: 13
       Study label: studylbl
        Study size: _meta_studysize
      Summary data: npost nnegt nposc nnegc
       Effect size
             Type: lnrratio
            Label: Log risk-ratio
         Variable: _meta_es
   Zero-cells adj.: None; no zero cells
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

Our specification of `meta esize` is similar to that from example 5, except here we specify the `esize(lnrratio)` option to compute log risk-ratios instead of the default log odds-ratios.

The output indicates that there are $K = 13$ studies in the meta-analysis and the default random-effects meta-analysis model (with the REML estimation method) will be used.

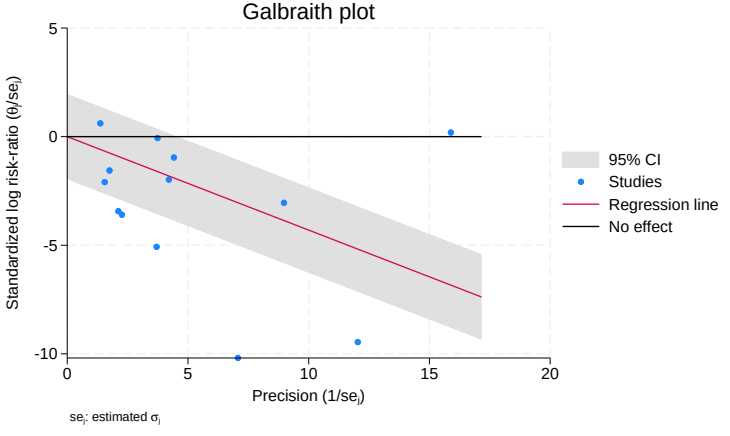
◁

Let's investigate the presence of heterogeneity in these data. For the purpose of illustration, we will do this using a Galbraith plot; see [META] **meta galbraithplot**.

▷ Example 8: Galbraith plot

We use `meta galbraithplot` to produce a Galbraith plot for the BCG data.

```
. meta galbraithplot
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Common effect
             Method: Inverse-variance
```



The blue circles form a scatterplot of the study-specific standardized log risk-ratios against study precisions. Studies that are close to the y axis have low precision. Precision of studies increases as you move toward the right on the x axis.

The reference black line ($y = 0$) represents the "no-effect" line. If a circle is above the reference line, the risk in the treatment group is higher than the risk in the control group for that study. Conversely, if a circle is below the line, the risk in the treatment group is lower than the risk in the control group.

The red line is the regression line through the origin. The slope of this line equals the estimate of the overall effect size. In the absence of substantial heterogeneity, we expect around 95% of the studies to lie within the 95% CI region (shaded area). In our example, there are 6 (out of 13) trials that are outside the CI region. We should suspect the presence of heterogeneity in these data, and we will investigate the reasons behind it in example 9. For more interpretation of the above Galbraith plot, see [META] **meta galbraithplot**.

◁

We have established that there is heterogeneity among the studies. Let's explore this further using meta-regression.

▷ Example 9: Meta-regression

As we discussed in *Subgroup meta-analysis*, when effect sizes vary greatly between different subgroups, one can perform separate meta-analysis on each subgroup to account for the between-study heterogeneity. But what if there is an association between the effect sizes and other study-level covariates or moderators that may be continuous? Meta-regression addresses this problem. Its goal is to investigate whether the differences between the effect sizes can be explained by one or more moderators. See *Introduction* of [META] **meta regress**.

The efficacy of the BCG vaccine against TB may depend on many factors such as the presence of environmental mycobacteria that provides some immunity to TB. Berkey et al. (1995) suggested that the distance of a study from the equator (the absolute latitude) may be used as a proxy for the presence of environmental mycobacteria and perhaps explain the lower efficacy of the BCG vaccine against TB in some studies. Borenstein et al. (2009) also commented that, in hotter climates, the vaccine may lose potency and certain bacteria necessary for the vaccine to work well are less likely to survive with more exposure to sunlight.

Following Berkey et al. (1995), we will explore these observations by using `meta regress` with the centered latitude as the moderator.

First, we generate a new variable, `latitude_c`, that is the mean-centered version of `latitude`. The mean value of `latitude`, 33.46, can be thought of as the latitude of the city of Atlanta in the United States or the city of Beirut in Lebanon.

```
. summarize latitude, meanonly
. generate double latitude_c = latitude - r(mean)
. label variable latitude_c "Mean-centered latitude"
```

We then fit meta-regression with `latitude_c` as the moderator.

```
. meta regress latitude_c
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Random-effects meta-regression                  Number of obs  =        13
Method: REML                                    Residual heterogeneity:
                                                          tau2 =  .07635
                                                       I2 (%)  =   68.39
                                                          H2   =    3.16
                                                R-squared (%)  =   75.63
                                                Wald chi2(1)   =    16.36
                                                Prob > chi2    =   0.0001
------------------------------------------------------------------------------
     _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
--------------+---------------------------------------------------------------
  latitude_c |  -.0291017   .0071953    -4.04   0.000    -.0432043   -.0149991
       _cons |  -.7223204   .1076535    -6.71   0.000    -.9333174   -.5113234
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(11) = 30.73   Prob > Q_res = 0.0012
```

The regression coefficient for `latitude_c` is −0.0291, which means that every one degree of latitude corresponds to a decrease of 0.0291 units in the log risk-ratio. In other words, the vaccine appears to work better in colder climates.

The proportion of between-study variance explained by the covariates can be assessed via the R^2 statistic. Here roughly 76% of the between-study variance is explained by the covariate `latitude_c`. From the value of I^2 in the output, roughly 68% of the residual variation is due to heterogeneity, which may potentially be explained by other covariates, with the other 32% due to the within-study sampling variability.

The test statistic for residual homogeneity, Q_{res}, is 30.73 with a p-value of 0.0012, so the null hypothesis of no residual heterogeneity is rejected, which is consistent with the reported residual heterogeneity summaries.

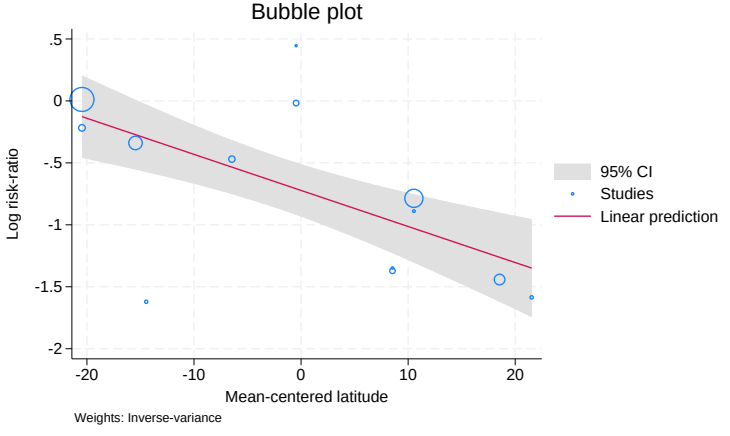
See [META] **meta regress** for more examples.

◁

▷ Example 10: Bubble plot

Whenever there is one continuous covariate in the meta-regression, we may explore the relationship between the effect sizes and that covariate via a bubble plot using the `estat bubbleplot` command. Continuing with example 9, we explore the relationship between the log risk-ratios and `latitude_c`.

```
. estat bubbleplot
```



The bubble plot is a scatterplot of effect sizes and covariate values. Each study is represented by a circle with the size of the circle proportional to the effect-size precision, $1/\hat{\sigma}_j^2$. The fitted line (predicted log risk-ratios) is also plotted on the graph.

The log risk-ratio for the BCG vaccine decreases as the distance from the equator increases. The plot also reveals a few outlying studies that require more thorough investigation. We continue exploring this model in [META] **meta regress postestimation**.

See [META] **estat bubbleplot**.

◁

Funnel plots for exploring small-study effects

A funnel plot (Light and Pillemer 1984) plots study-specific effect sizes against measures of study precision such as standard errors. This plot is commonly used to explore publication bias or, more precisely, small-study effects. Small-study effects (Sterne, Gavaghan, and Egger 2000) arise when smaller studies tend to report different results such as larger effect-size estimates than larger studies. In the absence of small-study effects, the shape of the plot should resemble a symmetric inverted funnel.

Publication bias arises when smaller studies with nonsignificant findings are being suppressed from publication. It is one of the more common reasons for the presence of small-study effects, which leads to the asymmetry of the funnel plot. Another common reason for the asymmetry in the funnel plot is the presence of between-study heterogeneity.

See *Introduction* in [META] **meta funnelplot** for details.

▷ Example 11: Funnel plot

Let's explore the funnel-plot asymmetry for the NSAIDS dataset described in *Effectiveness of nonsteroidal anti-inflammatory drugs (nsaids.dta)*.

```
. use https://www.stata-press.com/data/r18/nsaids, clear
(Effectiveness of nonsteroidal anti-inflammatory drugs)
. describe
Contains data from https://www.stata-press.com/data/r18/nsaids.dta
 Observations:            37                  Effectiveness of nonsteroidal
                                                anti-inflammatory drugs
    Variables:             5                  24 Apr 2022 17:09
                                              (_dta has notes)
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
study           byte    %8.0g                 Study ID
nstreat         byte    %8.0g                 Number of successes in the
                                                treatment arm
nftreat         byte    %9.0g                 Number of failures in the
                                                treatment arm
nscontrol       byte    %8.0g                 Number of successes in the
                                                control arm
nfcontrol       byte    %9.0g                 Number of failures in the control
                                                arm
-------------------------------------------------------------------------------
Sorted by:
```

As before, our first step is to declare our data. `nsaids.dta` records summary data for a two-group comparison of binary outcomes, so we will again use `meta esize` to compute our effect sizes as in example 5 and example 7.

Our effect size of interest is an odds ratio, so we can use the default specification of `meta esize`.

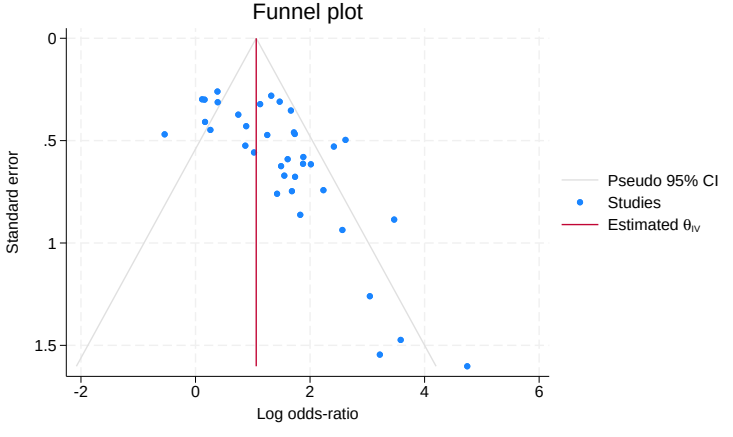
```
. meta esize nstreat-nfcontrol
Meta-analysis setting information
 Study information
    No. of studies: 37
       Study label: Generic
        Study size: _meta_studysize
      Summary data: nstreat nftreat nscontrol nfcontrol
       Effect size
             Type: lnoratio
            Label: Log odds-ratio
         Variable: _meta_es
   Zero-cells adj.: 0.5, only0
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

In the above, instead of listing all four variables with `meta esize` as we did in previous examples, we use one of the varlist shortcuts (see **[U] 11.4 varname and varlists**) to include all variables between `nstreat` and `nfcontrol`. We could do this because our variables appear in the dataset in the same order they need to be listed with `meta esize`: numbers of successes and failures in the treatment group followed by those in the control group.

There are $K = 37$ trials in this dataset. We will continue using the default random-effects meta-analysis model with the REML estimation method.

We use `meta funnelplot` to produce a funnel plot for the NSAIDS data.

```
. meta funnelplot
  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Common effect
             Method: Inverse-variance
```



On a funnel plot, the more precise trials (with smaller standard errors) are displayed at the top of the funnel, and the less precise ones (with larger standard errors) are displayed at the bottom. The red reference line is plotted at the estimate of the overall effect size, the overall log odds-ratio in our example. In the absence of small-study effects, we would expect the points to be scattered around the reference line with the effect sizes from smaller studies varying more around the line than those from larger studies, forming the shape of an inverted funnel.

In our plot, there is an empty space in the bottom left corner. This suggests that the smaller trials with log odds-ratio estimates close to zero may be missing from the meta-analysis.

See [META] **meta funnelplot** for more examples.

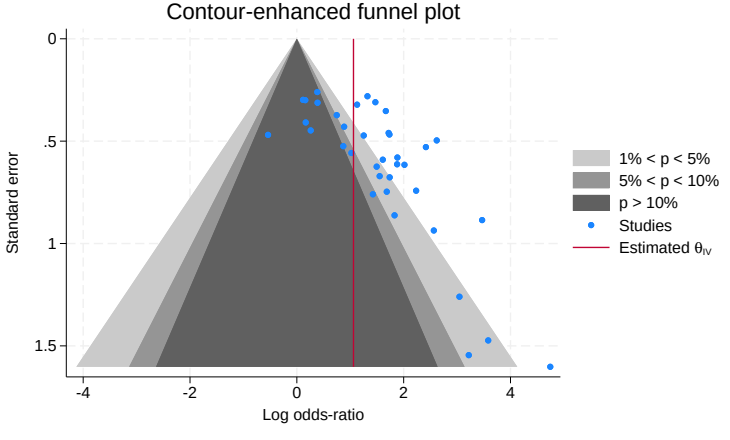
◁

▷ Example 12: Contour-enhanced funnel plot

The asymmetry is evident in the funnel plot from example 11, but we do not know the cause for this asymmetry. The asymmetry can be the result of publication bias or may be because of other reasons. The so-called contour-enhanced funnel plots can help determine whether the asymmetry of the funnel plot is because of publication bias. The contour lines that correspond to certain levels of statistical significance (1%, 5%, and 10%) of tests of individual effects are overlaid on the funnel plot. Generally, publication bias is suspect when smaller studies are missing in the nonsignificant regions.

Let's add the 1%, 5%, and 10% significance contours to our funnel plot by specifying them in the `contours()` option.

```
. meta funnelplot, contours(1 5 10)
  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Common effect
             Method: Inverse-variance
```



From this plot, we can see that the reported effects of almost all smaller trials (those at the bottom of the funnel) are statistically significant at a 5% level and less. On the other hand, a fair number of the larger trials (at the top of the funnel) reported nonsignificant results. For the funnel plot to look symmetric with respect to the reference line, we should have observed some trials in the middle and the bottom of the darkest region (with p-values larger than 10%). This suggests that we are missing some of the smaller trials with nonsignificant results, which would be consistent with the presence of publication bias.

There is also a chance that the funnel-plot asymmetry is induced by the between-study heterogeneity. Using a random-effects model and investigating the study-level covariates that may account for the heterogeneity should also be considered when exploring the funnel-plot asymmetry.

Also see example 5 of [META] **meta funnelplot** for more details about this example.

◁

Testing for small-study effects

We can test for the presence of small-study effects or, technically, the asymmetry in the funnel plot more formally by using, for example, one of the regression-based tests. The main idea behind these tests is to determine whether there is a statistically significant association between the effect sizes and their measures of precision such as effect-size standard errors.

See *Introduction* in [META] **meta bias** for details.

▷ Example 13: Harbord's regression-based test

In example 11, we investigated the funnel-plot asymmetry visually. Let's check for it more formally by using the `meta bias` command. We will use the Harbord regression-based test (Harbord, Egger, and Sterne 2006), which is often recommended when the effect size of interest is an odds ratio (or log odds-ratio).

To perform this test, we specify the `harbord` option with `meta bias`.

```
. meta bias, harbord

  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se

Regression-based Harbord test for small-study effects
Random-effects model
Method: REML

H0: beta1 = 0; no small-study effects
            beta1 =      3.03
      SE of beta1 =     0.741
                z =      4.09
      Prob > |z| =    0.0000
```

The test uses a type of weighted regression that explores the relationship between the effect sizes and their precision. The slope in that regression, labeled as `beta1` in the output, describes the asymmetry of the funnel plot and represents the magnitude of the small-study effects. The further it is from zero, the more asymmetry is present in the funnel plot.

`meta bias` reports the z-test statistic of 4.09 with a p-value less than 0.0000 for the test of `H0: beta1=0` assuming a random-effects model with the REML estimation method. We have statistically significant evidence to reject the null hypothesis of the funnel-plot symmetry.

See [META] **meta bias**.

◁

Trim-and-fill analysis for addressing publication bias

When the presence of publication bias is suspected, it is important to explore its impact on the final meta-analysis results. The trim-and-fill method of Duval and Tweedie (2000a, 2000b) provides a way to evaluate the impact of publication bias on the results. The idea of the method is to estimate the number of studies potentially missing because of publication bias, impute these studies, and use the observed and imputed studies to obtain the overall estimate of the effect size. This estimate can then be compared with the estimate obtained using only the observed studies. For details, see *Introduction* in [META] **meta trimfill**.

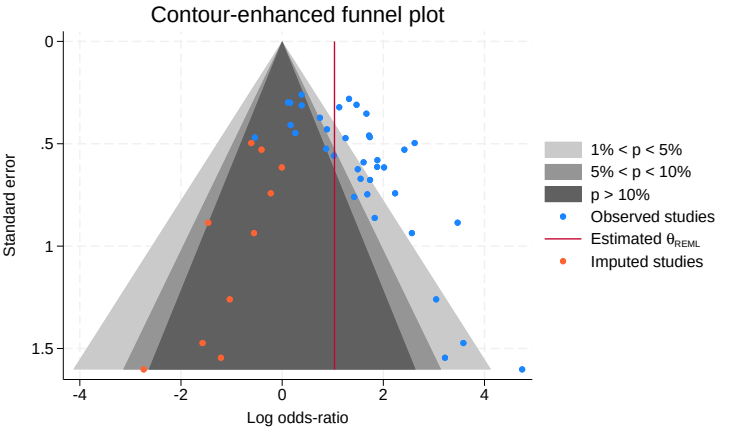
▷ Example 14: Trim-and-fill analysis

From example 12 and example 13, we suspect the presence of publication bias in the meta-analysis of the NSAIDS data. Let's use the trim-and-fill method to investigate the impact of potentially missing studies on the estimate of the overall log odds-ratio.

We use the `meta trimfill` command. We specify the `eform` option (synonym for `or` when the computed effect sizes are log odds-ratios) to report the results as odds ratios instead of the default log odds-ratios. We also draw a contour-enhanced funnel plot that contains both the observed and imputed studies.

```
. meta trimfill, eform funnel(contours(1 5 10))
  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Nonparametric trim-and-fill analysis of publication bias
Linear estimator, imputing on the left
Iteration                             Number of studies =     47
  Model: Random-effects                       observed =     37
 Method: REML                                 imputed =     10
Pooling
  Model: Random-effects
 Method: REML
```

Studies	Odds ratio	[95% conf.	interval]
Observed	3.752	2.805	5.018
Observed + Imputed	2.815	2.067	3.832



`meta trimfill` reports that 10 hypothetical studies are estimated to be missing. When 10 studies are imputed and added to the meta-analysis, the overall odds ratio reduces from 3.752 (based on 37 observed studies) to 2.815 (based on 47 observed and imputed studies). This suggests that the treatment benefit as reported in the literature may be larger than it would be in the absence of publication bias.

From the funnel plot, almost all the imputed studies fall in the darkest-gray region corresponding to a *p*-value of more than 10%. This further supports the conclusion that the small-study effect is most likely because of publication bias.

See [META] **meta trimfill**.

◁

Multivariate meta-regression

Multivariate meta-regression is a multivariate statistical technique used to investigate reasons behind between-study heterogeneity of multiple dependent effect sizes. The technique explores whether there are associations between the effect sizes and other study-level covariates or moderators. You can think of multivariate meta-regression as an extension of meta-regression in univariate meta-analysis to the multivariate setting.

▷ Example 15: Multivariate meta-regression

In this example, we will use the periodontal disease dataset described in *Treatment of moderate periodontal disease (periodontal.dta)* to explore whether the moderator `pubyear` can explain some of the between-study heterogeneity of the two dependent effect-size variables `y1` and `y2`. We will perform a random-effects multivariate meta-regression using the `meta mvregress` command. Unlike other `meta` commands that are designed for standard meta-analysis, the `meta mvregress` command does not require your dataset to be declared as meta data.

```
. meta mvregress y1 y2 = pubyear, wcovvariables(v11 v12 v22)
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -3.5544446
Iteration 1:  Log restricted-likelihood = -3.5402086
Iteration 2:  Log restricted-likelihood = -3.5399568
Iteration 3:  Log restricted-likelihood = -3.5399567
Multivariate random-effects meta-regression     Number of obs     =         10
Method: REML                                    Number of studies =          5
                                                Obs per study:
                                                              min =          2
                                                              avg =        2.0
                                                              max =          2
                                                Wald chi2(2)      =       0.40
Log restricted-likelihood = -3.5399567          Prob > chi2       =     0.8197
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
y1						
pubyear	.0048615	.0218511	0.22	0.824	-.0379658	.0476888
_cons	.3587569	.07345	4.88	0.000	.2147975	.5027163
y2						
pubyear	-.0115367	.0299635	-0.39	0.700	-.070264	.0471907
_cons	-.3357368	.0979979	-3.43	0.001	-.5278091	-.1436645

```
Test of homogeneity: Q_M = chi2(6) = 125.76                 Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Unstructured:	
sd(y1)	.1429917
sd(y2)	.2021314
corr(y1,y2)	.561385

The output shows information about the optimization algorithm, the iteration log, and the model (random-effects) and method (REML) used for estimation. It also displays the number of studies, $K = 5$, and the total number of observations on the outcomes, $N = 10$, which is equal to Kd because no observations are missing. The minimum, maximum, and average numbers of observations per study are also reported. Because there were no missing observations, all of these numbers are identical and are equal to 2. The Wald statistic, $\chi^2 = 0.4$, tests the joint hypothesis that the coefficients of `pubyear` for outcomes `y1` and `y2` are equal to 0.

The first table displays the fixed-effects coefficients for each dependent (outcome) variable. The coefficients of `pubyear` for outcomes `y1` and `y2` are not significant ($p = 0.824$ and $p = 0.7$, respectively), so it does not appear that `pubyear` explains much of the between-study heterogeneity of effect sizes `y1` and `y2`. In fact, the multivariate Cochran's homogeneity test strongly suggests the presence of a between-study heterogeneity even after accounting for `pubyear`: $Q_{\mathrm{M}} = 125.76$ with a $p < 0.0001$.

The second table displays the random-effects parameters, which are used to compute an estimate of the between-study covariance matrix $\boldsymbol{\Sigma}$. For details, see [META] **meta mvregress**.

After you fit your model, you can use `estat heterogeneity` to assess the residual heterogeneity in your model. To conduct other postestimation analysis, see [META] **meta mvregress postestimation**.

◁

Multilevel meta-regression

Multilevel meta-regression is a statistical technique used to study the relationship between potentially dependent effect sizes and covariates. The dependence among the effect sizes stems from a hierarchical or multilevel structure that is assumed present in the data. The standard random-effects meta-analysis can be viewed as a two-level meta-analysis model with studies as level-2 groups and subjects within studies as level-1 observations. When the term "multilevel meta-analysis" is used in the literature, it typically refers to models that incorporate more than two levels of hierarchy.

▷ Example 16: Multilevel meta-regression

Møller and Mousseau (2015) conducted a meta-analysis to investigate the effect of radiation from Chernobyl on mutation rates across different taxonomic groups (`taxon`) and species (`species`). The relation between radiation and mutation rates was quantified by Pearson's product-moment correlation coefficient (`correlation`). Study labels are stored in variable `studylbl`. A key feature of this dataset is that most studies contributed more than one observed effect size. Therefore, the effect sizes, identified by variable `id` (level 2), can be seen as nested within `studylbl` (level 3). The original dataset had 45 studies reporting 172 effect sizes corresponding to 8 different taxonomic groups. Here we focus only on the radiation effect on mutation rates for the largest two taxonomic groups in the dataset: mammals and plants. This leaves us with 42 studies reporting 158 effect sizes. We first describe the variables that will be used in our model:

```
. use https://www.stata-press.com/data/r18/chernobyl
(Effect of radiation from Chernobyl on mutation rates)
. describe studylbl - taxon
```

Variable name	Storage type	Display format	Value label	Variable label
studylbl	str29	%29s		Study label
id	int	%9.0g		Effect-size ID
z	double	%10.0g		Fisher's z-transformed correlations
var	float	%9.0g		Variance of Fisher's z-transformed correlations
taxon	byte	%9.0g	taxon1	Taxonomic group

Variables `z` and `var` store Fisher's z-transformed correlation values and their variances. This transformed metric is typically used for estimation when pooling correlations; see example 10 of [META] **meta summarize** for details about Fisher's z-transformed correlations and their asymptotic standard-errors computation.

Because multiple effect sizes are nested within each study, we fit the three-level random-intercepts model

$$\mathtt{z}_{jk} = \beta_1 I(\mathtt{taxon}_j = \mathtt{mammals}) + \beta_2 I(\mathtt{taxon}_j = \mathtt{plants}) + u_j^{(3)} + u_{jk}^{(2)} + \epsilon_{jk}, \quad j = 1, \ldots, 42$$

where $u_j^{(3)} \sim N(0, \tau_3^2)$, $u_{jk}^{(2)} \sim N(0, \tau_2^2)$, and $\epsilon_{jk} \sim N(0, \mathtt{var}_{jk})$. $I(\mathtt{taxon}_j = \mathtt{mammals})$ and $I(\mathtt{taxon}_j = \mathtt{plants})$ are indicator variables for the mammals and plants taxonomic groups, respectively. You can think of the above model as a form of multilevel subgroup analysis.

We will perform a multilevel meta-regression using the `meta meregress` command. Unlike other `meta` commands that are designed for standard meta-analysis, the `meta meregress` command does not require your dataset to be declared as `meta` data.

```
. meta meregress z ibn.taxon, noconstant || studylbl:|| id:, esvarvariable(var)
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -321.43393  (not concave)
Iteration 1:  Log restricted-likelihood = -187.61074
Iteration 2:  Log restricted-likelihood = -183.46421
Iteration 3:  Log restricted-likelihood = -181.89438
Iteration 4:  Log restricted-likelihood = -181.83596
Iteration 5:  Log restricted-likelihood = -181.83585
Iteration 6:  Log restricted-likelihood = -181.83585
Computing standard errors ...
Multilevel REML meta-regression                         Number of obs =    158
        Grouping information
------------------------------------------------------------------------------
                |     No. of       Observations per group
 Group variable |     groups    Minimum    Average    Maximum
----------------+-------------------------------------------------------------
       studylbl |         42          1        3.8         22
             id |        158          1        1.0          1
------------------------------------------------------------------------------
                                                Wald chi2(2)  =     130.11
Log restricted-likelihood = -181.83585          Prob > chi2   =     0.0000
------------------------------------------------------------------------------
           z |  Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       taxon |
    Mammals |   .6622741   .1066936     6.21   0.000     .4531586    .8713897
     Plants |   1.031014   .1077358     9.57   0.000     .8198556    1.242172
------------------------------------------------------------------------------
Test of homogeneity: Q_M = chi2(156) =  1.0e+05           Prob > Q_M = 0.0000

------------------------------------------------------------------------------
  Random-effects parameters |   Estimate
----------------------------+-------------------------------------------------
studylbl: Identity          |
                  sd(_cons) |   .2427429
----------------------------+-------------------------------------------------
id: Identity                |
                  sd(_cons) |   .7406531
------------------------------------------------------------------------------
```

In the syntax, we wrote `z ibn.taxon, noconstant` to specify the response (`z`) and the fixed-effects part of the model. The `esvarvariable(var)` option specifies the variable (`var` in our case) that stores the effect-size variances (sampling variances). The `|| studylbl: || id:` portion of the syntax adds to the model the random intercepts (the $u_j^{(3)}$'s and $u_{jk}^{(2)}$'s) at the respective `studylbl` and `id` levels. The order in which the levels are specified (from left to right) is important—`meta meregress` assumes that `id` is nested within `studylbl`. Because the above model is a random-intercepts three-level meta-regression (that is, a model without random slopes), it could have also been fit by using the `meta multilevel` command ([META] **meta multilevel**), which provides a simpler syntax for models with only random intercepts:

```
. meta multilevel z ibn.taxon, noconstant relevels(studylbl id) esvarvariable(var)
```

The output shows information about the optimization algorithm, the iteration log, and the estimation method (REML). It also displays the total number of effect sizes, $n = 158$. The minimum, maximum, and average numbers of observations per group at each hierarchical level are also reported. The Wald statistic, $\chi^2 = 130.11$, tests the joint hypothesis that Fisher's z-values for mammals and plants are equal to 0.

The second table displays the fixed-effects coefficients. Both overall effect sizes for mammals and plants are different from 0. The interpretation of the results, however, is easier in the natural correlation-coefficient metric, which we can compute using the inverse transformation:

$$\texttt{rho} = \frac{\exp(2\texttt{z}) - 1}{\exp(2\texttt{z}) + 1} = \texttt{tanh(z)}$$

For example, you may obtain the value of the correlation coefficient corresponding to mammals and its confidence interval as follows:

```
. display tanh(e(b)[1,1])
.57987485

. display "[" tanh(r(table)["ll",1]) ", " tanh(r(table)["ul",1]) "]"
[.42449189, .70207952]
```

The multilevel Cochran's homogeneity test strongly suggests the presence of heterogeneity among the effect sizes even after partitioning the data by taxonomic groups ($p < 0.0001$).

The third table displays the random-effects parameters, which are estimates of the level-3 and level-2 random-effects standard deviations, τ_3 and τ_2, respectively. For details, see [META] **meta meregress**.

After you fit your model, you can use `estat heterogeneity` to assess the multilevel heterogeneity in your model. To conduct other postestimation analysis, see [META] **meta me postestimation**.

◁

Acknowledgments

Previous and still ongoing work on meta-analysis in Stata influenced the design of `meta`. We gratefully acknowledge the contributions of the Stata developers who wrote the community-contributed commands. We thank Jonathan Sterne of the University of Bristol, Roger Harbord of the Met Office, Ross Harris of Public Health England, Thomas Steichen (retired) of RJRT, Mike Bradburn of the University of Sheffield, Jon Deeks of the University of Birmingham, and Doug Altman (1948–2018) of the University of Oxford, for `metan`; Evangelos Kontopantelis and David Reeves, both of the University of Manchester, for `metaan` and `ipdforest`; Roger Harbord, Julian Higgins of the University of Bristol, and Stephen Sharp of the MRC Epidemiology Unit, University of Cambridge, for `metareg`; Jonathan Sterne and Ross Harris for `metacum`; Jonathan Sterne and Roger Harbord for `metafunnel`; Tom Palmer of the University of Bristol, Alex Sutton of the University of Leicester, Santiago Moreno of HEOR Global, and Jaime Peters of the University of Exeter for `confunnel`; Roger Harbord, Ross Harris, and Jonathan Sterne for `metabias`; Thomas Steichen for `metatrim`; Roger Harbord and Penny Whiting of the University of Bristol for `metandi`; Ian White of the MRC Clinical Trials Unit at UCL for `mvmeta` and `network`; and David Fisher of the MRC Clinical Trials Unit at UCL for `ipdmetan` and `admetan`, and many more.

We also thank the editors, Tom Palmer and Jonathan Sterne, of the Stata Press book *Meta-Analysis in Stata: An Updated Collection from the Stata Journal* and the authors of the articles therein for providing valuable information about meta-analysis, in addition to developing the meta-analysis software.

References

Antczak-Bouckoms, A., K. Joshipura, E. Burdick, and J. F. Camilla Tulloch. 1993. Meta-analysis of surgical versus non-surgical methods of treatment for periodontal disease. *Journal of Clinical Periodontology* 20: 259–268. https://doi.org/10.1111/j.1600-051X.1993.tb00355.x.

Berkey, C. S., D. C. Hoaglin, F. Mosteller, and G. A. Colditz. 1995. A random-effects regression model for meta-analysis. *Statistics in Medicine* 14: 395–411. https://doi.org/10.1002/sim.4780140406.

———. 1998. Meta-analysis of multiple outcomes by regression with random effects. *Statistics in Medicine* 17: 2537–2550. https://doi.org/10.1002/(SICI)1097-0258(19981130)17:22%3C2537::AID-SIM953%3E3.0.CO;2-C.

Borenstein, M., L. V. Hedges, J. P. T. Higgins, and H. R. Rothstein. 2009. *Introduction to Meta-Analysis*. Chichester, UK: Wiley.

Colditz, G. A., T. F. Brewer, C. S. Berkey, M. E. Wilson, E. Burdick, H. V. Fineberg, and F. Mosteller. 1994. Efficacy of BCG vaccine in the prevention of tuberculosis: Meta-analysis of the published literature. *Journal of the American Medical Association* 271: 698–702. https://doi.org/10.1001/jama.1994.03510330076038.

Duval, S., and R. L. Tweedie. 2000a. Trim and fill: A simple funnel-plot–based method of testing and adjusting for publication bias in meta-analysis. *Biometrics* 56: 455–463. https://doi.org/10.1111/j.0006-341X.2000.00455.x.

———. 2000b. A nonparametric "trim and fill" method of accounting for publication bias in meta-analysis. *Journal of the American Statistical Association* 95: 89–98. https://doi.org/10.1080/01621459.2000.10473905.

Galbraith, R. F. 1988. A note on graphical representation of estimated odds ratios from several clinical trials. *Statistics in Medicine* 7: 889–894. https://doi.org/10.1002/sim.4780070807.

Harbord, R. M., M. Egger, and J. A. C. Sterne. 2006. A modified test for small-study effects in meta-analyses of controlled trials with binary endpoints. *Statistics in Medicine* 25: 3443–3457. https://doi.org/10.1002/sim.2380.

Lau, J., E. M. Antman, J. Jimenez-Silva, B. Kupelnick, F. Mosteller, and T. C. Chalmers. 1992. Cumulative meta-analysis of therapeutic trials for myocardial infarction. *New England Journal of Medicine* 327: 248–254. https://doi.org/10.1056/NEJM199207233270406.

Light, R. J., and D. B. Pillemer. 1984. *Summing Up: The Science of Reviewing Research*. Cambridge, MA: Harvard University Press. https://doi.org/10.2307/j.ctvk12px9.

Møller, A. P., and T. A. Mousseau. 2015. Strong effects of ionizing radiation from Chernobyl on mutation rates. *Scientific Reports* 5: 8363. https://doi.org/10.1038/srep08363.

Moore, R. A., M. R. Tramèr, D. Carroll, P. J. Wiffen, and H. J. McQuay. 1998. Quantitative systematic review of topically applied non-steroidal anti-inflammatory drugs. *BMJ* 316: 333–338. https://doi.org/10.1136/bmj.316.7128.333.

Raudenbush, S. W. 1984. Magnitude of teacher expectancy effects on pupil IQ as a function of the credibility of expectancy induction: A synthesis of findings from 18 experiments. *Journal of Educational Psychology* 76: 85–97. https://doi.org/10.1037/0022-0663.76.1.85.

Rice, K., J. P. T. Higgins, and T. S. Lumley. 2018. A re-evaluation of fixed effect(s) meta-analysis. *Journal of the Royal Statistical Society*, A ser., 181: 205–227. https://doi.org/10.1111/rssa.12275.

Rosenthal, R., and L. Jacobson. 1968. Pygmalion in the classroom. *Urban Review* 3: 16–20. https://doi.org/10.1007/BF02322211.

Sterne, J. A. C., D. Gavaghan, and M. Egger. 2000. Publication and related bias in meta-analysis: Power of statistical tests and prevalence in the literature. *Journal of Clinical Epidemiology* 53: 1119–1129. https://doi.org/10.1016/S0895-4356(00)00242-0.

Also see

[META] **Intro** — Introduction to meta-analysis

[META] **Glossary**

meta data — Declare meta-analysis data

Description Remarks and examples References Also see

Description

This entry describes how to prepare your data for meta-analysis using the `meta` commands.

In a nutshell, do the following:

1. If you have access to summary data, use `meta esize` to compute and declare effect sizes such as an odds ratio or a Hedges's g.
2. Alternatively, if you have only precomputed (generic) effect sizes, use `meta set`.
3. To update some of your meta-analysis settings after the declaration, use `meta update`.
4. To check whether your data are already meta set or to see the current meta settings, use `meta query`.
5. If you want to perform multivariate meta-regression using `meta mvregress`, you do not need to `meta set` your data.

Remarks and examples

Remarks are presented under the following headings:

Overview
Declaring meta-analysis information
  Declaring effect sizes and their precision
  Declaring a meta-analysis model
  Declaring a meta-analysis estimation method
  Default meta-analysis model and method
  Declaring a confidence level for meta-analysis
  Declaring display settings for meta-analysis
  Modifying default meta settings
Meta-analysis information
  Meta settings with meta set
  Meta settings with meta esize
System variables
Examples of data declaration for meta-analysis
  Declaring precomputed effect sizes using meta set
  Computing and declaring effect sizes using meta esize
  Displaying and updating meta settings

Overview

The declaration of your data to be `meta` data is the first step of your meta-analysis in Stata. `meta` data are your original data that also store key variables and characteristics about your specifications, which will be used by all `meta` commands during your meta-analysis session. The declaration step helps minimize mistakes and saves you time—you only need to specify the necessary information once.

You can use `meta set` or `meta esize` to declare your data to be `meta` data. If you have access only to precomputed effect sizes and their standard errors, use `meta set`. If you have access to summary data such as means and standard deviations from individual studies, use `meta esize` to compute the

effect sizes and their standard errors and declare them. The latter is preferable because it provides access to more features such as the Mantel–Haenszel estimation method for a two-group comparison of binary outcomes, which needs access to the actual 2×2 tables and not only the effect sizes for the computations.

For example, suppose that you have variables `es` and `se`, which contain the effect sizes and the corresponding standard errors. You can use

```
. meta set es se
```

to declare your data, and all subsequent `meta` commands will automatically use these variables in the meta-analysis.

To review the current meta settings or to check whether the data are `meta` set, you can use `meta query`; see [META] **meta update**. After your data are declared, you can update some of the meta-analysis specifications by using `meta update`. If you wish to clear the meta settings after your meta-analysis, you can use `meta clear`; see [META] **meta update**.

Declaring meta-analysis information

Two main components of meta-analysis are study-specific effect sizes and their precision. You must specify them during declaration. Other important components include the underlying meta-analysis model and an estimation method. You can specify them during declaration or later during analysis or use Stata's defaults. You can also specify options that affect the output of the `meta` commands. Below, we describe how you can declare various meta-analysis information.

Declaring effect sizes and their precision

As we mentioned above, you must declare study-specific effect sizes and their precision. This is done differently for `meta set` and `meta esize`.

`meta esize` computes effect sizes and their standard errors from summary data and then declares them. `meta set` declares already precomputed effect sizes and their standard errors. Thus, to use `meta set`, you do not need summary data from each study, but you need them for `meta esize`. Some analysis may not be available after `meta set` such as the Mantel–Haenszel estimation method and Harbord's test for the funnel-plot asymmetry because they require access to summary data.

Effect sizes and their precision using meta set. To use `meta set`, you must specify variables containing study-specific effect sizes and their precision. There are two ways to specify the precision of the effect sizes. You can either specify a variable containing the standard errors,

```
. meta set es se
```

Or, instead of the standard errors, specify the confidence intervals, and `meta set` will compute the corresponding standard errors based on them:

```
. meta set es cil ciu
```

In the above, the specified CI variables will be assumed to correspond to the 95% CIs. You can change this by specifying the `civarlevel()` option:

```
. meta set es cil ciu, civarlevel(90)
```

But do not confuse `civarlevel()` with `level()`. The former affects the confidence level only for the specified CI variables. The latter specifies the confidence level for the meta-analysis.

Effect sizes and their precision using meta esize. To use `meta esize`, you must specify summary data for each study. The type of summary data you specify depends on the effect size you wish to compute and consequently on the outcome of interest and type of analysis in the original studies.

`meta esize` computes and declares various effect sizes for two-group comparisons of continuous and binary outcomes, for estimating a single proportion or prevalence, and for estimating a correlation between two variables of interest. For a two-group comparison of continuous outcomes, you must specify the number of observations, means, and standard deviations for each treatment group (group 1) and control group (group 2).

```
. meta esize n1 m1 sd1 n2 m2 sd2
```

To compute effect sizes and their standard errors, `meta esize` also needs to know the type of the effect size. The above assumes Hedges's g standardized mean difference, but you can specify others in the `esize()` option; see effect sizes for a two-group comparison of continuous outcomes in the *estypecnt* table in *Syntax* of [META] **meta esize**.

For a two-group comparison of binary outcomes, you must specify 2×2 contingency tables for each study. You specify them as follows. Each of the four cells is represented by a variable such that each row represents a 2×2 table from a specific study. For instance,

```
. meta esize n11 n12 n21 n22
```

The order in which you specify the four variables is important: the top-left cell first, the top-right cell next, followed by the bottom-left cell, and finally the bottom-right cell. The above computes the log odds-ratio as an effect size, but you can select a different effect size; see effect sizes for a two-group comparison of binary outcomes in the *estypebin* table in *Syntax* of [META] **meta esize**.

`meta esize` can compute effect sizes for estimating a single proportion. You must specify the number of successes (events) and the study sample size.

```
. meta esize ns n
```

The above computes the Freeman–Tukey-transformed proportions, but you can specify other effect sizes in the `esize()` option; see effect sizes for a single proportion in the *estypeprop* table in *Syntax* of [META] **meta esize**.

In StataNow, `meta esize` can compute effect sizes for estimating a correlation between two variables of interest. You must specify the sample correlation coefficients and the study sample size. You must also specify either the `fisherz` option or the `correlation` option,

```
. meta esize rho n, fisherz
```

The above computes the Fisher's z-transformed correlations, but you can request untransformed (raw) correlations by specifying the `correlation` option; see effect sizes for estimating a correlation in the *estypecorr* table in *Syntax* of [META] **meta esize**.

Options affecting effect-size and precision computations with meta esize. Depending on the chosen effect size, `meta esize` provides alternative ways of computing effect sizes and their standard errors.

For the Hedges's g effect size, there are two ways to compute the bias-correction factor used in its formula. For consistency with meta-analysis literature, `meta esize` uses an approximation, but you can specify the `exact` option within `esize()` to use the exact computation:

```
. meta esize n1 m1 sd1 n2 m2 sd2, esize(hedgesg, exact)
```

Note that the `esize` command uses the exact computation.

Both Hedges's g and Cohen's d effect sizes support standard error adjustment of Hedges and Olkin (1985) with `esize()`'s option `holkinse`:

```
. meta esize n1 m1 sd1 n2 m2 sd2, esize(cohend, holkinse)
```

For the (unstandardized) mean difference, you can choose to compute standard errors assuming unequal variance between the two groups:

```
. meta esize n1 m1 sd1 n2 m2 sd2, esize(mdiff, unequal)
```

For two-sample binary data with log odds-ratios or log risk-ratios as effect sizes, `meta esize` automatically adjusts for zero cells when computing effect sizes. By default, it adds 0.5 to all cells of the 2×2 tables that contain at least one zero cell. You can specify other adjustments in the `zerocells(`*zcspec*`)` option. For example, with log odds-ratios, you can specify the treatment-arm continuity correction of Sweeting, Sutton, and Lambert (2004) as `zerocells(tacc)`, or you can request no zero-cell adjustment:

```
. meta esize n11 n12 n21 n22, zerocells(none)
```

Similarly, for one-sample binary data with the logit-transformed proportions or raw proportions as effect sizes, `meta esize` also adjusts for zero cells when computing effect sizes. By default, it adds 0.5 to the number of successes and to the number of failures for studies containing zero successes or zero failures. Other adjustments are also possible via the `zerocells(`*zcspec*`)` option.

See *Options* in [META] **meta esize**.

Declaring a meta-analysis model

Before you proceed with performing meta-analysis, we want you to think about the model underlying your meta-analysis. This decision is important because the selected meta-analysis model will determine the availability of some of the meta-analysis methods and, more importantly, how you interpret the obtained results; see *Comparison between the models and interpretation of the results* in [META] **Intro**. Also, most likely, you will want to use the chosen model during your entire meta-analysis session. Thus, we made the choices for the meta-analysis model and, consequently, the meta-analysis estimation method be part of the initial declaration step. But fear not! If desired, you can easily switch to a different meta-analysis model or method for the rest of your meta-analysis session or reset it temporarily for a particular analysis; see *Modifying default meta settings*.

We discuss the available models and the differences between them in detail in *Meta-analysis models* in [META] **Intro**.

Briefly, there are three models to choose from: a common-effect, fixed-effects, or random-effects model. They can be requested by specifying options `common`, `fixed`, or `random`. If you omit all of these options, the random-effects model will be assumed.

A common-effect model makes a strong assumption about the underlying true effect size being the same across (common to) all studies. When this assumption is true, this model is a reasonable choice. Most likely, you will want to verify the plausibility of this assumption for your data. So a model that allows the study effect sizes to be different may be a better choice during the initial analysis.

A fixed-effects model allows the effect sizes to be different across studies and assumes that they are fixed. You may ask: What does "fixed" mean? Different disciplines may have different definitions of a fixed effect. In the context of meta-analysis, you can think of fixed effects as effects of particular interest. In other words, your research questions and final inference are focused only on the specific studies that were selected in the meta-analysis.

Conversely, a random-effects model assumes that the study effect sizes are random, meaning that they represent a random sample from a larger population of similar studies. The results obtained from a random-effects model can be extended to the entire population of similar studies and not just the ones that were selected in the meta-analysis. The meta-analysis literature recommends to start with a random-effects model, which is also Stata's default for most `meta` commands.

So, which model should you choose? Our recommendation is to start with a random-effects model and explore the heterogeneity, publication bias, and other aspects of your meta-analysis data. If you are interested only in the inference about the particular studies in your data, a fixed-effects model may be a reasonable alternative. We suggest that you avoid using, or at least starting with, a common-effect model unless you verified that the underlying assumption of the common study effects is plausible for your data.

As we described in *Comparison between the models and interpretation of their results* in [META] **Intro**, a fixed-effects model and a common-effect model produce the same results in a meta-analysis. Although the final estimates are the same, their interpretation is different! In a common-effect model, the estimate of the overall effect size is an estimate of the true common effect size, whereas in a fixed-effects model, it is an estimate of the average of true, different study-specific effect sizes. Thus, the `meta` suite provides the two options `common` and `fixed` to emphasize the conceptual differences between the two models. Additionally, when you assume a common-effect model, you essentially imply that certain issues such as study heterogeneity are of no concern in your data. Therefore, when you specify the `common` option, certain commands such as `meta regression` will not be available. This is again our way of reminding you of the underlying assumption of a common-effect model. For other `meta` commands, specifying `common` versus `fixed` will merely change the reported title from, say, "Common-effect meta-analysis" to "Fixed-effects meta-analysis". Nevertheless, the title change is important because it encourages proper interpretation of the results.

Declaring a meta-analysis estimation method

Depending on a chosen meta-analysis model and effect size, there are a number of methods available to estimate the overall effect size. For a common-effect model and a fixed-effects model, the inverse-variance method, `common(invvariance)` and `fixed(invvariance)`, is used with generic effect sizes, which are declared by `meta set`, and with effect sizes for two-sample continuous data and for one-sample binary data, which are declared by `meta esize`. With effect sizes for two-sample binary data (except Peto's log odds-ratio), which are also declared by `meta esize`, the Mantel–Haenszel method, `common(mhaenszel)` or `fixed(mhaenszel)`, is also available.

For a random-effects model, there are several different methods to estimate the between-study variance, which contributes to the weights used to estimate the overall effect size. The default method is REML, `random(reml)`, but other methods such as ML, `random(ml)`, and DerSimonian–Laird, `random(dlaird)`, are also available. See *Syntax* in [META] **meta set** for a full list.

When you specify `random`, the REML method is assumed. When you specify `common` or `fixed`, the inverse-variance method is assumed for all effect sizes except log odds-ratios, log risk-ratios, and risk differences, as specified with `meta esize`. For these effect sizes, the Mantel–Haenszel method is the default method.

See *Meta-analysis estimation methods* in [META] **Intro** for detailed descriptions of the methods.

Default meta-analysis model and method

During declaration, `meta set` and `meta esize` assume a random-effects model unless you specify one of options `fixed` or `common`. It also assumes the REML estimation method unless you specify some other method in option `random()`; see *Declaring a meta-analysis estimation method*.

The declared model will be used by all `meta` commands except `meta funnelplot`, `meta galbraithplot`, and `meta labbeplot`, which, for historical reasons, assume a common-effect model with the inverse-variance estimation method. But you can change the assumed model and method by specifying the corresponding options such as `random(dlaird)` with a `meta` command.

Also see *Modifying default meta settings* for details.

Declaring a confidence level for meta-analysis

By default, `meta set` and `meta esize` assume the 95% confidence level (or as set by `set level`) for the entire meta-analysis. You can change this by specifying the `level()` option with these commands. You can also modify the confidence level after the declaration as we describe in *Modifying default meta settings*.

Declaring display settings for meta-analysis

`meta set` and `meta esize` also provide options to control the output of `meta` commands.

The `studylabel(`*varname*`)` option specifies a string variable that will be used by `meta` commands such as `meta summarize` and `meta forestplot` to label the studies in the output. By default, the generic labels—`Study 1`, `Study 2`, and so on—will be used.

The `eslabel(`*string*`)` option specifies a string that will be used by `meta` commands such as `meta summarize` and `meta forestplot` to label effect sizes in the output. The default label with `meta set` is `Effect size`. The default label with `meta esize` is specific to the chosen effect size. For instance, it is `Log Odds-Ratio` for log odds-ratios.

By default, all `meta` commands display a short summary about the declared meta settings such as the variables containing effect sizes and their standard errors. After the declaration, the `meta` commands do not require you to specify the effect-size variables and standard error variables again. They simply use the corresponding system variables (see *System variables*) created during declaration. The reported summary reminds you that those variables are part of your meta-analysis. You can suppress this summary from all `meta` commands by specifying the `nometashow` option with `meta set` or `meta esize`. You can also suppress this summary for a particular `meta` command by specifying the option with that command; see *Modifying default meta settings*.

Modifying default meta settings

You can modify the default meta settings both during and after the declaration. Some of the settings may even be modified (temporarily) for a particular `meta` command.

You can modify the default settings during the declaration by simply specifying the corresponding options with `meta set` or `meta esize`. For example, when we type

```
. meta set ...
```

a random-effects model with the REML estimation method is assumed. We can specify another estimation method, for example, ML, by using `random(ml)`:

```
. meta set ... , random(ml)
```

Or we can specify a different meta-analysis model, for example, a fixed-effects model:

```
. meta set ... , fixed
```

After the declaration, you can use `meta update` to modify the current settings. For example, we can switch to a common-effect model for the rest of our meta-analysis by typing

```
. meta update, common
```

Now all subsequent `meta` commands will automatically assume a common-effect model.

In the above examples, we used `meta set`, but you can use the same specifications with `meta esize`. We also demonstrated only a few options, but the same principles apply to the other options supported by `meta set` and `meta esize`.

For options `random()`, `common()` (and `common`), `fixed()` (and `fixed`), `level()`, and `nometashow`, we can also modify the current setting temporarily while running a particular `meta` command. For example, suppose that we want to obtain the results assuming a 90% confidence level with `meta summarize`. We can type

```
. meta summarize, level(90)
```

If we wanted all relevant `meta` commands to use the 90% confidence level, we would have typed

```
. meta update, level(90)
```

Meta-analysis information

When you use `meta set` or `meta esize`, they record information about your study, effect sizes and their precision, and meta-analysis model and meta-analysis estimation method, among other things. This information will be used by subsequent `meta` commands. The summary information is mostly the same between the two commands, but `meta esize` records several additional settings.

Let's get familiar with the meta setting information by looking at examples.

Meta settings with meta set

Consider a fictional dataset, `metaset.dta`, containing generic effect sizes and their standard errors stored in the corresponding variables `es` and `se`.

```
. use https://www.stata-press.com/data/r18/metaset
(Generic effect sizes; fictional data)
. describe es se
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
es              double  %10.0g                Effect sizes
se              double  %10.0g                Std. err. for effect sizes
```

At the minimum, with `meta set`, we must specify the variables containing effect sizes and their standard errors. (For other uses of `meta set`, see *Remarks and examples* in [META] **meta set**.)

```
. meta set es se
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: N/A
       Effect size
             Type: <generic>
            Label: Effect size
         Variable: es
        Precision
        Std. err.: se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

The summary is divided into four categories: information about the study, the specified effect sizes, their precision, and meta-analysis model and method. Below, we describe in detail each output category.

```
 Study information

    No. of studies:  10

       Study label:  Generic

        Study size:  N/A
```

The study information consists of the number of studies (10 in our example), a study label (`Generic`), and a study size (`N/A`). If the `studylabel(`*varname*`)` option is specified, the `Study label:` will contain the name of the specified variable. Otherwise, a generic study label—`Study 1`, `Study 2`, and so on—will be used in the output of `meta` commands. If the `studysize(`*varname*`)` option is specified with `meta set`, the `Study size:` will contain the name of the specified variable.

```
        Effect size

              Type:  Generic

             Label:  Effect size

          Variable:  es
```

The effect-size information consists of the type of the effect size, its label, and the variable containing study-specific effect sizes. The effect-size `Type:` is always `Generic` with `meta set`. The effect-size `Label:` is either a generic `Effect size` or as specified in the `eslabel(`*string*`)` option. This label will be used to label the effect sizes in the output of all `meta` commands. The effect-size `Variable:` displays the name of the declared variable containing effect sizes. After the declaration, both commands store study-specific effect sizes in the system variable `_meta_es` (see *System variables*). `meta set` simply copies them from the declared effect-size variable. Thus, `Variable:` will contain the name of the *esvar* variable, `es` in our example, with `meta set`.

```
Precision

Std. err.:  se

       CI:  [_meta_cil, _meta_ciu]

 CI level:  95%
```

The precision information consists of variables containing effect-size standard errors, confidence intervals, and the declared confidence level. As with the effect sizes, `meta set` uses the standard errors specified in the *sevar* variable (variable `se` here). The corresponding confidence intervals are computed using the effect sizes and their standard errors and stored in the system variables `_meta_cil` and `_meta_ciu`. With `meta set`, you can specify confidence intervals instead of the standard errors, in which case the standard errors will be computed from the effect sizes and confidence intervals and stored in `_meta_se`, in which case `Std. err.:` will contain `_meta_se`; see *Syntax* in [META] **meta set**. `CI:` always contains `_meta_cil` and `_meta_ciu`. The specified CI variables will be reported in `User CI:` with their corresponding confidence level reported in `User CI level:`, which is controlled by the `civarlevel()` option. The declared CI variables and the system CI variables will be the same only when `civarlevel()` is the same as `level()`, and the system variables are the ones that are used in the meta-analysis.

`CI level:` reports the confidence level, controlled by the `level()` option, that will be used by all `meta` commands when computing confidence intervals for various meta-analyses such as the CIs of the overall effect size, regression coefficients, and so on. The default confidence level is 95% or as set by `set level`.

```
Model and method

        Model:  Random-effects

       Method:  REML
```

As we pointed out in *Declaring a meta-analysis model*, the meta-analysis model and, consequently, the meta-analysis estimation method are important aspects of your meta-analysis. As such, we made them be part of your declaration step too. By default, a random-effects model with the REML estimation method is assumed for most `meta` commands; see *Default meta-analysis model and method*. You can change the defaults as we describe in *Modifying default meta settings*.

Meta settings with meta esize

Consider `metaescnt.dta`, containing fictional study-specific summary data for continuous outcomes for group 1 and group 2.

```
. use https://www.stata-press.com/data/r18/metaescnt, clear
(Fictional summary data for continuous outcomes)
. describe n1 m1 sd1 n2 m2 sd2
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
n1              byte    %9.0g                 Study sizes of group 1
m1              float   %9.0g                 Means of group 1
sd1             float   %9.0g                 Std. dev. of group 1
n2              byte    %9.0g                 Study sizes of group 2
m2              float   %9.0g                 Means of group 2
sd2             float   %9.0g                 Std. dev. of group 2
```

With meta esize, we must specify the summary data to compute an effect size. Let's focus on the studies comparing the mean differences between the two groups. Our summary data include the numbers of observations and the estimates of means and standard deviations for each group. We specify the variables containing these summaries following the command name.

```
. meta esize n1 m1 sd1 n2 m2 sd2

Meta-analysis setting information

 Study information
    No. of studies: 10
       Study label: Generic
        Study size: _meta_studysize
      Summary data: n1 m1 sd1 n2 m2 sd2

       Effect size
             Type: hedgesg
            Label: Hedges's g
         Variable: _meta_es
   Bias correction: Approximate

        Precision
        Std. err.: _meta_se
    Std. err. adj.: None
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%

  Model and method
            Model: Random effects
           Method: REML
```

The meta setting information from meta esize is almost the same as the one produced by meta set, which we described in *Meta settings with meta set*, but has several additional settings. The summary-data variables are listed under Summary data:. As we mentioned earlier, meta esize computes the effect sizes and their standard errors from the specified summary data, so effect-size Variable: and Std. err.: contain the names of the corresponding system variables, _meta_es and _meta_se. The summary data also include the information about the study size, so Study size: displays the name of the system variable, _meta_studysize, that contains study size, which is equal to the sum of n1 and n2 in our example.

By default, meta esize computes the Hedges's g effect size for the two-group mean comparison. You can specify the esize(*esspec*) option to select a different effect size. For the Hedges's g effect size, there are two methods to compute the underlying bias-correction term: approximate or exact. For consistency with the meta-analysis literature, meta esize, by default, uses an approximation, as indicated in Bias correction: under Effect size. But you can change this by specifying the exact option within esize().

Another additional setting describes the type of adjustment applied when computing the standard errors of the effect sizes; see Std. err. adj.: under Precision. This adjustment is applicable only with the Hedges's g or Cohen's d effect size. No adjustment is made by default, but you can use the holkinse option within esize() to specify the adjustment of Hedges and Olkin (1985). For the mean-difference effect size, you can request the adjustment for unequal group variances by specifying esize()'s option unequal.

Finally, for log odds-ratios or log risk-ratios, meta esize additionally reports the type of adjustment made to the zero cells of contingency tables, which represent the summary data for a two-group comparison of binary outcomes. For these effect sizes, the type of adjustment will be listed in Zero-cells adj.: under Effect size (not applicable in our example). By default, 0.5 is added to each zero cell,

but you can specify the zerocells() option with meta esize to apply a different adjustment or none. The zero-cells adjustment is also reported for meta-analysis of a single proportion when effect sizes are logit-transformed proportions or raw proportions.

System variables

meta set and meta esize store information about the meta-analysis settings in data characteristics ([P] **char**) and system variables.

meta system variables are the variables that begin with _meta_. There are four main variables that are stored by the two commands.

_meta_es stores study-specific effect sizes.

_meta_se stores the standard errors of study-specific effect sizes.

_meta_cil and _meta_ciu store the lower and upper limits of the CIs for study-specific effect sizes. These variables correspond to the confidence level declared for the meta-analysis, the value of which is stored in the data characteristic _meta_level.

Other system variables include integer study identifiers stored in _meta_id, study labels stored in a string variable _meta_studylabel, and study sizes stored in _meta_studysize. _meta_studysize is always stored with meta esize. With meta set, it is stored only when the variable containing study sizes is specified in the studysize() option.

Also see *Stored results* in [META] **meta set** and *Stored results* in [META] **meta esize**.

Examples of data declaration for meta-analysis

In this section, we demonstrate how to prepare data for meta-analysis in Stata for several case studies.

Declaring precomputed effect sizes using meta set

We will demonstrate how to use meta set to declare generic effect sizes.

▷ Example 1: Precomputed log hazard-ratios using meta set

We demonstrate how to declare the meta-analysis data from Steurer et al. (2006), who studied the effect of purine analogues for the treatment of chronic lymphocytic leukemia. Variables loghr and seloghr contain the log hazard-ratios and their standard errors.

```
. use https://www.stata-press.com/data/r18/leukemia2, clear
(Single-agent purine analogue treatment for leukemia)
. describe
Contains data from https://www.stata-press.com/data/r18/leukemia2.dta
 Observations:             4                  Single-agent purine analogue
                                                treatment for leukemia
    Variables:             6                  25 Apr 2022 12:09
                                              (_dta has notes)
------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
author          str14   %14s                * Author
year            int     %8.0g                 Publication year
ntreat          int     %8.0g                 Treatment-group sample size
ncontrol        int     %8.0g                 Control-group sample size
loghr           float   %9.0g                 Log hazard-ratio
seloghr         float   %9.0g                 Standard error for loghr
                                            * indicated variables have notes
------------------------------------------------------------------------------
Sorted by:
```

We use the `meta set` command to declare the effect sizes (log hazard-ratios) and their standard errors.

```
. meta set loghr seloghr
Meta-analysis setting information
 Study information
    No. of studies: 4
       Study label: Generic
        Study size: N/A
       Effect size
             Type: <generic>
            Label: Effect size
         Variable: loghr
        Precision
        Std. err.: seloghr
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

`meta set` reports that there are 4 studies in this dataset. The type of effect size is `Generic` because we used the precalculated effect size. The default label `Effect size` will be used in the output. The command also reports the variables that were used to declare the effect sizes, `loghr`, and their standard errors, `seloghr`. The other settings are as we described in *Meta settings with meta set*.

As we described in *System variables*, `meta set` created several system variables that will be used by other `meta` commands in the computations:

```
. describe _meta*
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
_meta_id        byte    %9.0g                 Study ID
_meta_studyla~l str7    %9s                   Study label
_meta_es        float   %9.0g                 Generic ES
_meta_se        float   %9.0g                 Std. err. for ES
_meta_cil       double  %10.0g                95% lower CI limit for ES
_meta_ciu       double  %10.0g                95% upper CI limit for ES
. list _meta*
```

	_meta_id	_meta~el	_meta_es	_meta_se	_meta_cil	_meta_ciu
1.	1	Study 1	-.592	.345	-1.2681876	.08418756
2.	2	Study 2	-.0791	.0787	-.23334916	.07514916
3.	3	Study 3	-.237	.144	-.51923481	.0452348
4.	4	Study 4	.163	.312	-.44850877	.77450878

`_meta_id` contains integers identifying the studies, and `_meta_studylabel` contains the study labels. `_meta_es` and `_meta_se` contain log hazard-ratios and their standard errors, and `_meta_cil` and `_meta_ciu` contain the corresponding lower and upper bounds of the 95% CIs for log hazard-ratios.

We did not specify the `studylabel()` option in this example, so generic labels will be used in the output of other `meta` commands such as `meta summarize`:

```
. meta summarize
  Effect-size label: Effect size
        Effect size: loghr
          Std. err.: seloghr
Meta-analysis summary                     Number of studies =      4
Random-effects model                      Heterogeneity:
Method: REML                                          tau2 =  0.0000
                                                   I2 (%) =    0.00
                                                      H2 =    1.00
```

Study	Effect size	[95% conf.	interval]	% weight
Study 1	-0.592	-1.268	0.084	3.68
Study 2	-0.079	-0.233	0.075	70.70
Study 3	-0.237	-0.519	0.045	21.12
Study 4	0.163	-0.449	0.775	4.50
theta	-0.120	-0.250	0.009	

```
Test of theta = 0: z = -1.82                  Prob > |z| = 0.0688
Test of homogeneity: Q = chi2(3) = 3.62         Prob > Q = 0.3049
```

Generic labels `Study 1`, `Study 2`, `Study 3`, and `Study 4` are used to label the studies. Also, the generic label `Effect size` is used to label the log hazard-ratios. See [META] **meta summarize** for details about `meta summarize`.

We can provide more descriptive labels for the studies and the effect sizes by specifying options `studylabel()` and `eslabel()`.

```
. generate studylbl = author + " (" + string(year) + ")"
. meta set loghr seloghr, studylabel(studylbl) eslabel("Ln(HR)")
Meta-analysis setting information
 Study information
    No. of studies: 4
       Study label: studylbl
        Study size: N/A
       Effect size
             Type: <generic>
            Label: Ln(HR)
         Variable: loghr
        Precision
        Std. err.: seloghr
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

We created a new variable, `studylbl`, that combines the author and year information of the published studies to use as our study labels. `meta set` reported that `studylbl` will be used to label the studies and `Ln(HR)` to label the effect sizes.

If we now rerun `meta summarize` (suppressing the table header), we see the new labels in the output.

```
. meta summarize, noheader
  Effect-size label: Ln(HR)
        Effect size: loghr
          Std. err.: seloghr
        Study label: studylbl
```

Study	Ln(HR)	[95% conf.	interval]	% weight
Johnson et al. (1996)	-0.592	-1.268	0.084	3.68
Leporrier (2001)	-0.079	-0.233	0.075	70.70
Rai (2000)	-0.237	-0.519	0.045	21.12
Robak (2000)	0.163	-0.449	0.775	4.50
theta	-0.120	-0.250	0.009	

```
Test of theta = 0: z = -1.82                     Prob > |z| = 0.0688
Test of homogeneity: Q = chi2(3) = 3.62            Prob > Q = 0.3049
```

After the original declaration, we can use `meta update` to update the meta settings instead of repeating `meta set`; see example 6.

◁

Also see *Remarks and examples* in [META] **meta set** for more examples of using `meta set`.

Computing and declaring effect sizes using meta esize

We demonstrate how to use `meta esize` to compute and declare effect sizes for continuous and binary outcomes.

▷ Example 2: Mean differences for two-sample continuous data using meta esize

Consider the study of Gibson et al. (2002), who compared the performance of asthma-management programs for adults with asthma.

The asthma dataset contains the following summary-data variables:

```
use https://www.stata-press.com/data/r18/asthma, clear
(Education and medical review for asthma patients)
. describe ni meani sdi nc meanc sdc
```

Variable name	Storage type	Display format	Value label	Variable label
ni	int	%9.0g		Intervention-group sample size
meani	double	%9.0g		Average days off work/school for intervention group
sdi	double	%9.0g		Std. dev. of days off work/school for intervention group
nc	int	%9.0g		Control-group sample size
meanc	double	%9.0g		Average days off work/school for control group
sdc	double	%9.0g		Std. dev. of days off work/school for control group

Variables ni, meani, and sdi record the study-specific sample sizes, mean numbers of days off work/school, and standard deviations in the intervention group, and variables nc, meanc, and sdc record those items in the control group.

To illustrate, we will compute and declare a couple of effect sizes using meta esize. We will start with the default effect size—Hedges's g standardized mean. We use meta esize to compute this effect size for each study from the summary variables and declare them for further meta-analysis.

```
. meta esize ni meani sdi nc meanc sdc
(2 missing values generated)
Meta-analysis setting information
 Study information
    No. of studies: 13
       Study label: Generic
        Study size: _meta_studysize
      Summary data: ni meani sdi nc meanc sdc
       Effect size
             Type: hedgesg
            Label: Hedges's g
         Variable: _meta_es
   Bias correction: Approximate
         Precision
         Std. err.: _meta_se
    Std. err. adj.: None
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

There are missing values in the summary variables, so some of the generated system variables will also contain missing values as reported by the note.

meta esize reports that the computed effect size is Hedges's g. See *Meta settings with meta esize* for the explanation of other settings.

The above command is equivalent to

```
. meta esize ni meani sdi nc meanc sdc, esize(hedgesg)
  (output omitted)
```

With this effect size, we can specify that the adjustment of Hedges and Olkin (1985) be applied to the standard errors.

```
. meta esize ni meani sdi nc meanc sdc, esize(hedgesg, holkinse)
(2 missing values generated)

Meta-analysis setting information

 Study information
    No. of studies: 13
       Study label: Generic
        Study size: _meta_studysize
      Summary data: ni meani sdi nc meanc sdc

       Effect size
             Type: hedgesg
            Label: Hedges's g
         Variable: _meta_es
   Bias correction: Approximate

        Precision
        Std. err.: _meta_se
    Std. err. adj.: Hedges-Olkin
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%

  Model and method
            Model: Random effects
           Method: REML
```

`meta esize` updates the adjustment in `Std. err. adj.:` under `Precision` to `Hedges-Olkin`.

Because all studies measured our outcome of interest on the same scale (number of days off work or school), we may consider the raw (unstandardized) mean difference as our effect size. We can compute it by specifying the `esize(mdiff)` option.

```
. meta esize ni meani sdi nc meanc sdc, esize(mdiff)
(2 missing values generated)

Meta-analysis setting information

 Study information
    No. of studies: 13
       Study label: Generic
        Study size: _meta_studysize
      Summary data: ni meani sdi nc meanc sdc

       Effect size
             Type: mdiff
            Label: Mean diff.
         Variable: _meta_es

        Precision
        Std. err.: _meta_se
    Std. err. adj.: None
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%

  Model and method
            Model: Random effects
           Method: REML
```

The information about the type of the effect size and its label is updated to correspond to the mean differences.

As with meta set, we could have used meta update to update the meta settings after the initial declaration instead of using meta esize; see example 6.

◁

▷ Example 3: Log odds-ratios and log risk-ratios for two-sample binary data

Let's revisit the declaration we used in example 1 in [META] **meta** for the bcg dataset from the BCG vaccine study (Colditz et al. 1994). The summary data (contingency tables) are recorded in the following variables:

```
use https://www.stata-press.com/data/r18/bcg, clear
(Efficacy of BCG vaccine against tuberculosis)
. describe npost nnegt nposc nnegc
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
npost           int     %9.0g                 Number of TB positive cases in
                                                treated group
nnegt           long    %9.0g                 Number of TB negative cases in
                                                treated group
nposc           int     %9.0g                 Number of TB positive cases in
                                                control group
nnegc           long    %9.0g                 Number of TB negative cases in
                                                control group
```

The summary variables represent the cells of the 2×2 tables for each study.

As with continuous data, we specify the summary variables for binary data following meta esize:

```
. meta esize npost nnegt nposc nnegc
Meta-analysis setting information

 Study information
    No. of studies: 13
       Study label: Generic
        Study size: _meta_studysize
      Summary data: npost nnegt nposc nnegc

       Effect size
             Type: lnoratio
            Label: Log odds-ratio
         Variable: _meta_es
   Zero-cells adj.: None; no zero cells

         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%

  Model and method
             Model: Random effects
            Method: REML
```

The computed default effect sizes are log odds-ratios, whereas the effect of interest in this study is the risk ratio or, equivalently, the log risk-ratio.

To compute log risk-ratios, we specify `esize(lnrratio)`. We also specify the variable `studylbl` containing study labels in the `studylabel()` option.

```
. meta esize npost nnegt nposc nnegc, esize(lnrratio) studylabel(studylbl)
Meta-analysis setting information
 Study information
    No. of studies: 13
       Study label: studylbl
        Study size: _meta_studysize
      Summary data: npost nnegt nposc nnegc
       Effect size
             Type: lnrratio
            Label: Log risk-ratio
         Variable: _meta_es
   Zero-cells adj.: None; no zero cells
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

Notice that there are no zero cells in our data, so there is no zero-cells adjustment (see `Zero-cells adj.:` under `Effect size`).

◁

Also see example 6 for how to update the above `meta` settings without having to respecify the summary variables.

▷ Example 4: Freeman–Tukey-transformed proportions for one-sample binary data

We will consider a dataset from Miller (1978) to illustrate `meta esize` to set up data for meta-analysis of a single proportion. The summary data consist of the number of animal deaths (`ndeaths`) and the animal pen size (`pensize`):

```
use https://www.stata-press.com/data/r18/miller1978, clear
(Number of animal deaths)
. describe
Contains data from https://www.stata-press.com/data/r18/miller1978.dta
 Observations:             4                  Number of animal deaths
    Variables:             2                  15 Mar 2022 17:04
                                              (_dta has notes)
------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
ndeaths         byte    %9.0g                 Number of animal deaths
pensize         byte    %9.0g                 Size of animal pen
------------------------------------------------------------------------------
Sorted by:
```

As with two-group comparisons of continuous and binary data, we specify the summary variables following `meta esize`:

```
. meta esize ndeaths pensize
Meta-analysis setting information
 Study information
    No. of studies: 4
       Study label: Generic
        Study size: _meta_studysize
     Summary data: ndeaths pensize
       Effect size
             Type: ftukeyprop
            Label: Freeman-Tukey's p
         Variable: _meta_es
       Precision
       Std. err.: _meta_se
              CI: [_meta_cil, _meta_ciu]
        CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

By default, `meta esize` computes the Freeman–Tukey-transformed proportions. For meta-analysis using the untransformed (raw) proportions, you can specify the `esize(proportion)` option:

```
. meta esize ndeaths pensize, esize(proportion)
Meta-analysis setting information
 Study information
    No. of studies: 4
       Study label: Generic
        Study size: _meta_studysize
     Summary data: ndeaths pensize
       Effect size
             Type: proportion
            Label: Proportion
         Variable: _meta_es
   Zero-cells adj.: None; no zero cells
       Precision
       Std. err.: _meta_se
              CI: [_meta_cil, _meta_ciu]
        CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

Unlike the Freeman–Tukey-transformed proportions, in the presence of zero successes or failures, the raw proportions require zero-cell adjustments; otherwise, their variances will be undefined. In our example, there are no zero cells, so no zero-cells adjustment was applied (see `Zero-cells adj.:` under `Effect size`).

We could have also used `meta update` to modify the above `meta` settings without having to respecify the summary variables; see example 6.

◁

▷ Example 5: Effect sizes for correlation data (StataNow)

Molloy, O'Carroll, and Ferguson (2013) conducted a meta-analysis to examine to what degree conscientiousness is related to medication adherence. Medication adherence is the extent to which typically chronically ill patients follow medical recommendations as prescribed. Conscientiousness is defined as "*socially prescribed impulse control* that facilitates task- and goal-directed behavior, such as thinking before acting, delaying gratification, following norms and rules, and planning, organizing and prioritizing tasks" (John and Srivastava 1999, 121).

The dataset contains the variables `studylbl`, `rho`, and `n` to indicate the authors and year of publication of the studies, the correlation coefficient between conscientiousness and medication adherence, and the study sample size, respectively.

```
. use https://www.stata-press.com/data/r18/adherence
(Conscientiousness and medication adherence)
. describe rho n studylbl
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
rho             double  %9.0g               * Correlation coefficient
n               int     %9.0g                 Sample size of the study
studylbl        str26   %26s                  Study label
```

As with one-sample binary data, we specify two summary variables, `rho` and `n`, following `meta esize`. To compute Fisher's z-transformed correlations, we specify the `fisherz` option:

```
. meta esize rho n, fisherz
Meta-analysis setting information
 Study information
    No. of studies: 16
       Study label: Generic
        Study size: _meta_studysize
      Summary data: rho n
       Effect size
             Type: fisherz
            Label: Fisher's z
         Variable: _meta_es
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

For meta-analysis using the untransformed (raw) correlations, you can specify the `correlation` option:

```
. meta esize rho n, correlation
Meta-analysis setting information
 Study information
    No. of studies: 16
       Study label: Generic
        Study size: _meta_studysize
      Summary data: rho n
       Effect size
             Type: correlation
            Label: Correlation
         Variable: _meta_es
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

We could have also used `meta update` to modify the above `meta` settings without having to respecify the summary variables; see example 6.

◁

Displaying and updating meta settings

We show examples of how to display the current `meta` settings by using `meta query` and update them by using `meta update`.

▷ Example 6: Commands meta query and meta update

Recall example 3. Let's reload the dataset and use `meta query` to check whether the dataset is `meta` set.

```
. use https://www.stata-press.com/data/r18/bcg, clear
(Efficacy of BCG vaccine against tuberculosis)
. meta query
(data not meta set; use meta set or meta esize to declare as meta data)
```

The data are not `meta` set.

Let's again use `meta esize` to declare the data (quietly) and use `meta query` to display the current settings.

```
. quietly meta esize npost nnegt nposc nnegc
. meta query
-> meta esize npost nnegt nposc nnegc
Meta-analysis setting information from meta esize
 Study information
    No. of studies: 13
       Study label: Generic
        Study size: _meta_studysize
      Summary data: npost nnegt nposc nnegc
       Effect size
             Type: lnoratio
            Label: Log odds-ratio
         Variable: _meta_es
   Zero-cells adj.: None; no zero cells
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

In example 3, we redeclared the data to use the log risk-ratios as effect sizes. After the initial declaration, it is more convenient to use `meta update` to update the meta settings because we do not need to respecify the summary variables with `meta update`.

```
. meta update, esize(lnrratio) studylabel(studylbl)
-> meta esize npost nnegt nposc nnegc , esize(lnrratio) studylabel(studylbl)
Meta-analysis setting information from meta esize
 Study information
    No. of studies: 13
       Study label: studylbl
        Study size: _meta_studysize
      Summary data: npost nnegt nposc nnegc
       Effect size
             Type: lnrratio
            Label: Log risk-ratio
         Variable: _meta_es
   Zero-cells adj.: None; no zero cells
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

If your summary variables change, however, you must specify them with `meta esize`.

You can use `meta update` after either `meta esize` or `meta set`. `meta update` will respect the options of `meta esize` and `meta set`.

For example, recall the `meta set` declaration from example 1:

```
. use https://www.stata-press.com/data/r18/leukemia2, clear
(Single-agent purine analogue treatment for leukemia)
. quietly meta set loghr seloghr
```

Let's update the meta settings to include the variable containing study sizes.

```
. generate ssize = ntreat + ncontrol
. meta update, studysize(ssize)
-> meta set loghr seloghr , random(reml) studysize(ssize)
Meta-analysis setting information from meta set
 Study information
    No. of studies: 4
       Study label: Generic
        Study size: ssize
       Effect size
             Type: <generic>
            Label: Effect size
         Variable: loghr
         Precision
         Std. err.: seloghr
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

The `studysize()` option is supported only with `meta set`. If we tried to specify this option with `meta update` after `meta esize`, we would have received an error message.

◁

References

Colditz, G. A., T. F. Brewer, C. S. Berkey, M. E. Wilson, E. Burdick, H. V. Fineberg, and F. Mosteller. 1994. Efficacy of BCG vaccine in the prevention of tuberculosis: Meta-analysis of the published literature. *Journal of the American Medical Association* 271: 698–702. https://doi.org/10.1001/jama.1994.03510330076038.

Gibson, P., H. Powell, A. Wilson, M. J. Abramson, P. Haywood, A. Bauman, M. J. Hensley, E. H. Walters, and J. J. L. Roberts. 2002. Self-management education and regular practitioner review for adults with asthma. *Cochrane Database of Systematic Reviews* 3. https://doi.org/10.1002/14651858.CD001117.

Hedges, L. V., and I. Olkin. 1985. *Statistical Methods for Meta-Analysis*. Orlando, FL: Academic Press.

John, O. P., and S. Srivastava. 1999. "The big five trait taxonomy: History, measurement, and theoretical perspectives". In *Handbook of Personality: Theory and Research*, edited by L. A. Pervin and O. P. John, 102–138. 2nd ed. New York: Guilford.

Miller, J. J. 1978. The inverse of the Freeman–Tukey double arcsine transformation. *American Statistician* 32: 138. https://doi.org/10.1080/00031305.1978.10479283.

Molloy, G. J., R. E. O'Carroll, and E. Ferguson. 2013. Conscientiousness and medication adherence: A meta-analysis. *Annals of Behavioral Medicine* 47: 92–101. https://doi.org/10.1007/s12160-013-9524-4.

Steurer, M., G. Pall, S. Richards, G. Schwarzer, J. Bohlius, and R. Greil. 2006. Single-agent purine analogues for the treatment of chronic lymphocytic leukaemia: A systematic review and meta-analysis. *Cancer Treatment Reviews* 32: 377–389. https://doi.org/10.1016/j.ctrv.2006.05.002.

Sweeting, M. J., A. J. Sutton, and P. C. Lambert. 2004. What to add to nothing? Use and avoidance of continuity corrections in meta-analysis of sparse data. *Statistics in Medicine* 23: 1351–1375. https://doi.org/10.1002/sim.1761.

Also see

[META] **meta esize** — Compute effect sizes and declare meta-analysis data+

[META] **meta set** — Declare meta-analysis data using generic effect sizes

[META] **meta update** — Update, describe, and clear meta-analysis settings

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta esize — Compute effect sizes and declare meta-analysis data[+]

[+]This command includes features that are part of StataNow.

Description Quick start Menu Syntax
Options Remarks and examples Stored results Methods and formulas
References Also see

Description

`meta esize` computes effect sizes from study summary data and uses the results to declare the data in memory to be `meta` data, informing Stata of key variables and their roles in a meta-analysis. It computes various effect sizes and their respective standard errors for two-group comparisons of continuous and binary outcomes and for estimating a single proportion of a binary outcome or, in StataNow, a correlation between two variables. It then uses the computed effect sizes and standard errors to declare the data in memory to be `meta` data. If you do not have the summary data from individual studies and, instead, you have precalculated effect sizes, you can use `meta set` to declare your meta-analysis data. You must use `meta esize` or `meta set` to perform univariate meta-analysis using the `meta` command; see [META] **meta data**.

If you need to update some of the meta settings after the data declaration, see [META] **meta update**. To display current meta settings, use `meta query`; see [META] **meta update**.

Quick start

Compute Hedges's g standardized mean differences and their standard errors from variables `nt` (sample size in treatment group), `meant` (mean of treatment group), `sdt` (standard deviation in treatment group), and their counterparts in the control group: `nc`, `meanc`, and `sdc`

```
meta esize nt meant sdt nc meanc sdc
```

Same as above, but compute Cohen's d instead of the default Hedges's g, and use the DerSimonian–Laird estimation method instead of the default REML method

```
meta esize nt meant sdt nc meanc sdc, esize(cohend) random(dlaird)
```

Compute log odds-ratios and their standard errors from variables `nst` (number of successes in treatment group), `nft` (number of failures in treatment group), and their respective counterparts in control group: `nsc` and `nfc`

```
meta esize nst nft nsc nfc
```

Same as above, but compute the log risk-ratios instead of the default log odds-ratios

```
meta esize nst nft nsc nfc, esize(lnrratio)
```

Same as above, but request a common-effect meta-analysis

```
meta esize nst nft nsc nfc, esize(lnrratio) common
```

Compute transformed proportions using the default Freeman–Tukey double-arcsine transformation and their standard errors from variables `ns` (number of successes) and `n` (study sample size)

```
meta esize ns n
```

Same as above, but compute the logit-transformed proportions instead of the default Freeman–Tukey-transformed proportions

```
meta esize ns n, esize(logitprop)
```

Compute Fisher's z-transformed correlations and their standard errors from variables `r` (correlation) and `n` (study sample size) (StataNow)

```
meta esize r n, fisherz
```

Same as above, but use the untransformed (raw) correlations and compute their standard errors (StataNow)

```
meta esize r n, correlation
```

Menu

Statistics > Meta-analysis

Syntax

Compute and declare effect sizes for two-group comparison of continuous outcomes

```
meta esize n1 mean1 sd1 n2 mean2 sd2 [if] [in] [, options_continuous options]
```

Compute and declare effect sizes for two-group comparison of binary outcomes

```
meta esize n11 n12 n21 n22 [if] [in] [, options_binary options]
```

Compute and declare effect sizes for estimating a single proportion (prevalence)

```
meta esize ns n [if] [in] [, options_proportion options]
```

Compute and declare effect sizes for estimating a correlation (StataNow)

```
meta esize r n [if] [in] , {fisherz|correlation} [options_correlation options]
```

Variables *n1*, *mean1*, and *sd1* contain sample sizes, means, and standard deviations from individual studies for group 1 (treatment), and variables *n2*, *mean2*, and *sd2* contain the respective summaries for group 2 (control).

Variables *n11* and *n12* contain numbers of successes and numbers of failures from individual studies for group 1 (treatment), and variables *n21* and *n22* contain the respective numbers for group 2 (control). A single observation defined by variables *n11*, *n12*, *n21*, and *n22* represents a 2×2 table from an individual study. Therefore, variables *n11*, *n12*, *n21*, and *n22* represent a sample of 2×2 tables from all studies. We will thus refer to observations on these variables as 2×2 tables and to values of these variables as cells.

Variables *ns* and *n* contain number of successes and sample sizes from individual studies. Here, the values of the variable *ns* (number of successes) and the values of an implicit "variable" $n - ns$ (number of failures) are referred to as cells.

Variables *r* and *n* contain correlations and sample sizes from individual studies.

options_continuous	Description
Main	
`esize(`*esspeccnt*`)`	specify effect size for two-group comparison of continuous outcomes to be used in the meta-analysis
Model	
`random[(`*remethod*`)]`	random-effects meta-analysis; default is `random(reml)`
`common`	common-effect meta-analysis; implies inverse-variance method
`fixed`	fixed-effects meta-analysis; implies inverse-variance method

options_binary	Description
Main	
`esize(`*estypebin*`)`	specify effect size for two-group comparison of binary outcomes to be used in the meta-analysis
Model	
`random[(`*remethod*`)]`	random-effects meta-analysis; default is `random(reml)`
`common[(`*cefemethod*`)]`	common-effect meta-analysis
`fixed[(`*cefemethod*`)]`	fixed-effects meta-analysis
Options	
`zerocells(`*zcspec*`)`	adjust for zero cells during computation; default is to add 0.5 to all cells of those 2×2 tables that contain zero cells

options_proportion	Description
Main	
`esize(`*estypeprop*`)`	specify effect size for estimating a single proportion to be used in the meta-analysis
Model	
`random[(`*remethod*`)]`	random-effects meta-analysis; default is `random(reml)`
`common`	common-effect meta-analysis; implies inverse-variance method
`fixed`	fixed-effects meta-analysis; implies inverse-variance method
Options	
`zerocells(`*zcspec*`)`	adjust for zero cells during computation; default is to add 0.5 to all cells of studies with zero successes or failures

options_correlation	Description
Main	
+* `fisherz`	Fisher's z-transformed correlation
+* `correlation`	untransformed (raw) correlation
+ `esize(`*estypecorr*`)`	synonym for `fisherz` or `correlation`
Model	
`random[(`*remethod*`)]`	random-effects meta-analysis; default is `random(reml)`
`common`	common-effect meta-analysis; implies inverse-variance method
`fixed`	fixed-effects meta-analysis; implies inverse-variance method

+These features are part of StataNow.

* Either `fisherz` or `correlation` is required.

options	Description
Options	
`studylabel(`*varname*`)`	variable to be used to label studies in all meta-analysis output
`eslabel(`*string*`)`	effect-size label to be used in all meta-analysis output; default is `eslabel(Effect size)`
`level(`#`)`	confidence level for all subsequent meta-analysis commands
[`no`]`metashow`	display or suppress meta settings with other `meta` commands

The syntax of *esspeccnt* is

estypecnt [, *esopts*]

estypecnt	Description
`hedgesg`	Hedges's g standardized mean difference; the default
`cohend`	Cohen's d standardized mean difference
`glassdelta2`	Glass's Δ mean difference standardized by group 2 (control) standard deviation; more common than `glassdelta1`
`glassdelta1`	Glass's Δ mean difference standardized by group 1 (treatment) standard deviation
`mdiff`	(unstandardized) mean difference

estypebin	Description
`lnoratio`	log odds-ratio; the default
`lnrratio`	log risk-ratio (also known as log rate-ratio and log relative-risk)
`rdiff`	risk difference
`lnorpeto`	Peto's log odds-ratio

estypeprop	Description
`ftukeyprop`	Freeman–Tukey-transformed proportion; the default
`logitprop`	logit-transformed proportion
`proportion`	untransformed (raw) proportion

estypecorr	Description
+`fisherz`	Fisher's z-transformed correlation
+`correlation`	untransformed (raw) correlation

+These features are part of StataNow.

remethod	Description
`reml`	restricted maximum likelihood; the default
`mle`	maximum likelihood
`ebayes`	empirical Bayes
`dlaird`	DerSimonian–Laird
`sjonkman`	Sidik–Jonkman
`hedges`	Hedges
`hschmidt`	Hunter–Schmidt

cefemethod	Description
mhaenszel	Mantel–Haenszel
invvariance	inverse variance
ivariance	synonym for invvariance

Options

Main

esize(*esspec*) specifies the effect size to be used in the meta-analysis. For a two-group comparison of continuous outcomes, *esspec* is *estypecnt* [, *esopts*]. For binary outcomes, *esspec* is *estypebin* for a two-group comparison or *estypeprop* for estimating a single proportion (prevalence). For correlation data, *esspec* is *estypecorr*.

For a two-group comparison of continuous outcomes, *estypecnt* is one of the following: hedgesg, cohend, glassdelta2, glassdelta1, or mdiff. Below, we describe each type with its specific options, *esopts*.

hedgesg [, exact holkinse] computes the effect size as the Hedges's g (1981) standardized mean difference. This is the default. For consistency with meta-analysis literature, hedgesg uses an approximation to compute g rather than the exact computation (see *Methods and formulas*), as provided by esize's option hedgesg. You can use the exact suboption to match the results from esize (see [R] **esize**).

cohend [, holkinse] computes the effect size as the Cohen's d (1969, 1988) standardized mean difference.

glassdelta2 computes the effect size as the Glass's Δ standardized mean difference, where the standardization uses the standard deviation of the group 2 (control group). glassdelta2 is more common in practice than glassdelta1.

glassdelta1 computes the effect size as the Glass's Δ standardized mean difference, where the standardization uses the standard deviation of the group 1 (treatment group). glassdelta2 is more common in practice than glassdelta1.

mdiff [, unequal] computes the effect size as the unstandardized or raw mean difference.

esopts are exact, holkinse, and unequal.

exact specifies that the exact computation be used for the bias-correction factor in Hedges's g instead of an approximation used by default.

holkinse specifies that the standard error of Hedges's g and Cohen's d be computed as described in Hedges and Olkin (1985). This is another approximation to the standard error of these effect sizes sometimes used in practice.

unequal specifies that the computation of the standard error of the mean difference (esize(mdiff)) assume unequal group variances.

For a two-group comparison of binary outcomes, *estypebin* is one of the following: lnoratio, lnrratio, rdiff, or lnorpeto.

lnoratio specifies that the effect size be the log odds-ratio. This is the default.

lnrratio specifies that the effect size be the log risk-ratio, also known as a log relative-risk and a log risk-rate.

rdiff specifies that the effect size be the risk difference.

lnorpeto specifies that the effect size be the log odds-ratio as defined by Peto et al. (1977). This effect size is preferable with rare events.

For estimating a proportion from one-sample binary or prevalence data, *estypeprop* is one of the following: ftukeyprop, logitprop, or proportion.

ftukeyprop specifies that the effect size be the Freeman–Tukey-transformed proportion (Freeman and Tukey 1950). This is the default. The Freeman–Tukey transformation is a variance-stabilizing transformation and is preferable when the estimated proportions are close to 0 or 1. This effect size does not require a zero-cell adjustment (continuity correction) for studies with zero successes or failures.

logitprop specifies that the effect size be the logit-transformed proportion. When a study proportion is close to 0 or 1, the estimated variance of this effect size is very large, and thus the study is assigned an artificially small weight in the meta-analysis.

proportion specifies that the effect size be the untransformed (raw) proportion. When a study proportion is close to 0 or 1, its estimated variance is very small, and thus the study is assigned an artificially large weight in the meta-analysis. Moreover, the study confidence limits may fall outside the $[0, 1]$ range.

For correlation data, *estypecorr* is one of fisherz or correlation. These options are part of StataNow.

fisherz specifies that the effect size be the Fisher's z-transformed correlation.

correlation specifies that the effect size be the untransformed (raw) correlation.

For effect sizes in the log metric such as log odds-ratios, the results by default are displayed in the log metric. You can use *eform_option* of meta summarize to obtain exponentiated results such as odds ratios. For effect sizes ftukeyprop and logitprop, the results by default are displayed in the respective Freeman–Tukey and logit metrics. You can use options transform() and proportion of meta summarize to report results as proportions. For effect size fisherz, the results are displayed as Fisher's z-transformed correlations. You can use options transform() and correlation of meta summarize to report results as correlations.

Model

Options random(), common(), and fixed() declare the meta-analysis model globally throughout the entire meta-analysis; see *Declaring a meta-analysis model* in [META] **meta data**. In other words, once you set your meta-analysis model using meta esize, all subsequent meta commands will assume that same model. You can update the declared model by using meta update or change it temporarily by specifying the corresponding option with the meta commands. Options random(), common(), and fixed() may not be combined. If these options are omitted, random(reml) is assumed; see *Default meta-analysis model and method* in [META] **meta data**. Also see *Meta-analysis models* in [META] **Intro**.

random and random(*remethod*) specify that a random-effects model be assumed for meta-analysis; see *Random-effects model* in [META] **Intro**.

remethod specifies the type of estimator for the between-study variance τ^2. *remethod* is one of reml, mle, ebayes, dlaird, sjonkman, hedges, or hschmidt. random is a synonym for random(reml). Below, we provide a short description for each method based on Veroniki et al. (2016). Also see *Declaring a meta-analysis estimation method* in [META] **meta data**.

`reml`, the default, specifies that the REML method (Raudenbush 2009) be used to estimate τ^2. This method produces an unbiased, nonnegative estimate of the between-study variance and is commonly used in practice. Method `reml` requires iteration.

`mle` specifies that the ML method (Hardy and Thompson 1996) be used to estimate τ^2. It produces a nonnegative estimate of the between-study variance. With a few studies or small studies, this method may produce biased estimates. With many studies, the ML method is more efficient than the REML method. Method `mle` requires iteration.

`ebayes` specifies that the empirical Bayes estimator (Berkey et al. 1995), also known as the Paule–Mandel estimator (Paule and Mandel 1982), be used to estimate τ^2. From simulations, this method, in general, tends to be less biased than other random-effects methods, but it is also less efficient than `reml` or `dlaird`. Method `ebayes` produces a nonnegative estimate of τ^2 and requires iteration.

`dlaird` specifies that the DerSimonian–Laird method (DerSimonian and Laird 1986) be used to estimate τ^2. This method, historically, is one of the most popular estimation methods because it does not make any assumptions about the distribution of random effects and does not require iteration. But it may underestimate the true between-study variance, especially when the variability is large and the number of studies is small. This method may produce a negative value of τ^2 and is thus truncated at zero in that case.

`sjonkman` specifies that the Sidik–Jonkman method (Sidik and Jonkman 2005) be used to estimate τ^2. This method always produces a nonnegative estimate of the between-study variance and thus does not need truncating at 0, unlike the other noniterative methods. Method `sjonkman` does not require iteration.

`hedges` specifies that the Hedges method (Hedges 1983) be used to estimate τ^2. When the sampling variances of effect-size estimates can be estimated without bias, this estimator is exactly unbiased (before truncation), but it is not widely used in practice (Veroniki et al. 2016). Method `hedges` does not require iteration.

`hschmidt` specifies that the Hunter–Schmidt method (Schmidt and Hunter 2015) be used to estimate τ^2. Although this estimator achieves a lower MSE than other methods, except ML, it is known to be negatively biased. Method `hschmidt` does not require iteration.

`common` specifies that a common-effect model be assumed for meta-analysis; see *Common-effect ("fixed-effect") model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about common-effect versus fixed-effects models.

`common` and `common(`*cefemethod*`)` specify that a common-effect model be assumed for meta-analysis; see *Common-effect ("fixed-effect") model* in [META] **Intro**. Also see the discussion in [META] **meta data** about common-effect versus fixed-effects models.

`common` implies `common(mhaenszel)` for effect sizes `lnoratio`, `lnrratio`, and `rdiff` and `common(invvariance)` for all other effect sizes.

cefemethod is one of `mhaenszel` or `invvariance` (synonym `ivariance`). Below, we provide a short description for each method. Also see *Declaring a meta-analysis estimation method* in [META] **meta data**.

mhaenszel is available only for a two-group comparison of binary outcomes. It specifies a meta-analysis using the Mantel–Haenszel method to estimate the overall effect size. This method is the default for effect sizes lnoratio, lnrratio, and rdiff but is not available for effect size lnorpeto.

invvariance specifies a meta-analysis using the inverse-variance method to estimate the overall effect size. This method is available for all types of analyses and effect sizes. It is the default for a two-group comparison of continuous outcomes, for a two-group comparison of binary outcomes using effect size lnorpeto, for correlation data, and for estimating a single proportion (or prevalence).

ivariance is a synonym for invvariance.

fixed and fixed(*cefemethod*) specify that a fixed-effects model be assumed for meta-analysis; see *Fixed-effects model* in [META] **Intro**. Also see the discussion in [META] **meta data** about fixed-effects versus common-effect models.

fixed implies fixed(mhaenszel) for effect sizes lnoratio, lnrratio, and rdiff and fixed(invvariance) for all other effect sizes.

cefemethod is one of mhaenszel or invvariance (synonym ivariance); see descriptions above.

fixed specifies that a fixed-effects model be assumed for meta-analysis; see *Fixed-effects model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about fixed-effects versus common-effect models.

Options

zerocells(*zcspec*) is for use with binary outcomes when the effect size is either lnoratio or lnrratio for the two-sample case or either logitprop or proportion for the one-sample case. It specifies the adjustment to be used for the cells in the presence of zero cells. The cells are the values of variables *n11*, *n12*, *n21*, and *n22* for the two-sample case and the number of successes and the number of failures for the one-sample case. The adjustment is applied during computation—the original data are not modified. The default is zerocells(0.5, only0); it adds 0.5 to all cells of studies with at least one zero cell. To request no adjustment, specify zerocells(none). More generally, the syntax of *zcspec* is

[, *zcadj*]

where # is the adjustment value, also known as the continuity-correction value in the meta-analysis literature, and *zcadj* is only0 or allif0.

only0 specifies that # be added to all cells of only those studies with at least one zero cell. For the two-sample case, during computation, # is added to each observation defined by variables *n11*, *n12*, *n21*, and *n22* if that observation contains a value of zero in any of those variables. For the one-sample case, # is added to all values (cells) corresponding to zero successes and to zero failures.

allif0 specifies that # be added to all cells of all studies but only if there is at least one study with a zero cell. For the two-sample case, during computation, # is added to all values of variables *n11*, *n12*, *n21*, and *n22* but only if there is a zero value in one of the four variables. For the one-sample case, # is added to all cells (number of successes and number of failures) if at least one study contains zero successes or zero failures.

For the effect size `lnoratio`, *zcspec* may also be `tacc`, which implements the treatment-arm continuity correction of Sweeting, Sutton, and Lambert (2004). This method estimates the group-specific adjustment values from the data to minimize the bias of the overall odds-ratio estimator in the presence of zero cells. This method is recommended when the groups are unbalanced.

`studylabel(`*varname*`)` specifies a string variable containing labels for the individual studies to be used in all applicable meta-analysis output. The default study labels are `Study 1`, `Study 2`, ..., `Study` K, where K is the total number of studies in the meta-analysis.

`eslabel(`*string*`)` specifies that *string* be used as the effect-size label in all relevant meta-analysis output. The default label is `Effect size`.

`level(`#`)` specifies the confidence level, as a percentage, for confidence intervals. It will be used by all subsequent meta-analysis commands when computing confidence intervals. The default is `level(95)` or as set by `set level`; see [R] **level**. After the declaration, you can specify `level()` with `meta update` to update the confidence level to be used throughout the rest of the meta-analysis session. You can also specify `level()` directly with the `meta` commands to modify the confidence level, temporarily, during the execution of the command.

`metashow` and `nometashow` display or suppress the meta setting information in the output of other `meta` commands. By default, this information is displayed at the top of their output. You can also specify `nometashow` with `meta update` to suppress the meta setting output for the entire meta-analysis session after the declaration.

Remarks and examples

Remarks are presented under the following headings:

Meta-analysis for two-group comparison of binary outcomes
Meta-analysis for two-group comparison of continuous outcomes
Meta-analysis for estimating a single proportion
Meta-analysis for correlation data (StataNow)

`meta esize` computes various effect sizes, their standard errors, and CIs for continuous and binary outcomes from the summary data provided for each study. It then declares the computed effect-size data as the `meta` data; see [META] **meta data**. Different types of effect sizes may be specified in the `esize()` option. They depend on the type of analysis and outcome, so we describe them separately for various situations below, together with other data-specific options. Also see *Declaring meta-analysis information* in [META] **meta data**.

Meta-analysis for two-group comparison of binary outcomes

Meta-analysis is often used with studies comparing two groups. The first group is commonly referred to as the experimental or treatment group. The second group is commonly referred to as the control group.

For two-sample binary data, each study typically reports cell counts from the following 2×2 table.

group	success	failure	size
treatment	n_{11}	n_{12}	$n_1 = n_{11} + n_{12}$
control	n_{21}	n_{22}	$n_2 = n_{21} + n_{22}$

The cells of the table are composed of the numbers of "successes" and "failures" within each of the comparison groups. If a subject experiences an event of interest, it is a success; otherwise, it is a failure. Thus, the summary data for a two-group comparison of binary outcomes include the above 2×2 table for each study.

In this case, `meta esize` requires that four variables be specified containing the numbers of successes and failures in the treatment and control groups.

The goal of each study is to compare the probabilities of a success between the two groups. Various effect-size measures can be used for the comparison. For two-sample binary data, `meta esize` provides the following effect sizes: log odds-ratios (including Peto's log odds-ratios), the default; log risk-ratios; and risk differences. These are specified, respectively, as `lnoratio`, `lnorpeto`, `lnrratio`, and `rdiff` in the `esize()` option.

As described in *Declaring a meta-analysis model* in [META] **meta data**, you can choose between a random-effects, a fixed-effects, or a common-effect model. You can also choose from a number of estimation methods that are specific to the chosen model. For fixed-effects and common-effect models, in addition to the inverse-variance method, the Mantel–Haenszel method is available (and is the default) with effect sizes `lnoratio`, `lnrratio`, and `rdiff`; see *Declaring a meta-analysis estimation method* in [META] **meta data** and *Meta-analysis estimation methods* in [META] **Intro** for details.

Zero cell counts are known to create computational difficulties for odds ratios and risk ratios. A common solution is to add a small number, say, 0.5, to all cells of tables containing zero cells. This and other zero-cells adjustments are available in the `zerocells()` option.

Let's now look at several examples. Consider the following fictional meta-analysis dataset:

```
. use https://www.stata-press.com/data/r18/metaesbin
(Fictional data for binary outcomes)
. describe
Contains data from https://www.stata-press.com/data/r18/metaesbin.dta
 Observations:             4                  Fictional data for binary
                                                outcomes
    Variables:             5                  23 Apr 2022 12:14
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
study           str7    %9s                   Study label
tdead           byte    %9.0g                 Deaths in treatment group
tsurv           int     %9.0g                 Survivors in treatment group
cdead           byte    %9.0g                 Deaths in control group
csurv           int     %9.0g                 Survivors in control group
-------------------------------------------------------------------------------
Sorted by:
```

We will use this dataset to demonstrate how to compute effect sizes, specify different meta-analysis models, and adjust for zero cells with two-sample binary data.

▷ Example 1: A simple case

When working with meta-analysis data that do not have precomputed effect sizes, we can choose to compute effect sizes in a few different ways such as odds ratios and risk ratios. Using the simplest syntactical specification, we can compute the effect sizes, their standard errors, and the corresponding confidence intervals by specifying the number of successes and failures for one group, as well as the successes and failures for the second group, in that order.

```
. meta esize tdead tsurv cdead csurv
Meta-analysis setting information
 Study information
    No. of studies: 4
       Study label: Generic
        Study size: _meta_studysize
      Summary data: tdead tsurv cdead csurv
       Effect size
             Type: lnoratio
            Label: Log odds-ratio
         Variable: _meta_es
   Zero-cells adj.: 0.5, only0
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

The output indicates that there are 4 studies in the meta-analysis and, by default, a random-effects meta-analysis is to be assumed, where the heterogeneity parameter τ^2 is estimated via the REML method. The default computed effect size is the log odds-ratio. `meta esize` creates multiple system variables (see *System variables* in [META] **meta data**) that store the effect-size values, their standard errors, and the upper and lower limits of the CIs for the effect sizes.

We can now use, for example, `meta summarize` to list the individual log odds-ratios and the overall log odds-ratio, which is denoted as `theta`.

```
. meta summarize
  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Meta-analysis summary                     Number of studies =      4
Random-effects model                      Heterogeneity:
Method: REML                                          tau2 =  1.4417
                                                   I2 (%) =   69.33
                                                       H2 =    3.26

------------------------------------------------------------------------------
       Study | Log odds-ratio    [95% conf. interval]  % weight
-------------+----------------------------------------------------------------
     Study 1 |         -0.600      -2.079       0.879     27.80
     Study 2 |          0.351      -2.510       3.212     15.65
     Study 3 |          0.778      -0.031       1.586     34.69
     Study 4 |         -2.567      -4.638      -0.495     21.85
-------------+----------------------------------------------------------------
       theta |         -0.403      -1.869       1.063
------------------------------------------------------------------------------
Test of theta = 0: z = -0.54                 Prob > |z| = 0.5899
Test of homogeneity: Q = chi2(3) = 9.93       Prob > Q = 0.0192
```

See [META] **meta summarize** for details.

If we have a variable that stores the labels for each study, perhaps noting the study authors or journal, we can specify it in the `studylabel()` option with `meta esize`. Because we do not have such a variable in this dataset, each study is denoted generically by `Study #`. See example 4 in [META] **meta set** for an example of how to specify the study label and effect-size label.

◁

Example 2: Specify the effect size

The default is to compute the log odds-ratio for the effect size. To specify another metric, we can use the `esize()` option. For example, below we use the risk ratio (on the log scale) as our effect size by specifying `esize(lnrratio)`:

```
. meta esize tdead tsurv cdead csurv, esize(lnrratio)
Meta-analysis setting information
 Study information
    No. of studies: 4
       Study label: Generic
        Study size: _meta_studysize
      Summary data: tdead tsurv cdead csurv
       Effect size
             Type: lnrratio
            Label: Log risk-ratio
         Variable: _meta_es
   Zero-cells adj.: 0.5, only0
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

Example 3: Sparse data and adjustments for zero cells

Note that when we `list` the data, one of the studies has zero deaths.

```
. list tdead tsurv cdead csurv
```

	tdead	tsurv	cdead	csurv
1.	2	116	17	541
2.	0	15	15	682
3.	8	61	37	614
4.	1	421	9	291

By default, meta esize adds a constant value of 0.5 (that is, option zerocells(0.5, only0) is assumed) to each cell of a study that has a zero cell; see Zero-cells adj.: in the output of meta set in example 1. We can modify this adjustment by specifying a different constant factor. For example, we might add 0.003 to each zero cell:

```
. meta esize tdead tsurv cdead csurv, zerocells(.003)
Meta-analysis setting information
 Study information
    No. of studies: 4
       Study label: Generic
        Study size: _meta_studysize
     Summary data: tdead tsurv cdead csurv
       Effect size
             Type: lnoratio
            Label: Log odds-ratio
         Variable: _meta_es
   Zero-cells adj.: .003, only0
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

Or we may instead choose a different type of continuity correction, for example, the treatment-arm continuity correction (TACC), which we specify as zerocells(tacc):

```
. meta esize tdead tsurv cdead csurv, zerocells(tacc)
Meta-analysis setting information
 Study information
    No. of studies: 4
       Study label: Generic
        Study size: _meta_studysize
     Summary data: tdead tsurv cdead csurv
       Effect size
             Type: lnoratio
            Label: Log odds-ratio
         Variable: _meta_es
   Zero-cells adj.: tacc
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

Note that this option can be specified only when using the log odds-ratio as the effect size.

◁

▷ Example 4: Specify the meta-analysis model

In the examples above, we have been using the default random-effects model, but we could specify a different model. For example, we can use a common-effect model using the Mantel–Haenszel method to estimate the overall effect size:

```
. meta esize tdead tsurv cdead csurv, common(mhaenszel)
Meta-analysis setting information
 Study information
    No. of studies: 4
       Study label: Generic
        Study size: _meta_studysize
      Summary data: tdead tsurv cdead csurv
       Effect size
             Type: lnoratio
            Label: Log odds-ratio
         Variable: _meta_es
   Zero-cells adj.: 0.5, only0
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Common effect
            Method: Mantel-Haenszel
```

In the above, we could have specified simply `common` because the Mantel–Haenszel method is the default for a common-effect model with log odds-ratios.

◁

Meta-analysis for two-group comparison of continuous outcomes

We can also use meta-analysis to compare two groups for continuous outcomes. As before, the first group is commonly referred to as the experimental or treatment group, and the second group is commonly referred to as the control group.

For a two-group comparison of continuous outcomes, each study often reports the numbers of observations, means, and standard deviations in the two groups. Various effect sizes are then computed from these summary data for each study. Thus, to compute effect sizes for two-sample continuous data, `meta esize` requires that six variables be specified containing the numbers of observations, means, and standard deviations of the treatment and control groups. The supported effect sizes are the raw mean difference, `esize(mdiff)`, and standardized mean differences: Hedges's g, `esize(hedgesg)` (the default); Cohen's d, `esize(cohend)`; and Glass's Δs, `esize(glassdelta2)` and `esize(glassdelta1)`; see *Methods and formulas* for their definitions.

As described in *Declaring a meta-analysis model* in [META] **meta data**, you can choose between a random-effects, a fixed-effects, and a common-effect model. You can also choose from several estimation methods for random-effects models. Fixed-effects and common-effect models assume the inverse-variance estimation method. Also see *Declaring a meta-analysis estimation method* in [META] **meta data** and *Meta-analysis estimation methods* in [META] **Intro** for details.

Let's now demonstrate several usages of `meta esize` for a two-group comparison of continuous outcomes. Consider the following fictional meta-analysis dataset:

```
. use https://www.stata-press.com/data/r18/metaescnt, clear
(Fictional summary data for continuous outcomes)
. describe
Contains data from https://www.stata-press.com/data/r18/metaescnt.dta
 Observations:            10                  Fictional summary data for
                                                continuous outcomes
    Variables:             6                  19 Apr 2022 14:00
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
n1              byte    %9.0g                 Study sizes of group 1
m1              float   %9.0g                 Means of group 1
sd1             float   %9.0g                 Std. dev. of group 1
n2              byte    %9.0g                 Study sizes of group 2
m2              float   %9.0g                 Means of group 2
sd2             float   %9.0g                 Std. dev. of group 2
-------------------------------------------------------------------------------
Sorted by:
```

We will use this dataset to demonstrate different usages of the `meta esize` command with continuous-outcomes meta-analysis data.

▷ Example 5: The assumed model

In the simplest specification, `meta esize` requires that we specify the sample sizes, means, and standard deviations for each group in the meta-analysis.

```
. meta esize n1 m1 sd1 n2 m2 sd2
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: _meta_studysize
      Summary data: n1 m1 sd1 n2 m2 sd2
       Effect size
             Type: hedgesg
            Label: Hedges's g
         Variable: _meta_es
   Bias correction: Approximate
       Precision
        Std. err.: _meta_se
    Std. err. adj.: None
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

We see from the output that the Hedges's g standardized mean difference is used for the effect size, and, as for binary outcomes, a random-effects REML model is assumed. See *Meta settings with meta esize* in [META] **meta data** for a detailed description of all settings for this dataset.

◁

Example 6: Selecting an effect size

If we do not feel the need to standardize the mean differences, we could instead use the raw mean difference as the effect size by specifying `esize(mdiff)`.

```
. meta esize n1 m1 sd1 n2 m2 sd2, esize(mdiff)
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: _meta_studysize
      Summary data: n1 m1 sd1 n2 m2 sd2
       Effect size
             Type: mdiff
            Label: Mean diff.
         Variable: _meta_es
        Precision
        Std. err.: _meta_se
    Std. err. adj.: None
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

Example 7: Specifying different meta-analysis models and methods

Rather than using the default REML estimation method, we may want to use a different method, such as the DerSimonian–Laird method. We can specify this method in the `random()` option.

```
. meta esize n1 m1 sd1 n2 m2 sd2, random(dlaird)
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: _meta_studysize
      Summary data: n1 m1 sd1 n2 m2 sd2
       Effect size
             Type: hedgesg
            Label: Hedges's g
         Variable: _meta_es
   Bias correction: Approximate
        Precision
        Std. err.: _meta_se
    Std. err. adj.: None
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: DerSimonian-Laird
```

Or, instead of the random-effects model, we may specify a fixed-effects model, which implies the inverse-variance estimation method.

```
. meta esize n1 m1 sd1 n2 m2 sd2, fixed
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: _meta_studysize
      Summary data: n1 m1 sd1 n2 m2 sd2
       Effect size
             Type: hedgesg
            Label: Hedges's g
         Variable: _meta_es
   Bias correction: Approximate
        Precision
        Std. err.: _meta_se
    Std. err. adj.: None
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Fixed effects
           Method: Inverse-variance
```

Meta-analysis for estimating a single proportion

Meta-analysis is also used to estimate an overall proportion (or prevalence) from one-sample binary data by pooling proportions from single-arm studies whenever this is sensible.

The data contain the number of successes (or the number of events) and the study sample size for each study. (Success is a generic term and occurs when a subject experiences an event of interest.) To estimate a proportion, `meta esize` provides the following effect sizes: Freeman–Tukey-transformed proportions (the default), logit-transformed proportions, and untransformed (raw) proportions. These are specified, respectively, as `ftukeyprop`, `logitprop`, and `proportion` in the `esize()` option.

As described in *Declaring a meta-analysis model* in [META] **meta data**, you can choose between a random-effects, a fixed-effects, or a common-effect model. You can also choose from several estimation methods for random-effects models. Fixed-effects and common-effect models assume the inverse-variance estimation method. Also see *Declaring a meta-analysis estimation method* in [META] **meta data** and *Meta-analysis estimation methods* in [META] **Intro** for details.

Let's now look at several examples. Consider the following fictional meta-analysis dataset:

```
. use https://www.stata-press.com/data/r18/metaesprop, clear
(Fictional summary data to estimate proportion)
. describe
Contains data from https://www.stata-press.com/data/r18/metaesprop.dta
 Observations:             6                  Fictional summary data to
                                                estimate proportion
    Variables:             3                  26 Apr 2022 11:14
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
study           str7    %9s                   Study label
nsucc           byte    %9.0g                 Number of successes
ssize           int     %9.0g                 Study sample size
-------------------------------------------------------------------------------
Sorted by:
```

We will use this dataset to demonstrate different usages of the `meta esize` command to declare the data for meta-analysis of a single proportion.

▷ Example 8: The default setting

In its most basic form, `meta esize` requires that we specify the number of successes (`nsucc`) and the study sample sizes (`ssize`).

```
. meta esize nsucc ssize
Meta-analysis setting information
 Study information
    No. of studies: 6
       Study label: Generic
        Study size: _meta_studysize
      Summary data: nsucc ssize
       Effect size
             Type: ftukeyprop
            Label: Freeman-Tukey's p
         Variable: _meta_es
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

The output shows that the summary data are defined by variables `nsucc` and `ssize` and that, by default, the Freeman–Tukey-transformed proportion is used as the effect size. A random-effects REML model is assumed. Other settings are exactly as described in example 1.

◁

▷ Example 9: Specify the effect size

Instead of using the default Freeman–Tukey-transformed proportion, we can choose a different effect size, such as the logit-transformed proportion, using the `esize()` option.

```
. meta esize nsucc ssize, esize(logitprop)
Meta-analysis setting information
 Study information
    No. of studies: 6
       Study label: Generic
        Study size: _meta_studysize
      Summary data: nsucc ssize
       Effect size
             Type: logitprop
            Label: Logit proportion
         Variable: _meta_es
   Zero-cells adj.: None; no zero cells
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

The output differs from that in example 8 in the Effect size section. It now reflects that logit-transformed proportion is the effect size of choice instead of the default Freeman–Tukey-transformed proportion. There is also a new row for the zero-cells adjustment. This row did not show up in the output of example 8 because the Freeman–Tukey-transformed proportion does not need continuity correction. In our dataset, there are no zero cells, so the output in that row shows that no zero-cells adjustment was applied.

The logit-transformed proportion (and the untransformed proportion, esize(proportion)) should be avoided when there are study proportions that are close to 0 or 1.

◁

▷ Example 10: Specify an alternative meta-analysis model and method

Instead of using the default REML estimation method, you may specify an alternative random-effects method, such as the DL method. This can be done via the random() option.

```
. meta esize nsucc ssize, random(dlaird)
Meta-analysis setting information
 Study information
    No. of studies: 6
       Study label: Generic
        Study size: _meta_studysize
      Summary data: nsucc ssize
       Effect size
             Type: ftukeyprop
            Label: Freeman-Tukey's p
         Variable: _meta_es
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: DerSimonian-Laird
```

Or perhaps you believe that your proportions are similar across the studies and that a common-effect model is adequate to synthesize the overall proportion. You may request a common-effect model with the inverse-variance method by specifying the common option.

```
. meta esize nsucc ssize, common
Meta-analysis setting information
 Study information
    No. of studies: 6
       Study label: Generic
        Study size: _meta_studysize
      Summary data: nsucc ssize
       Effect size
             Type: ftukeyprop
            Label: Freeman-Tukey's p
         Variable: _meta_es
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Common effect
           Method: Inverse-variance
```

Meta-analysis for correlation data (StataNow)

Meta-analysis is similarly used to estimate an overall correlation in studies examining relationships between two variables. This technique involves pooling correlations from multiple studies, each of which explores the relationship between the two specific variables. In this context, the data include the correlations and their respective sample sizes for each study. To estimate a correlation, `meta esize` provides two effect sizes: Fisher's z-transformed correlations and untransformed (raw) correlations. These are specified, respectively, as options `fisherz` and `correlation` with `meta esize`.

As described in *Declaring a meta-analysis model* in [META] **meta data**, you can choose between a random-effects, a fixed-effects, or a common-effect model. You can also choose from several estimation methods for random-effects models. Fixed-effects and common-effect models assume the inverse-variance estimation method. Also see *Declaring a meta-analysis estimation method* in [META] **meta data** and *Meta-analysis estimation methods* in [META] **Intro** for details.

Let's now look at a few examples. Consider the following fictional meta-analysis dataset:

```
. use https://www.stata-press.com/data/r18/metaescorr, clear
(Fictional summary data for meta-analysis of correlations)
. describe
Contains data from https://www.stata-press.com/data/r18/metaescorr.dta
 Observations:             7                  Fictional summary data for
                                                meta-analysis of correlations
    Variables:             3                  8 Dec 2023 09:14
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
study           str7    %9s                   Study label
n               int     %9.0g                 Sample size of the study
rho             double  %6.3f                 Correlation coefficient
-------------------------------------------------------------------------------
Sorted by:
```

We will use this dataset to demonstrate different usages of the `meta esize` command to declare the data for meta-analysis of correlations.

▷ Example 11: Specify the effect size and the default setting

When we deal with correlation data, the most rudimentary syntax of `meta esize` requires that we specify the correlations (`rho`) and the study sample sizes (`n`). We must also specify one of `fisherz` (Fisher's z-transformed correlations) or `correlation` (untransformed correlations) as the effect size.

```
. meta esize rho n, fisherz
Meta-analysis setting information
 Study information
    No. of studies: 7
       Study label: Generic
        Study size: _meta_studysize
      Summary data: rho n
       Effect size
             Type: fisherz
            Label: Fisher's z
         Variable: _meta_es
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

The output shows that the summary data are defined by variables `rho` and `n` and that the Fisher's z-transformed correlation is used as the effect size. A random-effects REML model is assumed. Other settings are exactly as described in example 1.

Instead of using the Fisher's z-transformed correlation, we can choose the untransformed (raw) correlation as the effect size by using the `correlation` option.

```
. meta esize rho n, correlation
Meta-analysis setting information
 Study information
    No. of studies: 7
       Study label: Generic
        Study size: _meta_studysize
      Summary data: rho n
       Effect size
             Type: correlation
            Label: Correlation
         Variable: _meta_es
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: REML
```

The output differs from that above in the `Effect size` section. It now reflects that the untransformed (raw) correlation is the effect size of choice instead of the Fisher's z-transformed correlation.

Many authors (Borenstein and Hedges 2019) argue that untransformed correlations should be avoided particularly when there are study correlations that are close to -1 or 1.

◁

▷ Example 12: Specify an alternative meta-analysis model and method

Instead of using the default REML estimation method, you may specify an alternative random-effects method, such as the DL method. This can be done via the `random()` option.

```
. meta esize rho n, fisherz random(dlaird)
Meta-analysis setting information
 Study information
    No. of studies: 7
       Study label: Generic
        Study size: _meta_studysize
      Summary data: rho n
       Effect size
             Type: fisherz
            Label: Fisher's z
         Variable: _meta_es
       Precision
       Std. err.: _meta_se
              CI: [_meta_cil, _meta_ciu]
        CI level: 95%
  Model and method
            Model: Random effects
           Method: DerSimonian-Laird
```

Or perhaps you believe that your correlations are similar across the studies and that a common-effect model is adequate to synthesize the overall correlation. You may request a common-effect model with the inverse-variance method by specifying the `common` option.

```
. meta esize rho n, fisherz common
Meta-analysis setting information
 Study information
    No. of studies: 7
       Study label: Generic
        Study size: _meta_studysize
      Summary data: rho n
       Effect size
             Type: fisherz
            Label: Fisher's z
         Variable: _meta_es
       Precision
       Std. err.: _meta_se
              CI: [_meta_cil, _meta_ciu]
        CI level: 95%
  Model and method
            Model: Common effect
           Method: Inverse-variance
```

◁

Stored results

meta esize stores the following characteristics and system variables:

Characteristics

_dta[_meta_marker]	"_meta_ds_1"
_dta[_meta_K]	number of studies in the meta-analysis
_dta[_meta_studylabel]	name of string variable containing study labels or Generic
_dta[_meta_estype]	type of effect size; varies
_dta[_meta_eslabelopt]	eslabel(*eslab*), if specified
_dta[_meta_eslabel]	effect-size label from eslabel(); default varies
_dta[_meta_eslabeldb]	effect-size label for dialog box
_dta[_meta_esvardb]	_meta_es
_dta[_meta_level]	default confidence level for meta-analysis
_dta[_meta_esizeopt]	esize(*estype*), if specified
_dta[_meta_esopt_exact]	exact, if esize(, exact) is specified
_dta[_meta_esopt_holkinse]	holkinse, if esize(, holkinse) is specified
_dta[_meta_esopt_unequal]	unequal, if esize(, unequal) is specified
_dta[_meta_modellabel]	meta-analysis model label: Random effects, Common effect, or Fixed effects
_dta[_meta_model]	meta-analysis model: random, common, or fixed
_dta[_meta_methodlabel]	meta-analysis method label; varies by meta-analysis model
_dta[_meta_method]	meta-analysis method; varies by meta-analysis model
_dta[_meta_randomopt]	random(*remethod*), if specified
_dta[_meta_zcopt]	zerocells(*zcspec*), if specified
_dta[_meta_zcadj]	type of adjustment for zero cells, if zerocells() specified
_dta[_meta_zcvalue]	value added to cells to adjust for zero cells, if specified
_dta[_meta_show]	empty or nometashow
_dta[_meta_n1var]	name of group 1 sample-size variable; for two-sample continuous data
_dta[_meta_mean1var]	name of group 1 mean variable; for two-sample continuous data
_dta[_meta_sd1var]	name of group 1 std. dev. variable; for two-sample continuous data
_dta[_meta_n2var]	name of group 2 sample-size variable; for two-sample continuous data
_dta[_meta_mean2var]	name of group 2 mean variable; for two-sample continuous data
_dta[_meta_sd2var]	name of group 2 std. dev. variable; for two-sample continuous data
_dta[_meta_n11var]	name of *n11* variable; for two-sample binary data (contingency table)
_dta[_meta_n12var]	name of *n12* variable; for two-sample binary data (contingency table)
_dta[_meta_n21var]	name of *n21* variable; for two-sample binary data (contingency table)
_dta[_meta_n22var]	name of *n22* variable; for two-sample binary data (contingency table)
_dta[_meta_nsvar]	name of *ns* variable; for one-sample binary data (proportion)
_dta[_meta_nvar]	name of *n* variable; for one-sample binary data (proportion) or correlation data
_dta[_meta_rvar]	name of *r* variable; for correlation data
_dta[_meta_datatype]	data type; continuous, binary, proportion, or correlation
_dta[_meta_datavars]	variables specified with meta esize
_dta[_meta_setcmdline]	meta esize command line
_dta[_meta_ifexp]	*if* specification
_dta[_meta_inexp]	*in* specification

System variables

_meta_id	study ID variable
_meta_es	variable containing effect sizes
_meta_se	variable containing effect-size standard errors
_meta_cil	variable containing lower bounds of CIs for effect sizes
_meta_ciu	variable containing upper bounds of CIs for effect sizes
_meta_studylabel	string variable containing study labels
_meta_studysize	variable containing total sample size per study

Methods and formulas

Methods and formulas are presented under the following headings:

Effect sizes for two-group comparison of continuous outcomes
Unstandardized mean difference
Standardized mean difference
Effect sizes for two-group comparison of binary outcomes
Odds ratio
Risk ratio (rate ratio)
Risk difference
Zero-cells adjustments for two-sample case
Effect sizes for estimating a single proportion
Untransformed (raw) proportion
Freeman–Tukey-transformed proportion
Logit-transformed proportion
Zero-cells adjustments for one-sample case
Effect sizes for correlation data (StataNow)
Untransformed (raw) correlation
Fisher's z-transformed correlation
Confidence intervals for effect sizes

Effect sizes for two-group comparison of continuous outcomes

As we described in *Meta-analysis for two-group comparison of continuous outcomes*, meta-analysis often compares two groups: experimental (or treated) group and control group.

When the response (measurement) is continuous, studies typically report a mean and standard deviation for each group. For a given study, the following table denotes the underlying population parameters and the reported summary statistics (data) for each group.

	population		sample		
group	mean	sd	mean	sd	size
treatment	μ_1	σ_1	$\overline{x}_1$	s_1	n_1
control	μ_2	σ_2	$\overline{x}_2$	s_2	n_2

The majority of this section is based on Borenstein (2009).

Unstandardized mean difference

Consider the population mean difference

$$\theta = \mu_1 - \mu_2$$

For each study in the meta-analysis, `meta esize` with option `esize(mdiff)` estimates θ using the difference in sample means,

$$D = \overline{x}_1 - \overline{x}_2$$

The variance of D, assuming that the two population standard deviations are equal, is estimated by

$$\widehat{\text{Var}}(D) = \left(\frac{1}{n_1} + \frac{1}{n_2}\right) s^2$$

where s is the pooled sample standard deviation

$$s = \sqrt{\frac{(n_1 - 1)\, s_1^2 + (n_2 - 1)\, s_2^2}{n_1 + n_2 - 2}}$$

For unequal population standard deviations, use option `esize(mdiff, unequal)`; then the variance of D is estimated by

$$\widehat{\text{Var}}(D) = \frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}$$

Unstandardized (raw) mean differences are not comparable across studies if the underlying means are measured on different scales.

Standardized mean difference

The standardized mean difference is

$$\theta = \frac{\mu_1 - \mu_2}{\sigma}$$

Note that θ does not depend on the scale of measurement. The definition of the standardized mean difference implicitly assumes that the population standard deviations, σ_1 and σ_2, are the same: $\sigma_1 = \sigma_2 = \sigma$.

`meta esize` with option `esize(cohend)` estimates θ using Cohen's d statistic (Cohen 1969, 1988),

$$d = \frac{\overline{x}_1 - \overline{x}_2}{s}$$

The estimated variance of d is given by

$$\widehat{\text{Var}}(d) = \frac{n_1 + n_2}{n_1 n_2} + \frac{d^2}{2\,(n_1 + n_2)}$$

Hedges (1981) introduced an adjustment to Cohen's d for small samples that accounts for a small upward bias in the absolute value of θ. `meta esize` with option `esize(hedgesg, exact)` computes Hedges's g as

$$g = c(m) \times d$$

where $m = n_1 + n_2 - 2$ is the degrees of freedom used to estimate s and

$$c(m) = \frac{\Gamma\left(\frac{m}{2}\right)}{\sqrt{\frac{m}{2}}\,\Gamma\left(\frac{m-1}{2}\right)}$$

The adjustment $c(m)$ is less than 1 and approaches 1 as m gets large. The variance estimate of Hedges's g is

$$\widehat{\text{Var}}(g) = c(m)^2 \times \widehat{\text{Var}}(d)$$

Hedges (1981) also introduced an accurate approximation for $c(m)$ that has been traditionally used in meta-analysis. The approximation for $c(m)$ is

$$J = 1 - \frac{3}{4m - 1}$$

`meta esize` with option `esize(hedgesg)` computes Hedges's g using J for $c(m)$; thus,

$$g = J \times d$$

and

$$\widehat{\text{Var}}(g) = J^2 \times \widehat{\text{Var}}(d)$$

`meta esize` with option `esize(glassdelta2)` estimates θ using Glass's Δ (Smith and Glass 1977),

$$\Delta = \frac{\overline{x}_1 - \overline{x}_2}{s_2}$$

Notice that the standard deviation in the denominator is s_2, the sample standard deviation from the control group, which is considered to be a more reliable estimate of the common variance. The estimated variance of Δ is given by

$$\widehat{\text{Var}}(\Delta) = \frac{n_1 + n_2}{n_1 n_2} + \frac{\Delta^2}{2\,(n_2 - 1)}$$

In the absence of the control group, such as in observational studies, Kline (2013), among others, suggests providing statistics standardized by the standard deviation of each group. Glass's Δ where standardization is based on the treatment group may be computed via option `esize(glassdelta1)`.

Alternative standard error estimators are available for Hedges's g and Cohen's d effect sizes.

Hedges and Olkin (1985, eq. 8, 80) provide another commonly used estimator for the variance of Hedges's g.

$$\widehat{\text{Var}}(g) = \frac{n_1 + n_2}{n_1 n_2} + \frac{g^2}{2\,(m - 1.94)}$$

`meta esize` uses this formula when option `esize(hedgesg, holkinse)` is specified.

The alternative variance estimator of d is given by

$$\widehat{\text{Var}}(d) = \frac{n_1 + n_2}{n_1 n_2} + \frac{d^2}{2\,(n_1 + n_2 - 2)}$$

This variance estimator may be requested via option `esize(cohend, holkinse)`.

Effect sizes for two-group comparison of binary outcomes

As we described in *Meta-analysis for two-group comparison of binary outcomes*, meta-analysis often compares two groups: experimental (or treated) group and control group. When the response (measurement) is binary, each study typically reports cell counts from the following 2×2 table.

group	success	failure	size
treatment	a	b	$n_1 = a + b$
control	c	d	$n_2 = c + d$

Here, for simplicity, we use a different notation for the cell counts (a, b, c, and d) compared with the similar table in *Meta-analysis for two-group comparison of binary outcomes*.

For the treatment group, n_1 is assumed fixed, $a \sim \text{binomial}(n_1, \pi_1)$, and π_1 is the probability of a success. For the control group, n_2 is assumed fixed, $c \sim \text{binomial}(n_2, \pi_2)$, and π_2 is the probability of a success. The goal of each study is to compare the two success probabilities, π_1 and π_2.

Estimates of the success probabilities are $\hat{\pi}_1 = a/n_1$ for the treatment group and $\hat{\pi}_2 = c/n_2$ for the control group.

Odds ratio

`meta esize` with option `esize(lnoratio)` computes estimates of the log odds-ratios. Odds ratio is the ratio of the odds of a success in the treatment group over the odds of a success in the control group.

$$\text{OR} = \frac{\pi_1/(1-\pi_1)}{\pi_2/(1-\pi_2)}$$

The odds ratio is estimated by

$$\widehat{\text{OR}} = \frac{ad}{bc}$$

The distribution of $\widehat{\text{OR}}$ is typically skewed, but the natural logarithm of $\widehat{\text{OR}}$, $\ln(\widehat{\text{OR}})$, is asymptotically normally distributed. The estimate of the variance of $\ln(\widehat{\text{OR}})$ is

$$\widehat{\text{Var}}\left\{\ln(\widehat{\text{OR}})\right\} = \frac{1}{a}+\frac{1}{b}+\frac{1}{c}+\frac{1}{d}$$

`meta esize` with option `esize(lnorpeto)` computes estimates of effect size using Peto's log odds-ratio (Peto et al. 1977 ; Yusuf et al. 1985). Peto's odds ratio and log odds-ratio are

$$\widehat{\text{OR}}^{\text{Peto}} = \exp\left\{\frac{a-E(a)}{\text{Var}(a)}\right\}$$

$$\ln\left(\widehat{\text{OR}}^{\text{Peto}}\right) = \frac{a-E(a)}{\text{Var}(a)}$$

where the expectation and variance of a are estimated assuming a hypergeometric distribution:

$$E(a) = \frac{(a+c)n_1}{n}$$

$$\text{Var}(a) = \frac{n_1 n_2 (a+c)(b+d)}{n^2(n-1)}$$

The variance estimate of $\ln\left(\widehat{\text{OR}}^{\text{Peto}}\right)$ is

$$\widehat{\text{Var}}\left\{\ln\left(\widehat{\text{OR}}^{\text{Peto}}\right)\right\} = \frac{1}{\text{Var}(a)}$$

See, for instance, Fleiss 1993, Fleiss, Levin, and Paik 2003, and Bradburn et al. (2007) for a discussion of potential bias of Peto's odds ratio and its performance in sparse data.

Risk ratio (rate ratio)

`meta esize` with option `esize(lnrratio)` computes estimates of the log risk-ratios. The risk ratio (RR), also known as the rate ratio or relative risk in the health sciences, is

$$\text{RR} = \frac{\pi_1}{\pi_2}$$

RR is estimated by

$$\widehat{\text{RR}} = \frac{a/n_1}{c/n_2}$$

Similarly to odds ratios, $\widehat{\text{RR}}$ typically has a skewed distribution, but the natural logarithm of $\widehat{\text{RR}}$, $\ln\left(\widehat{\text{RR}}\right)$, is asymptotically normally distributed. The estimate of the variance of $\ln\left(\widehat{\text{RR}}\right)$ is

$$\widehat{\text{Var}}\left\{\ln\left(\widehat{\text{RR}}\right)\right\} = \frac{1}{a} + \frac{1}{c} - \frac{1}{a+b} - \frac{1}{c+d}$$

Risk difference

`meta esize` with option `esize(rdiff)` computes estimates of the risk differences. The risk difference is

$$\text{RD} = \pi_1 - \pi_2$$

and is estimated by

$$\widehat{\text{RD}} = \frac{a}{n_1} - \frac{c}{n_2}$$

$\widehat{\text{RD}}$ is asymptotically normally distributed and is thus typically used without a transformation in meta-analysis.

The estimated variance of $\widehat{\text{RD}}$ is

$$\widehat{\text{Var}}\left(\widehat{\text{RD}}\right) = \frac{ab}{n_1^3} + \frac{cd}{n_2^3}$$

Zero-cells adjustments for two-sample case

The variance estimates of $\ln(\widehat{\text{OR}})$ and $\ln\left(\widehat{\text{RR}}\right)$ are not defined if there are any empty (zero count) cells in a 2×2 table. In this case, it is customary to add a small value, often referred to as "continuity correction", to each cell prior to computing the log odds- or risk-ratio.

By default, `meta esize` adds 0.5 to each cell of 2×2 tables containing empty cells (Gart and Zweifel 1967 and Gart, Pettigrew, and Thomas 1985). Alternatively, you can add a different number or add a number to each cell of all 2×2 tables, as long as there is at least one 2×2 table with zero cells; see option `zerocells()`.

For odds ratios, Sweeting, Sutton, and Lambert (2004) proposed the treatment-arm continuity correction (TACC) method, which estimates the continuity-correction values from the data separately for each group; see `zerocells(tacc)`.

Effect sizes for estimating a single proportion

As we described in *Meta-analysis for estimating a single proportion*, meta-analysis may be used to aggregate proportions of a certain event of interest in single-group or single-arm studies. Each study typically reports the number of successes (number of events), e, and the study sample size, n. The number of successes e is assumed to follow a binomial(n, p) distribution, where p is the probability of success. For details, see Barendregt et al. (2013) and Nyaga, Arbyn, and Aerts (2014).

Untransformed (raw) proportion

`meta esize` with option `esize(proportion)` computes estimates of proportions for each study and uses them as effect sizes in the meta-analysis. The proportion is estimated by

$$\hat{p} = \frac{e}{n}$$

When the proportion p is near 0.5 and when n is sufficiently large, the binomial distribution of e is well approximated by the normal distribution, and a meta-analysis may be performed in the natural (untransformed) metric.

The estimated variance of $\hat{p}$ is

$$\widehat{\text{Var}}\,(\hat{p}) = \frac{\hat{p}\,(1-\hat{p})}{n}$$

Because the expression of the variance depends on $\hat{p}$, meta-analysis of this effect size tends to assign artificially large weights for studies with $\hat{p}$ close to 0 or 1. In this case, the variance of $\hat{p}$ is close to 0, and the study weights, which are the inverse variances, will be large. Also, study-specific CI limits may fall outside the range of $[0, 1]$ and, in practice, are truncated when this happens.

Freeman–Tukey-transformed proportion

By default (or with option `esize(ftukeyprop)`), `meta esize` computes the Freeman–Tukey-transformed proportions and uses them as effect sizes in the meta-analysis. The Freeman–Tukey transformation is also known as the Freeman–Tukey double-arcsine transformation in the literature. The Freeman–Tukey-transformed proportion is given by

$$\hat{p}_{\text{FT}} = \texttt{asin}\left(\sqrt{\frac{e}{n+1}}\right) + \texttt{asin}\left(\sqrt{\frac{e+1}{n+1}}\right) \tag{1}$$

with the corresponding estimated variance

$$\widehat{\text{Var}}\,(\hat{p}_{\text{FT}}) = \frac{1}{n+0.5}$$

This is a variance-stabilizing transformation (variance does not depend on e) and is particularly preferable when $\hat{p}$ is close to 0 or 1. This transformation also addresses the issue of assigning artificially small or large weights to studies in the meta-analysis when e is close to 0 or n. And it guarantees that the back-transformed CIs (see *Inverse Freeman–Tukey transformation* in *Methods and formulas* in [META] **meta summarize**) fall within the $[0, 1]$ range.

Logit-transformed proportion

`meta esize` with option `esize(logitprop)` computes logit-transformed proportions and uses them as effect sizes in the meta-analysis. The logit-transformed proportion is estimated by

$$\texttt{logit}\,(\hat{p}) = \ln\left(\frac{\hat{p}}{1-\hat{p}}\right)$$

with the corresponding estimated variance

$$\widehat{\text{Var}}\,\{\texttt{logit}\,(\hat{p})\} = \frac{1}{n\hat{p}} + \frac{1}{n-n\hat{p}}$$

This transformation allows aggregating the proportions in a metric that is closer to normality and guarantees that the back-transformed CI limits (computed using the `invlogit()` function) are between 0 and 1 (inclusive). Because the expression of the variance depends on $\hat{p}$, meta-analysis of this effect size tends to assign artificially low weights for studies with $\hat{p}$ close to 0 or 1. In this case, the variances for such studies are large, and the study weights, which are the inverse variances, will be low.

Zero-cells adjustments for one-sample case

When a study reports a zero cell (zero successes or zero failures), the variance of $\hat{p}$ is equal to 0, and the variance of `logit` $(\hat{p})$ is not defined. In this case, it is customary to add a small value, often referred to as "continuity correction", to each cell prior to computing the proportion or the logit-transformed proportion.

By default, `meta esize` adds 0.5 to each cell of studies containing zero cells (Gart and Zweifel 1967 and Gart, Pettigrew, and Thomas 1985). In other words, for a study reporting zero cells, the number of successes, e, will be incremented by 0.5, the number of failures will be incremented by 0.5, and therefore, the total sample size, n, will increase by 1.

Alternatively, you can add a different number or add a number to each cell of all studies, as long as there is at least one study with zero cells; see option `zerocells()`.

Effect sizes for correlation data (StataNow)

As we described in Meta-analysis for correlation data (StataNow), meta-analysis may be used to aggregate correlations between two variables of interest. Each study typically reports the sample correlation, r, and the study sample size, n. For details, see Borenstein and Hedges (2019).

Untransformed (raw) correlation

The estimate of the correlation parameter ρ is the sample correlation, denoted by r. The estimated variance of r is given by

$$\widehat{\text{Var}}\,(r) = \frac{\left(1-r^2\right)^2}{n-1}$$

When you deal with the sample correlation, it is well known that the large-sample theory does not hold up well for small sample sizes, particularly when r is close to 1 or -1. In contrast, large-sample theory of the Fisher z-transformed correlation often yields reliable results with sample sizes of 20 or more (Hedges 2019). Therefore, it is generally recommended to apply the Fisher's z-transformation for analyses where correlations are used as the measure of effect size.

Fisher's z-transformed correlation

When the `fisherz` option is specified, `meta esize` computes the Fisher's z-transformed correlations and uses them as effect sizes in the meta-analysis. The Fisher's z-transformed correlation is given by

$$f_z = \frac{1}{2}\log\left(\frac{1+r}{1-r}\right) = \texttt{atanh}(r)$$

where `atanh(`$\cdot$`)` is the inverse hyperbolic tangent function. The corresponding estimated variance is

$$\widehat{\text{Var}}\,(f_z) = \frac{1}{n-3}$$

This is a variance-stabilizing transformation (variance does not depend on r) and is particularly preferable when r is close to -1 or 1.

Confidence intervals for effect sizes

For the jth study in a given meta-analysis, let $\hat{\theta}_j$ be one of the effect-size estimators described above; then the asymptotic $100(1-\alpha)\%$ confidence interval computed by `meta esize` is

$$\hat{\theta}_j \pm z_{1-\alpha/2}\sqrt{\widehat{\text{Var}}(\hat{\theta}_j)}$$

where $z_{1-\alpha/2}$ is the usual critical value from the standard normal distribution.

References

Barendregt, J. J., S. A. Doi, Y. Y. Lee, R. E. Norman, and T. Vos. 2013. Meta-analysis of prevalence. *Journal of Epidemiology and Community Health* 67: 974–978. https://doi.org/10.1136/jech-2013-203104.

Berkey, C. S., D. C. Hoaglin, F. Mosteller, and G. A. Colditz. 1995. A random-effects regression model for meta-analysis. *Statistics in Medicine* 14: 395–411. https://doi.org/10.1002/sim.4780140406.

Borenstein, M. 2009. "Effect sizes for continuous data". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 221–235. 2nd ed. New York: Russell Sage Foundation.

Borenstein, M., and L. V. Hedges. 2019. "Effect sizes for meta-analysis". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 207–243. 3rd ed. New York: Russell Sage Foundation.

Bradburn, M. J., J. J. Deeks, J. A. Berlin, and A. R. Localio. 2007. Much ado about nothing: A comparison of the performance of meta-analytical methods with rare events. *Statistics in Medicine* 26: 53–77. https://doi.org/10.1002/sim.2528.

Cohen, J. 1969. *Statistical Power Analysis for the Behavioral Sciences*. New York: Academic Press.

———. 1988. *Statistical Power Analysis for the Behavioral Sciences*. 2nd ed. Hillsdale, NJ: Erlbaum.

DerSimonian, R., and N. M. Laird. 1986. Meta-analysis in clinical trials. *Controlled Clinical Trials* 7: 177–188. https://doi.org/10.1016/0197-2456(86)90046-2.

Fleiss, J. L. 1993. Review papers: The statistical basis of meta-analysis. *Statistical Methods in Medical Research* 2: 121–145. https://doi.org/10.1177/096228029300200202.

Fleiss, J. L., B. Levin, and M. C. Paik. 2003. *Statistical Methods for Rates and Proportions*. 3rd ed. New York: Wiley. https://doi.org/10.1002/0471445428.

Freeman, M. F., and J. W. Tukey. 1950. Transformations related to the angular and the square root. *Annals of Mathematical Statistics* 21: 607–611. https://doi.org/10.1214/aoms/1177729756.

Gart, J. J., H. M. Pettigrew, and D. G. Thomas. 1985. The effect of bias, variance estimation, skewness and kurtosis of the empirical logit on weighted least squares analyses. *Biometrika* 72: 179–190. https://doi.org/10.2307/2336348.

Gart, J. J., and J. R. Zweifel. 1967. On the bias of various estimators of the logit and its variance with application to quantal bioassay. *Biometrika* 54: 181–187. https://doi.org/10.2307/2333861.

Hardy, R. J., and S. G. Thompson. 1996. A likelihood approach to meta-analysis with random effects. *Statistics in Medicine* 15: 619–629. https://doi.org/10.1002/(SICI)1097-0258(19960330)15:6<619::AID-SIM188>3.0.CO;2-A.

Hedges, L. V. 1981. Distribution theory for Glass's estimator of effect size and related estimators. *Journal of Educational Statistics* 6: 107–128. https://doi.org/10.2307/1164588.

———. 1983. A random effects model for effect sizes. *Psychological Bulletin* 93: 388–395. https://doi.org/10.1037/0033-2909.93.2.388.

———. 2019. "Statistical considerations". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 37–48. 3rd ed. New York: Russell Sage Foundation.

Hedges, L. V., and I. Olkin. 1985. *Statistical Methods for Meta-Analysis*. Orlando, FL: Academic Press.

Kline, R. B. 2013. *Beyond Significance Testing: Statistics Reform in the Behavioral Sciences*. 2nd ed. Washington, DC: American Psychological Association.

Nyaga, V. N., M. Arbyn, and M. Aerts. 2014. Metaprop: A Stata command to perform meta-analysis of binomial data. *Archives of Public Health* 72. https://doi.org/10.1186/2049-3258-72-39.

Paule, R. C., and J. Mandel. 1982. Consensus values and weighting factors. *Journal of Research of the National Bureau of Standards* 87: 377–385. https://doi.org/10.6028/jres.087.022.

Peto, R., M. C. Pike, P. Armitage, N. E. Breslow, D. R. Cox, S. V. Howard, N. Mantel, K. McPherson, J. Peto, and P. G. Smith. 1977. Design and analysis of randomized clinical trials requiring prolonged observation of each patient. II. Analysis and examples. *British Journal of Cancer* 35: 1–39. https://doi.org/10.1038/bjc.1977.1.

Raudenbush, S. W. 2009. "Analyzing effect sizes: Random-effects models". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 295–316. 2nd ed. New York: Russell Sage Foundation.

Schmidt, F. L., and J. E. Hunter. 2015. *Methods of Meta-Analysis: Correcting Error and Bias in Research Findings*. 3rd ed. Thousand Oaks, CA: Sage. https://doi.org/10.4135/9781483398105.

Sidik, K., and J. N. Jonkman. 2005. A note on variance estimation in random effects meta-regression. *Journal of Biopharmaceutical Statistics* 15: 823–838. https://doi.org/10.1081/BIP-200067915.

Smith, M. L., and G. V. Glass. 1977. Meta-analysis of psychotherapy outcome studies. *American Psychologist* 32: 752–760. https://doi.org/10.1037/0003-066X.32.9.752.

Sweeting, M. J., A. J. Sutton, and P. C. Lambert. 2004. What to add to nothing? Use and avoidance of continuity corrections in meta-analysis of sparse data. *Statistics in Medicine* 23: 1351–1375. https://doi.org/10.1002/sim.1761.

Veroniki, A. A., D. Jackson, W. Viechtbauer, R. Bender, J. Bowden, G. Knapp, O. Kuss, J. P. T. Higgins, D. Langan, and G. Salanti. 2016. Methods to estimate the between-study variance and its uncertainty in meta-analysis. *Research Synthesis Methods* 7: 55–79. https://doi.org/10.1002/jrsm.1164.

Yusuf, S., R. Peto, J. A. Lewis, R. Collins, and P. Sleight. 1985. Beta blockade during and after myocardial infarction: An overview of the randomized trials. *Progress in Cardiovascular Diseases* 27: 335–371. https://doi.org/10.1016/S0033-0620(85)80003-7.

Also see

[META] **meta data** — Declare meta-analysis data

[META] **meta set** — Declare meta-analysis data using generic effect sizes

[META] **meta update** — Update, describe, and clear meta-analysis settings

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

[R] **esize** — Effect size based on mean comparison

meta set — Declare meta-analysis data using generic effect sizes

Description Quick start Menu Syntax Options
Remarks and examples Stored results References Also see

Description

`meta set` declares the data in memory to be `meta` data, informing Stata of key variables and their roles in a meta-analysis. It is used with generic (precomputed) effect sizes specified in the metric closest to normality; see [META] **meta esize** if you need to compute and declare effect sizes. You must use `meta set` or `meta esize` to perform univariate meta-analysis using the `meta` command; see [META] **meta data**.

If you need to update some of the meta settings after the data declaration, see [META] **meta update**. To display current meta settings, use `meta query`; see [META] **meta update**.

Quick start

Declare generic effect sizes and their standard errors from individual studies stored in variables `es` and `se`

```
meta set es se
```

Same as above, but request a random-effects meta-analysis where between-study heterogeneity is estimated using the DerSimonian–Laird method instead of the default REML method

```
meta set es se, random(dlaird)
```

Specify a common-effect meta-analysis, study labels stored in a string variable `studylab`, and label effect sizes as `log(HR)` in the output

```
meta set es se, common studylabel(studylab) eslabel("log(HR)")
```

Use 90% confidence level, and suppress the display of meta settings for all subsequent meta-analysis commands

```
meta set es se, level(90) nometashow
```

Specify study sizes stored in variable `ssize`

```
meta set es se, studysize(ssize)
```

Declare generic effect sizes, and compute their standard errors based on the specified 90% CI variables, `cil` and `ciu`

```
meta set es cil ciu, civarlevel(90)
```

Menu

Statistics > Meta-analysis

Syntax

Specify generic effect sizes and their standard errors

```
meta set esvar sevar [if] [in] [, options]
```

Specify generic effect sizes and their confidence intervals

```
meta set esvar cilvar ciuvar [if] [in] [, civarlevel(#) civartolerance(#)
    options]
```

esvar specifies a variable containing the effect sizes, *sevar* specifies a variable containing standard errors of the effect sizes, and *cilvar* and *ciuvar* specify variables containing the respective lower and upper bounds of (symmetric) confidence intervals for the effect sizes. *esvar* and the other variables must correspond to effect sizes specified in the metric closest to normality, such as log odds-ratios instead of odds ratios.

options	Description
Model	
`random`[`(`*remethod*`)`]	random-effects meta-analysis; default is `random(reml)`
`common`	common-effect meta-analysis; implies inverse-variance method
`fixed`	fixed-effects meta-analysis; implies inverse-variance method
Options	
`studylabel(`*varname*`)`	variable to be used to label studies in all meta-analysis output
`studysize(`*varname*`)`	total sample size per study
`eslabel(`*string*`)`	effect-size label to be used in all meta-analysis output; default is `eslabel(Effect size)`
`level(#)`	confidence level for all subsequent meta-analysis commands
[`no`]`metashow`	display or suppress meta settings with other `meta` commands

remethod	Description
`reml`	restricted maximum likelihood; the default
`mle`	maximum likelihood
`ebayes`	empirical Bayes
`dlaird`	DerSimonian–Laird
`sjonkman`	Sidik–Jonkman
`hedges`	Hedges
`hschmidt`	Hunter–Schmidt

Options

Main

`civarlevel(#)` is relevant only when you specify CI variables *cilvar* and *ciuvar* with `meta set`. It specifies the confidence level corresponding to these variables. The default is `civarlevel(95)`. This option affects the computation of the effect-size standard errors stored in the system variable `_meta_se`.

Do not confuse `civarlevel()` with `level()`. The former affects the confidence level only for the specified CI variables. The latter specifies the confidence level for the meta-analysis.

`civartolerance(#)` is relevant only when you specify CI variables *cilvar* and *ciuvar* with `meta set`. *cilvar* and *ciuvar* must define a symmetric CI based on the normal distribution. `civartolerance()` specifies the tolerance to check whether the CI is symmetric. The default is `civartolerance(1e-6)`. Symmetry is declared when `reldif(`*ciuvar* − *esvar*`,`*esvar* − *cilvar*`)` $<$ #.

`meta set` expects the effect sizes and CIs to be specified in the metric closest to normality, which implies symmetric CIs. Effect sizes and their CIs are often reported in the original metric and with limited precision that, after the normalizing transformation, may lead to asymmetric CIs. In that case, the default of 1e–6 may be too stringent. You may use `civartolerance()` to loosen the default.

Model

Options `random()`, `common`, and `fixed` declare the meta-analysis model globally throughout the entire meta-analysis; see *Declaring a meta-analysis model* in [META] **meta data**. In other words, once you set your meta-analysis model using `meta set`, all subsequent `meta` commands will assume that same model. You can update the declared model by using `meta update` or change it temporarily by specifying the corresponding option with the `meta` commands. Options `random()`, `common`, and `fixed` may not be combined. If these options are omitted, `random(reml)` is assumed; see *Default meta-analysis model and method* in [META] **meta data**. Also see *Meta-analysis models* in [META] **Intro**.

`random` and `random(`*remethod*`)` specify that a random-effects model be assumed for meta-analysis; see *Random-effects model* in [META] **Intro**.

remethod specifies the type of estimator for the between-study variance τ^2. *remethod* is one of `reml`, `mle`, `ebayes`, `dlaird`, `sjonkman`, `hedges`, or `hschmidt`. `random` is a synonym for `random(reml)`. Below, we provide a short description for each method based on Veroniki et al. (2016). Also see *Declaring a meta-analysis estimation method* in [META] **meta data**.

`reml`, the default, specifies that the REML method (Raudenbush 2009) be used to estimate τ^2. This method produces an unbiased, nonnegative estimate of the between-study variance and is commonly used in practice. Method `reml` requires iteration.

`mle` specifies that the ML method (Hardy and Thompson 1996) be used to estimate τ^2. It produces a nonnegative estimate of the between-study variance. With a few studies or small studies, this method may produce biased estimates. With many studies, the ML method is more efficient than the REML method. Method `mle` requires iteration.

`ebayes` specifies that the empirical Bayes estimator (Berkey et al. 1995), also known as the Paule–Mandel estimator (Paule and Mandel 1982), be used to estimate τ^2. From simulations, this method, in general, tends to be less biased than other random-effects methods, but it is also less efficient than `reml` or `dlaird`. Method `ebayes` produces a nonnegative estimate of τ^2 and requires iteration.

`dlaird` specifies that the DerSimonian–Laird method (DerSimonian and Laird 1986) be used to estimate τ^2. This method, historically, is one of the most popular estimation methods because it does not make any assumptions about the distribution of random effects and does not require iteration. But it may underestimate the true between-study variance, especially when the variability is large and the number of studies is small. This method may produce a negative value of τ^2 and is thus truncated at zero in that case.

`sjonkman` specifies that the Sidik–Jonkman method (Sidik and Jonkman 2005) be used to estimate τ^2. This method always produces a nonnegative estimate of the between-study variance and thus does not need truncating at 0, unlike the other noniterative methods. Method `sjonkman` does not require iteration.

`hedges` specifies that the Hedges method (Hedges 1983) be used to estimate τ^2. When the sampling variances of effect-size estimates can be estimated without bias, this estimator is exactly unbiased (before truncation), but it is not widely used in practice (Veroniki et al. 2016). Method `hedges` does not require iteration.

`hschmidt` specifies that the Hunter–Schmidt method (Schmidt and Hunter 2015) be used to estimate τ^2. Although this estimator achieves a lower MSE than other methods, except ML, it is known to be negatively biased. Method `hschmidt` does not require iteration.

`common` specifies that a common-effect model be assumed for meta-analysis; see *Common-effect ("fixed-effect") model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about common-effect versus fixed-effects models.

`fixed` specifies that a fixed-effects model be assumed for meta-analysis; see *Fixed-effects model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about fixed-effects versus common-effect models.

Options

`studylabel(`*varname*`)` specifies a string variable containing labels for the individual studies to be used in all applicable meta-analysis output. The default study labels are `Study 1`, `Study 2`, ..., `Study` K, where K is the total number of studies in the meta-analysis.

`studysize(`*varname*`)` specifies the variable that contains the total sample size for each study. This option is useful for subsequent `meta` commands that use this information in computations such as `meta funnelplot` using the sample-size metric.

`eslabel(`*string*`)` specifies that *string* be used as the effect-size label in all relevant meta-analysis output. The default label is `Effect size`.

`level(`#`)` specifies the confidence level, as a percentage, for confidence intervals. It will be used by all subsequent meta-analysis commands when computing confidence intervals. The default is `level(95)` or as set by `set level`; see [R] **level**. After the declaration, you can specify `level()` with `meta update` to update the confidence level to be used throughout the rest of the meta-analysis session. You can also specify `level()` directly with the `meta` commands to modify the confidence level, temporarily, during the execution of the command.

`metashow` and `nometashow` display or suppress the meta setting information in the output of other `meta` commands. By default, this information is displayed at the top of their output. You can also specify `nometashow` with `meta update` to suppress the meta setting output for the entire meta-analysis session after the declaration.

Remarks and examples

Remarks are presented under the following headings:

Overview
Using meta set

Overview

When you perform meta-analysis, it is common for studies included in the meta-analysis to contain precalculated effect sizes, which we refer to as generic effect sizes, such as mean differences, odds ratios, correlations, and hazard ratios. You can use `meta set` to declare the generic effect sizes specified in the metric closest to normality. (If you have summary data from which effect sizes can be computed, use [META] **meta esize** instead.)

In addition to effect sizes, their standard errors must be available for meta-analysis. Sometimes, the standard errors are not available, but the confidence intervals (CIs) are. In that case, the standard errors can be computed from the effect-size estimates and CIs. `meta set` supports both cases. You can supply the variables containing effect sizes and their standard errors, or, instead of the standard errors, you can specify the variables containing the CIs.

When you specify the CI variables, you can specify their corresponding confidence level in the `civarlevel()` option. (Do not confuse this option with the `level()` option. The former corresponds to the specified CI variables, whereas the latter specifies the confidence level for the entire meta-analysis.)

Meta-analysis uses effect sizes in a metric that makes them approximately normally distributed such as log odds-ratios instead of odds ratios and log hazard-ratios instead of hazard ratios. As such, `meta set` expects the effect sizes and measures of their precision to be specified in the metric closest to normality. So, the corresponding standard errors or CIs should be provided in the same metric as effect sizes. For example, if you are working with hazard ratios, you should specify log hazard-ratios with `meta set` and provide CIs for the log hazard-ratios and not the hazard ratios.

See [META] **meta data** for more details.

Using meta set

Consider the following fictional meta-analysis dataset:

```
. use https://www.stata-press.com/data/r18/metaset
(Generic effect sizes; fictional data)
. describe
Contains data from https://www.stata-press.com/data/r18/metaset.dta
 Observations:            10                  Generic effect sizes; fictional
                                                data
    Variables:             9                  19 Apr 2022 01:28
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
study           byte    %9.0g                 Study ID
es              double  %10.0g                Effect sizes
se              double  %10.0g                Std. err. for effect sizes
cil             double  %10.0g                95% lower CI limit
ciu             double  %10.0g                95% upper CI limit
cil90           double  %10.0g                90% lower CI limit
ciu90           double  %10.0g                90% upper CI limit
studylab        str23   %23s                  Study label
ssize           byte    %9.0g                 Study size
-------------------------------------------------------------------------------
Sorted by:
```

We will use it to describe various usages of the `meta set` command. For examples of declarations of real datasets, see [META] **meta data**. We assume that `es` contains the effect sizes that are approximately normal (perhaps after a suitable transformation) and that `se`, `cil`, and `ciu` contain their corresponding standard errors and CIs.

▷ Example 1: Declaring effect sizes and standard errors

Meta-analysis datasets often contain precomputed effect sizes and their standard errors. To declare them for meta-analysis using the `meta` commands, we specify the corresponding variables with `meta set`.

```
. meta set es se
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: N/A
       Effect size
             Type: <generic>
            Label: Effect size
         Variable: es
        Precision
        Std. err.: se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

Briefly, `meta set` reports that there are 10 studies, that `es` and `se` are the variables used to declare effect sizes and their standard errors, that the default confidence level is 95%, and more. See *Meta settings with meta set* in [META] **meta data** for a detailed description of all settings for this dataset.

We can now use, for example, `meta summarize` to compute the overall effect size (labeled as `theta` in the output below).

```
. meta summarize
  Effect-size label: Effect size
        Effect size: es
          Std. err.: se
Meta-analysis summary                     Number of studies =     10
Random-effects model                      Heterogeneity:
Method: REML                                          tau2 =  0.0157
                                                   I2 (%) =    5.30
                                                       H2 =    1.06
```

Study	Effect size	[95% conf.	interval]	% weight
Study 1	1.480	-0.352	3.311	2.30
Study 2	0.999	-0.933	2.931	2.07
Study 3	1.272	0.427	2.117	10.15
Study 4	1.001	0.750	1.252	63.77
Study 5	1.179	-0.527	2.884	2.65
Study 6	1.939	0.427	3.452	3.35
Study 7	2.377	1.005	3.750	4.05
Study 8	0.694	-0.569	1.956	4.75
Study 9	1.099	-0.147	2.345	4.88
Study 10	1.805	-0.151	3.761	2.02
theta	1.138	0.857	1.418	

```
Test of theta = 0: z = 7.95                  Prob > |z| = 0.0000
Test of homogeneity: Q = chi2(9) = 6.34        Prob > Q = 0.7054
```

See [META] **meta summarize** for details about this command.

◁

▷ Example 2: Declaring effect sizes and confidence intervals

Continuing with example 1, we find that some meta-analysis datasets contain confidence intervals associated with effect sizes instead of standard errors. In that case, you can specify confidence intervals with `meta set` instead of the standard errors. For example, variables `cil` and `ciu` contain the 95% lower and upper CI limits for the effect sizes stored in variable `es`. We can declare them as follows.

```
. meta set es cil ciu
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: N/A
       Effect size
              Type: <generic>
             Label: Effect size
          Variable: es
         Precision
         Std. err.: _meta_se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%, controlled by level()
           User CI: [cil, ciu]
     User CI level: 95%, controlled by civarlevel()
  Model and method
             Model: Random effects
            Method: REML
```

Compared with `Std. err.:` in example 1, `Std. err.:` under `Precision` now contains the system variable `_meta_se`; see *System variables* in [META] **meta data**. The standard errors are computed from `cil` and `ciu` and stored in this system variable. The CI values are stored in the corresponding system variables `_meta_cil` and `_meta_ciu`.

The output additionally reports the user-specified CI variables, `cil` and `ciu`, under `User CI:` and their corresponding confidence level, 95%, under `User CI level:`. As we will see later, `User CI level`, controlled by the `civarlevel()` option, and `CI level`, controlled by the `level()` option, may be different.

Let's now check that we obtain the same results as before using the equivalent CI declaration.

```
. meta summarize
  Effect-size label: Effect size
        Effect size: es
          Std. err.: _meta_se
Meta-analysis summary                     Number of studies =     10
Random-effects model                      Heterogeneity:
Method: REML                                          tau2 =  0.0157
                                                  I2 (%) =    5.30
                                                      H2 =    1.06
```

Study	Effect size	[95% conf.	interval]	% weight
Study 1	1.480	-0.352	3.311	2.30
Study 2	0.999	-0.933	2.931	2.07
Study 3	1.272	0.427	2.117	10.15
Study 4	1.001	0.750	1.252	63.77
Study 5	1.179	-0.527	2.884	2.65
Study 6	1.939	0.427	3.452	3.35
Study 7	2.377	1.005	3.750	4.05
Study 8	0.694	-0.569	1.956	4.75
Study 9	1.099	-0.147	2.345	4.88
Study 10	1.805	-0.151	3.761	2.02
theta	1.138	0.857	1.418	

```
Test of theta = 0: z = 7.95                  Prob > |z| = 0.0000
Test of homogeneity: Q = chi2(9) = 6.34       Prob > Q = 0.7054
```

In the earlier `meta set`, we assumed that the `cil` and `ciu` variables correspond to the 95% CIs. Although typical, this may not always be the case. You can use the `civarlevel()` option to specify the confidence level of the CI variables. We have variables `cil90` and `ciu90` in our dataset, which contain the 90% CIs for `es`. We can use them in the declaration as long as we also specify the `civarlevel(90)` option.

```
. meta set es cil90 ciu90, civarlevel(90)
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: N/A
       Effect size
             Type: <generic>
            Label: Effect size
         Variable: es
        Precision
        Std. err.: _meta_se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%, controlled by level()
          User CI: [cil90, ciu90]
    User CI level: 90%, controlled by civarlevel()
  Model and method
            Model: Random effects
           Method: REML
```

The User CI level now contains 90%. Do not confuse the civarlevel() option, whose value is reported in User CI level, with the level() option, whose value is reported in CI level. The former specifies the confidence level corresponding to the declared CI variables. The latter specifies the confidence level that will be used to compute various confidence intervals during your meta-analysis session. Note that the system CI variables, _meta_cil and _meta_ciu, always correspond to the confidence level controlled by level().

```
. meta summarize

  Effect-size label: Effect size
        Effect size: es
          Std. err.: _meta_se

Meta-analysis summary                     Number of studies =     10
Random-effects model                      Heterogeneity:
Method: REML                                          tau2 =  0.0157
                                                  I2 (%) =    5.30
                                                      H2 =    1.06

------------------------------------------------------------------------------
       Study |  Effect size    [95% conf. interval]  % weight
-------------+----------------------------------------------------------------
     Study  1 |       1.480      -0.352       3.311      2.30
     Study  2 |       0.999      -0.933       2.931      2.07
     Study  3 |       1.272       0.427       2.117     10.15
     Study  4 |       1.001       0.750       1.252     63.77
     Study  5 |       1.179      -0.527       2.884      2.65
     Study  6 |       1.939       0.427       3.452      3.35
     Study  7 |       2.377       1.005       3.750      4.05
     Study  8 |       0.694      -0.569       1.956      4.75
     Study  9 |       1.099      -0.147       2.345      4.88
     Study 10 |       1.805      -0.151       3.761      2.02
-------------+----------------------------------------------------------------
       theta |       1.138       0.857       1.418
------------------------------------------------------------------------------
Test of theta = 0: z = 7.95                  Prob > |z| = 0.0000
Test of homogeneity: Q = chi2(9) = 6.34       Prob > Q = 0.7054
```

Although the specified CI variables corresponded to the 90% confidence level, the CIs reported by meta summarize are the 95% CIs because the default confidence level is 95%, level(95).

◁

❑ Technical note

As we mentioned earlier, meta set expects the effect sizes and measures of their precision such as CIs to be specified in the metric closest to normality, which implies symmetric CIs. When you specify CIs with meta set, the command checks that the CIs are symmetric within a certain tolerance. The default tolerance is 1e–6.

In practice, effect sizes and their CIs are often reported in the original metric and with limited precision that, after the normalizing transformation, may lead to asymmetric CIs. In that case, the default of 1e–6 may be too stringent. You may loosen the tolerance by specifying the civartolerance() option.

❑

▷ Example 3: Declaring meta-analysis methods and models

In *Declaring a meta-analysis model* in [META] **meta data**, we describe the importance of choosing the appropriate meta-analysis model and method for the analysis. Here we demonstrate how to specify different meta-analysis models and methods.

From example 1 and as described in *Default meta-analysis model and method* in [META] **meta data**, the default meta-analysis model and estimation method are random-effects and REML. We can specify a different random-effects method in the `random()` option. For example, let's use the DerSimonian–Laird estimation method.

```
. meta set es se, random(dlaird)
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: N/A
       Effect size
              Type: <generic>
             Label: Effect size
          Variable: es
         Precision
         Std. err.: se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Random effects
            Method: DerSimonian-Laird
```

`meta set` reports in `Method:` that the current method is now DerSimonian–Laird.

We can also choose a different meta-analysis model. For example, we can specify a fixed-effects model by using the `fixed` option.

```
. meta set es se, fixed
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: N/A
       Effect size
              Type: <generic>
             Label: Effect size
          Variable: es
         Precision
         Std. err.: se
                CI: [_meta_cil, _meta_ciu]
          CI level: 95%
  Model and method
             Model: Fixed effects
            Method: Inverse-variance
```

The inverse-variance estimation method is assumed for the fixed-effects model.

We can also specify a common-effect model, although the literature does not recommend starting your meta-analysis with this model.

```
. meta set es se, common
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: N/A
       Effect size
             Type: <generic>
            Label: Effect size
         Variable: es
       Precision
       Std. err.: se
              CI: [_meta_cil, _meta_ciu]
        CI level: 95%
  Model and method
            Model: Common effect
           Method: Inverse-variance
```

The inverse-variance estimation method is assumed for the common-effect model.

As we describe in *Declaring a meta-analysis model* in [META] **meta data**, some of the meta-analysis will not be available for common-effect models. For example, because a common-effect model implies no heterogeneity, you will not be able to perform tests of small-study effects using `meta bias` in the presence of moderators.

```
. meta bias x, egger
meta bias with moderators not supported with a common-effect model
    The declared model is a common-effect model, which assumes no
    heterogeneity. Specifying moderators that account for potential
    heterogeneity is not valid in this case. You may override this
    assumption by specifying one of options fixed or random(remethod).
r(498);
```

See [META] **meta bias**.

◁

▷ Example 4: Specifying study and effect-size labels, confidence level, and more

In *Declaring display settings for meta-analysis* of [META] **meta data**, we describe the options to control the display from the `meta` commands. Below, we use `studylabel()` and `eslabel()` to specify our own study and effect-size labels, `level(90)` to report the 90% CIs, and `nometashow` to suppress the information about the effect-size variables and standard error variables in the output of all `meta` commands.

```
. meta set es se, studylabel(studylab) eslabel("Mean diff.") level(90)
> nometashow

Meta-analysis setting information

 Study information
    No. of studies: 10
       Study label: studylab
        Study size: N/A
       Effect size
             Type: <generic>
            Label: Mean diff.
         Variable: es
        Precision
        Std. err.: se
               CI: [_meta_cil, _meta_ciu]
         CI level: 90%
  Model and method
            Model: Random effects
           Method: REML
```

If we now run `meta summarize`, we will see the new labels for the studies in the `Study` column, the effect-size column labeled as `Mean diff.`, the 90% CIs, and no meta setting information above the table header.

```
. meta summarize

Meta-analysis summary                           Number of studies =     10
Random-effects model                            Heterogeneity:
Method: REML                                                 tau2 =  0.0157
                                                          I2 (%) =    5.30
                                                              H2 =    1.06
```

Study	Mean diff.	[90% conf.	interval]	% weight
Smith et al. (1984)	1.480	-0.057	3.016	2.30
Jones and Miller (1989)	0.999	-0.622	2.620	2.07
Johnson et al. (1991)	1.272	0.563	1.981	10.15
Brown et al. (1995)	1.001	0.790	1.211	63.77
Clark and Thomas (1998)	1.179	-0.252	2.610	2.65
Williams et al. (2003)	1.939	0.670	3.209	3.35
Davis and Wilson (2010)	2.377	1.226	3.529	4.05
Moore and Parker (2014)	0.694	-0.366	1.753	4.75
Miller et al. (2018)	1.099	0.053	2.144	4.88
Assaad et al. (2019)	1.805	0.164	3.446	2.02
theta	1.138	0.902	1.373	

```
Test of theta = 0: z = 7.95                     Prob > |z| = 0.0000
Test of homogeneity: Q = chi2(9) = 6.34           Prob > Q = 0.7054
```

◁

▷ Example 5: Specifying study size

Some analysis such as a funnel plot with sample-size metrics (see [META] **meta funnelplot**) requires that you specify the sample size for each study with `meta set`. You can use the `studysize()` option for this.

```
. meta set es se, studysize(ssize)
Meta-analysis setting information
 Study information
    No. of studies: 10
       Study label: Generic
        Study size: ssize
       Effect size
             Type: <generic>
            Label: Effect size
         Variable: es
        Precision
        Std. err.: se
               CI: [_meta_cil, _meta_ciu]
         CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

The name of the study-size variable, `ssize`, is now reported in `Study size:`.

◁

Stored results

`meta set` stores the following characteristics and system variables:

Characteristics

`_dta[_meta_marker]`	"`_meta_ds_1`"
`_dta[_meta_K]`	number of studies in the meta-analysis
`_dta[_meta_studylabel]`	name of string variable containing study labels or `Generic`
`_dta[_meta_studysize]`	name of numeric variable containing study sizes, when `studysize()` specified
`_dta[_meta_estype]`	type of effect size; `Generic`
`_dta[_meta_eslabelopt]`	`eslabel(`*eslab*`)`, if specified
`_dta[_meta_eslabel]`	effect-size label from `eslabel()`; default is `Effect size`
`_dta[_meta_eslabeldb]`	effect-size label for dialog box
`_dta[_meta_esvar]`	name of effect-size variable
`_dta[_meta_esvardb]`	abbreviated name of effect-size variable for dialog box
`_dta[_meta_sevar]`	name of standard-error variable, if specified, or `_meta_se`
`_dta[_meta_cilvar]`	name of variable containing lower CI bounds, if specified, or `_meta_cil`
`_dta[_meta_ciuvar]`	name of variable containing upper CI bounds, if specified, or `_meta_ciu`
`_dta[_meta_civarlevel]`	confidence level associated with CI variables, if specified
`_dta[_meta_civartol]`	tolerance for checking CI symmetry; default is `1e-6`
`_dta[_meta_level]`	default confidence level for meta-analysis
`_dta[_meta_modellabel]`	meta-analysis model label: `Random effects`, `Common effect`, or `Fixed effects`
`_dta[_meta_model]`	meta-analysis model: `random`, `common`, or `fixed`
`_dta[_meta_methodlabel]`	meta-analysis method label; varies by meta-analysis model
`_dta[_meta_method]`	meta-analysis method; varies by meta-analysis model
`_dta[_meta_randomopt]`	`random(`*remethod*`)`, if specified
`_dta[_meta_show]`	empty or `nometashow`

_dta[_meta_datatype]	data type; Generic
_dta[_meta_datavars]	variables specified with meta set
_dta[_meta_setcmdline]	meta set command line
_dta[_meta_ifexp]	*if* specification
_dta[_meta_inexp]	*in* specification
System variables	
_meta_id	study ID variable
_meta_es	variable containing effect sizes
_meta_se	variable containing effect-size standard errors
_meta_cil	variable containing lower bounds of CIs for effect sizes
_meta_ciu	variable containing upper bounds of CIs for effect sizes
_meta_studylabel	string variable containing study labels
_meta_studysize	variable containing total sample size per study

References

Berkey, C. S., D. C. Hoaglin, F. Mosteller, and G. A. Colditz. 1995. A random-effects regression model for meta-analysis. *Statistics in Medicine* 14: 395–411. https://doi.org/10.1002/sim.4780140406.

DerSimonian, R., and N. M. Laird. 1986. Meta-analysis in clinical trials. *Controlled Clinical Trials* 7: 177–188. https://doi.org/10.1016/0197-2456(86)90046-2.

Hardy, R. J., and S. G. Thompson. 1996. A likelihood approach to meta-analysis with random effects. *Statistics in Medicine* 15: 619–629. https://doi.org/10.1002/(SICI)1097-0258(19960330)15:6<619::AID-SIM188>3.0.CO;2-A.

Hedges, L. V. 1983. A random effects model for effect sizes. *Psychological Bulletin* 93: 388–395. https://doi.org/10.1037/0033-2909.93.2.388.

Paule, R. C., and J. Mandel. 1982. Consensus values and weighting factors. *Journal of Research of the National Bureau of Standards* 87: 377–385. https://doi.org/10.6028/jres.087.022.

Raudenbush, S. W. 2009. "Analyzing effect sizes: Random-effects models". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 295–316. 2nd ed. New York: Russell Sage Foundation.

Schmidt, F. L., and J. E. Hunter. 2015. *Methods of Meta-Analysis: Correcting Error and Bias in Research Findings*. 3rd ed. Thousand Oaks, CA: Sage. https://doi.org/10.4135/9781483398105.

Sidik, K., and J. N. Jonkman. 2005. A note on variance estimation in random effects meta-regression. *Journal of Biopharmaceutical Statistics* 15: 823–838. https://doi.org/10.1081/BIP-200067915.

Veroniki, A. A., D. Jackson, W. Viechtbauer, R. Bender, J. Bowden, G. Knapp, O. Kuss, J. P. T. Higgins, D. Langan, and G. Salanti. 2016. Methods to estimate the between-study variance and its uncertainty in meta-analysis. *Research Synthesis Methods* 7: 55–79. https://doi.org/10.1002/jrsm.1164.

Also see

[META] **meta data** — Declare meta-analysis data

[META] **meta esize** — Compute effect sizes and declare meta-analysis data+

[META] **meta update** — Update, describe, and clear meta-analysis settings

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta update — Update, describe, and clear meta-analysis settings

Description Quick start Menu Syntax
Options Remarks and examples Stored results Also see

Description

`meta update` updates certain components of the meta-analysis after it was declared by `meta set` or `meta esize`. This command is useful for updating some of the meta settings without having to fully respecify your meta-analysis variables. The updated settings will be used throughout the rest of your meta-analysis session.

`meta query` reports whether the data in memory are `meta data` and, if they are, displays the current meta setting information identical to that produced by `meta set` or `meta esize`.

`meta clear` clears meta settings, including meta data characteristics and system variables. The original data remain unchanged. You do not need to use `meta clear` before doing another `meta set` or `meta esize`.

Quick start

Check whether data are declared as `meta` data, and, if they are, describe their current meta-analysis setting information

```
meta query
```

Keep the same meta-analysis setting (specified earlier using `meta set` or `meta esize`), but use a DerSimonian–Laird random-effects model

```
meta update, random(dlaird)
```

Keep the same meta-analysis setting (specified earlier using `meta esize`), but use the log risk-ratio as the effect size

```
meta update, esize(lnrratio)
```

Clear meta-analysis declaration

```
meta clear
```

Menu

Statistics > Meta-analysis

Syntax

Update meta-analysis settings declared using `meta esize` *for two-group comparison of continuous outcomes*

```
meta update [ , options_continuous options ]
```

Update meta-analysis settings declared using `meta esize` *for two-group comparison of binary outcomes*

```
meta update [ , options_binary options ]
```

Update meta-analysis settings declared using `meta esize` *for estimating a single proportion*

```
meta update [ , options_proportion options ]
```

Update meta-analysis settings declared using `meta set`

```
meta update [ , options_generic options ]
```

Describe meta data

```
meta query [ , short ]
```

Clear meta data

```
meta clear
```

options_continuous	Description
`esize(`*esspeccnt*`)`	specify effect size for two-group comparison of continuous outcomes to be used in the meta-analysis
`random[ (`*remethod*`) ]`	random-effects meta-analysis
`common`	common-effect meta-analysis; implies inverse-variance method
`fixed`	fixed-effects meta-analysis; implies inverse-variance method

options_binary	Description
`esize(`*estypebin*`)`	specify effect size for two-group comparison of binary outcomes to be used in the meta-analysis
`random[ (`*remethod*`) ]`	random-effects meta-analysis
`common[ (`*cefemethod*`) ]`	common-effect meta-analysis
`fixed[ (`*cefemethod*`) ]`	fixed-effects meta-analysis
`zerocells(`*zcspec*`)`	adjust for zero cells during computation; default is to add 0.5 to all cells of those 2×2 tables that contain zero cells

options_proportion	Description
esize(*estypeprop*)	specify effect size for estimating a single proportion to be used in the meta-analysis
random[(*remethod*)]	random-effects meta-analysis
common	common-effect meta-analysis; implies inverse-variance method
fixed	fixed-effects meta-analysis; implies inverse-variance method
zerocells(*zcspec*)	adjust for zero cells during computation; default is to add 0.5 to all cells of a study with zero successes or failures

options_generic	Description
random[(*remethod*)]	random-effects meta-analysis
common	common-effect meta-analysis; implies inverse-variance method
fixed	fixed-effects meta-analysis; implies inverse-variance method
studysize(*varname*)	total sample size per study

options	Description
studylabel(*varname*)	variable to be used to label studies in all meta-analysis output
eslabel(*string*)	effect-size label to be used in all meta-analysis output; default is eslabel(Effect size)
level(#)	confidence level for all subsequent meta-analysis commands
[no]metashow	display or suppress meta settings in the output

Options

For meta update options, see *Options* of [META] **meta set** and *Options* of [META] **meta esize**.

short is used with meta query. It displays a short summary of the meta settings containing the information about the declared type of the effect size, effect-size variables and standard error variables, and meta-analysis model and estimation method. This option does not appear in the dialog box.

Remarks and examples

When conducting a meta-analysis, you may wish to explore how your results are affected by modifying certain characteristics of your model. For example, suppose you are using log odds-ratios as your effect sizes and the DerSimonian–Laird random-effects model. You want to investigate how your results would change if you were to use log risk-ratios instead. You could use meta esize, but you would need to respecify all four of your summary-data variables.

```
. meta esize summary_data, esize(lnrratio) random(dlaird)
```

Instead, you can use meta update to simply update the effect sizes.

```
. meta update, esize(lnrratio)
```

meta update will run meta esize keeping all the model components unchanged except for those you specified.

You can use `meta query` to describe the current meta-analysis settings. With `meta` data in memory, `meta query` produces the same output as `meta set` and `meta esize`. If the data in memory are not declared to be `meta` data, `meta query` will report the following:

```
. meta query
(data not meta set; use meta set or meta esize to declare as meta data)
```

To clear meta settings, use `meta clear`.

For more details and examples, see *Modifying default meta settings* and *Displaying and updating meta settings* in [META] **meta data**.

Stored results

`meta update` updates characteristics and contents of system variables described in *Stored results* of [META] **meta set** and *Stored results* of [META] **meta esize**.

Also see

[META] **meta data** — Declare meta-analysis data

[META] **meta esize** — Compute effect sizes and declare meta-analysis data+

[META] **meta set** — Declare meta-analysis data using generic effect sizes

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta forestplot — Forest plots[+]

[+]This command includes features that are part of StataNow.

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Methods and formulas | References
Also see

Description

`meta forestplot` summarizes `meta data` in a graphical format. It reports individual effect sizes and the overall effect size (ES), their confidence intervals (CIs), heterogeneity statistics, and more. `meta forestplot` can perform random-effects (RE), common-effect (CE), and fixed-effects (FE) meta-analyses. It can also perform subgroup, cumulative, and sensitivity meta-analyses. For tabular display of meta-analysis summaries, see [META] **meta summarize**.

Quick start

Default forest plot after data are declared by using either `meta set` or `meta esize`

```
meta forestplot
```

Same as above, but apply the hyperbolic tangent transformation to effect sizes and their CIs

```
meta forestplot, transform(tanh)
```

Add vertical lines at the overall effect-size and no-effect values

```
meta forestplot, esrefline nullrefline
```

Customize the overall effect-size line, and annotate the sides of the plot, with respect to the no-effect line, favoring the treatment or control

```
meta forestplot, esrefline(lcolor(green))   ///
  nullrefline(favorsleft("Favors vaccine") ///
  favorsright("Favors control"))
```

Add a custom diamond with a label for the overall effect-size ML estimate by specifying its value and CI limits

```
meta forestplot, customoverall(-.71 -1.05 -.37, label("{bf:ML Overall}"))
```

Forest plot based on subgroup meta-analysis

```
meta forestplot, subgroup(groupvar)
```

Forest plot based on cumulative meta-analysis

```
meta forestplot, cumulative(ordervar)
```

Forest plot based on leave-one-out meta-analysis

```
meta forestplot, leaveoneout
```

Default forest plot after data are declared with `meta set` but with the columns spelled out

```
meta forestplot _id _plot _esci _weight
```

Default forest plot after data are declared with `meta esize` but with the columns spelled out

```
meta forestplot _id _data _plot _esci _weight
```

Same as above, but with the weights omitted

```
meta forestplot _id _data _plot _esci
```

Same as above, but the columns are rearranged

```
meta forestplot _id _data _esci _plot
```

Same as above, but plot variables x1 and x2 as the second and last columns

```
meta forestplot _id x1 _data _esci _plot x2
```

Change the format of the _esci column

```
meta forestplot, columnopts(_esci, format(%7.4f))
```

Menu

Statistics > Meta-analysis

Syntax

```
meta forestplot [column_list] [if] [in] [, options]
```

column_list is a list of column names given by *col*. In the *Meta-Analysis Control Panel*, the columns can be specified on the **Forest plot** tab of the Forest plot pane.

options	Description
Main	
`random[(`*remethod*`)]`	random-effects meta-analysis
`common[(`*cefemethod*`)]`	common-effect meta-analysis
`fixed[(`*cefemethod*`)]`	fixed-effects meta-analysis
reopts	random-effects model options
`subgroup(`*varlist*`)`	subgroup meta-analysis for each variable in *varlist*
`cumulative(`*cumulspec*`)`	cumulative meta-analysis
`leaveoneout`	leave-one-out meta-analysis
Options	
`level(`#`)`	set confidence level; default is as declared for meta-analysis
`citype(`*citype*`)`	specify the type of study CI (for meta-analysis of a single proportion)
`proportion`	report proportions (for meta-analysis of a single proportion)
`prevalence`	synonym for `proportion` but labels the effect sizes as `Prevalence` in the output
+`correlation`	report correlations (for meta-analysis of correlations)
eform_option	report exponentiated results
`transform(`*transfspec*`)`	report transformed results
`sort(`*varlist*`[, ...])`	sort studies according to *varlist*
`tdistribution`	report t test instead of z test
`[no]metashow`	display or suppress meta settings in the output
Maximization	
maximize_options	control the maximization process; seldom used
Forest plot	
`columnopts(`*col*`, [`*colopts*`])`	column options; can be repeated
`cibind(`*bind*`)`	change binding of CIs for columns `_esci` and `_ci`; default is `cibind(brackets)`
`sebind(`*bind*`)`	change binding of standard errors for column `_esse`; default is `sebind(parentheses)`
`nohrule`	suppress horizontal rule
`hruleopts(`*hrule_options*`)`	change look of horizontal rule
text_options	change looks of text options such as column titles, supertitles, and more
plot_options	change look or suppress markers, restrict range of CIs, and more
test_options	suppress information about heterogeneity statistics and tests
graph_options	change the lines, labels, ticks, titles, scheme, etc. on the forest plot
`nooverall`	suppress row corresponding to the overall effect size
`olabel(`*string*`)`	modify default overall effect-size label under the `_id` column; default label is `Overall`

+These features are part of StataNow.

`nooverall` and `olabel()` do not appear in the dialog box.

col	Description
Default columns and order	
_id	study label
_data	summary data; _data1 and _data2 for two-group comparisons of continuous and binary outcomes (only after meta esize)
_plot	forest graph
_esci	effect size and its confidence interval
_weight	percentage of total weight given to each study
Summary-data columns and order	
Two-sample continuous data	
Treatment group	
_data1	summary data for treatment group; _n1, _mean1, and _sd1
_n1	sample size in the treatment group
_mean1	mean in the treatment group
_sd1	standard deviation in the treatment group
Control group	
_data2	summary data for control group; _n2, _mean2, and _sd2
_n2	sample size in the control group
_mean2	mean in the control group
_sd2	standard deviation in the control group
Two-sample binary data	
Treatment group	
_data1	summary data for treatment group; _a and _b
_a	number of successes in the treatment group
_b	number of failures in the treatment group
Control group	
_data2	summary data for control group; _c and _d
_c	number of successes in the control group
_d	number of failures in the control group
One-sample binary data	
_data	summary data; _e and _n
_e	number of successes
_n	study sample size
Correlation data (StataNow)	
_data	summary data; _r and _n
_r	correlation
_n	study sample size

Other columns

_es	effect size
_ci	confidence interval for effect size
_lb	lower confidence limit for effect size
_ub	upper confidence limit for effect size
_se	standard error of effect size
_esse	effect size and its standard error
_pvalue	*p*-value for significance test with subgroup(), cumulative(), or leaveoneout
_K	number of studies with subgroup()
_size	within-group sample size with subgroup()
_order	order variable for cumulative meta-analysis with cumulative()
varname	variable in the dataset (except meta system variables)

Columns _data, _data1, _data2, and the other corresponding data columns are not available after the declaration by using meta set.

Columns _n1, _mean1, _sd1, _n2, _mean2, and _sd2 are available only after the declaration by using meta esize for a two-group comparison of continuous outcomes.

Columns _a, _b, _c, and _d are available only after the declaration by using meta esize for a two-group comparison of binary outcomes.

Columns _e and _n are available only after the declaration by using meta esize for estimating a single proportion.

Columns _r and _n are available only after the declaration by using meta esize for correlation data.

Column _pvalue is available only when option subgroup() with multiple variables is specified or when cumulative() or leaveoneout is specified.

Columns _K and _size are available only when option subgroup() with multiple variables is specified.

Column *varname* is not available when option subgroup() with multiple variables is specified.

colopts	Description
supertitle(*string*)	super title specification
title(*string*)	title specification
format(*%fmt*)	numerical format for column items
mask(*mask*)	string mask for column items
plotregion(*region_options*)	attributes of plot region
textbox_options	appearance of textboxes

text_options	Description
coltitleopts(*textbox_options*)	change look of column titles and supertitles
itemopts(*textbox_options*)	change look of study rows
overallopts(*textbox_options*)	change look of the overall row
groupopts(*textbox_options*)	change look of subgroup rows
bodyopts(*textbox_options*)	change look of study, subgroup, and overall rows
nonotes	suppress notes about the meta-analysis model, method, and more

plot_options	Description
crop(# #)	restrict the range of CI lines
ciopts(*ci_options*)	change look of CI lines (size, color, etc.)
nowmarkers	suppress weighting of study markers
nomarkers	suppress study markers
markeropts(*marker_options*)	change look of study markers (size, color, etc.)
noomarker	suppress the overall marker
omarkeropts(*marker_options*)	change look of the overall marker (size, color, etc.)
nogmarkers	suppress subgroup markers
gmarkeropts(*marker_options*)	change look of subgroup markers (size, color, etc.)
insidemarker[(*marker_options*)]	add a marker at the center of the study marker
esrefline[(*line_options*)]	add a vertical line corresponding to the overall effect size
noesrefline	suppress vertical line corresponding to the overall effect size plotted on leave-one-out forest plot
nullrefline[(*nullopts*)]	add a vertical line corresponding to no effect
customoverall(*customspec*)	add a custom diamond representing an overall effect; can be repeated

test_options	Description
ohetstatstext(*string*)	modify default text for overall heterogeneity statistics
noohetstats	suppress overall heterogeneity statistics
ohomtesttext(*string*)	modify default text for overall homogeneity test
noohomtest	suppress overall homogeneity test
osigtesttext(*string*)	modify default text for test of significance of overall effect size
noosigtest	suppress test of significance of overall effect size
ghetstats#text(*string*)	modify default text for subgroup heterogeneity statistics in the #th subgroup
noghetstats	suppress subgroup heterogeneity statistics
gwhomtest#text(*string*)	modify default text for within-subgroup homogeneity test in the #th subgroup
nogwhomtests	suppress within-subgroup homogeneity tests
gsigtest#text(*string*)	modify default text for test of significance of the subgroup effect size in the #th subgroup
nogsigtests	suppress tests of significance of subgroup effect size
gbhomtest#text(*string*)	modify default text for between-subgroup homogeneity test in the #th subgroup
nogbhomtests	suppress between-subgroup homogeneity tests

graph_options	Description
xline(*linearg*)	add vertical lines at specified x values
xtitle(*axis_title*)	specify x-axis title
xlabel(*rule_or_values*)	major ticks plus labels
xtick(*rule_or_values*)	major ticks only
xmlabel(*rule_or_values*)	minor ticks plus labels
xmtick(*rule_or_values*)	minor ticks only
title(*tinfo*)	overall title
subtitle(*tinfo*)	subtitle of title
note(*tinfo*)	note about graph
caption(*tinfo*)	explanation of graph
t1title(*tinfo*) t2title(*tinfo*)	rarely used
b1title(*tinfo*) b2title(*tinfo*)	rarely used
l1title(*tinfo*) l2title(*tinfo*)	vertical text
r1title(*tinfo*) r2title(*tinfo*)	vertical text
scheme(*schemename*)	overall look
nodraw	suppress display of graph
name(*name*, ...)	specify name for graph
saving(*filename*, ...)	save graph in file

nullopts	Description
favorsleft(*string*[, *textbox_options*])	add a label to the left of the no-effect reference line
favorsright(*string*[, *textbox_options*])	add a label to the right of the no-effect reference line
line_options	affect the rendition of the no-effect reference line

Options

Main

random[(*remethod*)], common[(*cefemethod*)], fixed[(*cefemethod*)], subgroup(*varlist*), cumulative(*cumulspec*), and leaveoneout; see *Options* in [META] **meta summarize**.

reopts are tau2(#), i2(#), predinterval, predinterval(#[, *line_options*]), and se(*seadj*). These options are used with random-effects meta-analysis. See *Options* in [META] **meta summarize**.

predinterval and predinterval(#[, *line_options*]) draw whiskers extending from the overall effect marker and spanning the width of the prediction interval. *line_options* affect how the whiskers are rendered; see [G-3] ***line_options***.

Options

level(#), citype(), proportion, prevalence, correlation, *eform_option*, transform(), sort(*varlist*[, ...]), tdistribution, and [no]metashow; see *Options* in [META] **meta summarize**.

Maximization

maximize_options: iterate(#), tolerance(#), nrtolerance(#), nonrtolerance, from(#), and showtrace; see *Options* in [META] **meta summarize**.

Forest plot

columnopts(*col* [, *colopts*]) changes the look of the column identified by *col*. This option can be repeated.

colopts are the following options:

supertitle(*string*) specifies that the column's supertitle is *string*.

title(*string*) specifies that the column's title is *string*.

format(*%fmt*) specifies the format for the column's numerical values.

mask(*mask*) specifies a string composed of formats for the column's statistics. For example, *mask* for column _weight that identifies the column of weight percentages may be specified as "%6.2f %%".

plotregion(*region_options*) modifies attributes for the plot region. You can change the margins, background color, an outline, and so on; see [G-3] ***region_options***.

textbox_options affect how the column's items (study and group) are rendered. These options override what is specified in global options bodyopts(), itemopts(), and groupopts(). See [G-3] ***textbox_options***.

Options format(), mask(), and *textbox_options* are ignored by _plot.

cibind(*bind*) changes the binding of the CIs for columns _esci and _ci. *bind* is one of brackets, parentheses, or none. By default, the CIs are bound by using brackets, cibind(brackets). This option is relevant only when _esci or _ci appears in the plot.

sebind(*bind*) changes the binding of the standard errors for column _esse. *bind* is one of parentheses, brackets, or none. By default, the standard errors are bound by using parentheses, cibind(parentheses). This option is relevant only when _esse appears in the plot.

nohrule suppresses the horizontal rule.

hruleopts(*hrule_options*) affects the look of the horizontal rule.

hrule_options are the following options:

lcolor(*colorstyle*) specifies the color of the rule; see [G-4] ***colorstyle***.

lwidth(*linewidthstyle*) specifies the width of the rule; see [G-4] ***linewidthstyle***.

lalign(*linealignmentstyle*) specifies the alignment of the rule; see [G-4] ***linealignmentstyle***.

lpattern(*linepatternstyle*) specifies the line pattern of the rule; see [G-4] ***linepatternstyle***.

lstyle(*linestyle*) specifies the overall style of the rule; see [G-4] ***linestyle***.

margin(*marginstyle*) specifies the margin of the rule; see [G-4] ***marginstyle***.

text_options are the following options:

coltitleopts(*textbox_options*) affects the look of text for column titles and supertitles. See [G-3] ***textbox_options***.

`itemopts(`*textbox_options*`)` affects the look of text for study rows; see [G-3] ***textbox_options***. This option is ignored when option `subgroup()` is specified and contains multiple variables or when option `cumulative()` or `leaveoneout` is specified.

`overallopts(`*textbox_options*`)` affects the look of text for the overall row. See [G-3] ***textbox_options***.

`groupopts(`*textbox_options*`)` (synonym `subgroupopts()`) affects the look of text for subgroup rows when option `subgroup()` is specified. See [G-3] ***textbox_options***.

`bodyopts(`*textbox_options*`)` affects the look of text for study, subgroup, and overall rows. See [G-3] ***textbox_options***.

`nonotes` suppresses the notes displayed on the graph about the specified meta-analysis model and method and the standard error adjustment.

plot_options are the following options:

`crop(`$\#_1$ $\#_2$`)` restricts the range of the CI lines to be between $\#_1$ and $\#_2$. A missing value may be specified for any of the two values to indicate that the corresponding limit should not be cropped. Otherwise, lines that extend beyond the specified value range are cropped and adorned with arrows. This option is useful in the presence of small studies with large standard errors, which lead to confidence intervals that are too wide to be displayed nicely on the graph. Option `crop()` may be used to handle this case.

`ciopts(`*ci_options*`)` affects the look of the CI lines and, in the presence of cropped CIs (see option `crop()`), arrowheads.

ci_options are any options documented in [G-3] ***line_options*** and the following options of [G-2] **graph twoway pcarrow**: `mstyle()`, `msize()`, `mangle()`, `barbsize()`, `mcolor()`, `mfcolor()`, `mlcolor()`, `mlwidth()`, `mlstyle()`, and `color()`.

`nowmarkers` suppresses weighting of the study markers.

`nomarkers` suppresses the study markers.

`markeropts(`*marker_options*`)` affects the look of the study markers.

marker_options: `msymbol()`, `mcolor()`, `mfcolor()`, `mlcolor()`, `mlwidth()`, `mlalign()`, `mlstyle()`, and `mstyle()`; see [G-3] ***marker_options***.

`nowmarkers`, `nomarkers`, and `markeropts()` are ignored when option `subgroup()` is specified and contains multiple variables or when option `cumulative()` or `leaveoneout` is specified.

`noomarker` suppresses the overall marker.

`omarkeropts(`*marker_options*`)` affects the look of the overall marker.

marker_options: `mcolor()`, `mfcolor()`, `mlcolor()`, `mlwidth()`, `mlalign()`, `mlstyle()`, and `mstyle()`; see [G-3] ***marker_options***.

`nogmarkers` suppresses the subgroup markers.

`gmarkeropts(`*marker_options*`)` affects the look of the subgroup markers.

marker_options: `mcolor()`, `mfcolor()`, `mlcolor()`, `mlwidth()`, `mlalign()`, `mlstyle()`, and `mstyle()`; see [G-3] ***marker_options***.

`nogmarkers` and `gmarkeropts()` are ignored when option `subgroup()` is not specified.

insidemarker and insidemarker(*marker_options*) add markers at the center of study markers. *marker_options* control how the added markers are rendered.

marker_options: msymbol(), mcolor(), mfcolor(), mlcolor(), mlwidth(), mlalign(), mlstyle(), and mstyle(); see [G-3] ***marker_options***.

insidemarker() is not allowed when option subgroup() is specified and contains multiple variables or when option cumulative() or leaveoneout is specified.

esrefline and esrefline(*line_options*) specify that a vertical line be drawn at the value corresponding to the overall effect size. The optional *line_options* control how the line is rendered; see [G-3] ***line_options***.

noesrefline suppresses the overall effect-size line plotted by default on the leave-one-out forest plot, which is produced when you specify option leaveoneout.

nullrefline and nullrefline(*nullopts*) specify that a vertical line be drawn at the value corresponding to no overall effect. *nullopts* are the following options:

favorsleft(*string*[, *textbox_options*]) adds a label, *string*, to the left side (with respect to the no-effect line) of the forest graph. *textbox_options* affect how *string* is rendered; see [G-3] ***textbox_options***.

favorsright(*string*[, *textbox_options*]) adds a label, *string*, to the right side (with respect to the no-effect line) of the forest graph. *textbox_options* affect how *string* is rendered; see [G-3] ***textbox_options***.

favorsleft() and favorsright() are typically used to annotate the sides of the forest graph (column _plot) favoring the treatment or control.

line_options affect the rendition of the vertical line; see [G-3] ***line_options***.

customoverall(*customspec*) draws a custom-defined diamond representing an overall effect size. This option can be repeated. *customspec* is *#es #lb #ub* [, *customopts*], where *#es*, *#lb*, and *#ub* correspond to an overall effect-size estimate and its lower and upper CI limits, respectively. *customopts* are the following options:

label(*string*) adds a label, *string*, under the _id column describing the custom diamond.

textbox_options affect how label(*string*) is rendered; see [G-3] ***textbox_options***.

marker_options affect how the custom diamond is rendered. *marker_options* are mcolor(), mfcolor(), mlcolor(), mlwidth(), mlalign(), mlstyle(), and mstyle(); see [G-3] ***marker_options***.

Option customoverall() may not be combined with option cumulative() or leaveoneout.

test_options are defined below. These options are not relevant with cumulative and leave-one-out meta-analysis.

ohetstatstext(*string*) modifies the default text for the overall heterogeneity statistics reported under the Overall row heading on the plot.

noohetstats suppresses overall heterogeneity statistics reported under the Overall row heading on the plot.

ohomtesttext(*string*) modifies the default text for the overall homogeneity test labeled as Test of θ_i=θ_j under the Overall row heading on the plot.

noohomtest suppresses the overall homogeneity test labeled as Test of θ_i=θ_j under the Overall row heading on the plot.

osigtesttext(*string*) modifies the default text of the test of significance of the overall effect size labeled as Test of θ=0 under the Overall row heading on the plot.

noosigtest suppresses the test of significance of the overall effect size labeled as Test of θ=0 under the Overall row heading on the plot.

ghetstats#text(*string*) modifies the default text for the heterogeneity statistics in the #th subgroup. These statistics are reported under the group-specific row headings when a single subgroup analysis is performed, that is, when option subgroup() is specified with one variable.

noghetstats suppresses subgroup heterogeneity statistics reported when a single subgroup analysis is performed, that is, when option subgroup() is specified with one variable. These statistics are reported under the group-specific row headings.

gwhomtest#text(*string*) modifies the default text for the within-subgroup homogeneity test in the #th subgroup. This test is reported when a single subgroup analysis is performed, that is, when option subgroup() is specified with one variable. The test is labeled as Test of θ_i=θ_j under the group-specific row headings.

nogwhomtests suppresses within-subgroup homogeneity tests. These tests investigate the differences between effect sizes of studies within each subgroup. These tests are reported when a single subgroup analysis is performed, that is, when option subgroup() is specified with one variable. The tests are labeled as Test of θ_i=θ_j under the group-specific row headings.

gsigtest#text(*string*) modifies the default text for the test of significance of the subgroup effect size labeled as Test of θ=0 in the #th subgroup.

nogsigtests suppresses tests of significance of the subgroup effect size labeled as Test of θ=0 within each subgroup. These tests are reported when a single subgroup analysis is performed, that is, when option subgroup() is specified with one variable.

gbhomtest#text(*string*) modifies the default text for the between-subgroup homogeneity test in the #th subgroup. The #th between-subgroup homogeneity test corresponds to the #th variable specified within option subgroup(). The test is labeled as Test of group differences on the plot.

nogbhomtests suppresses between-subgroup homogeneity tests. These tests investigate the differences between the subgroup effect sizes reported when any subgroup analysis is performed, that is, when option subgroup() is specified. The tests are labeled as Test of group differences on the plot.

graph_options: xline(), xtitle(), xlabel(), xtick(), xmlabel(), xmtick(), title(), subtitle(), note(), caption(), t1title(), t2title(), b1title(), b2title(), l1title(), l2title(), r1title(), r2title(), scheme(), nodraw, name(), and saving(); see [G-3] ***twoway_options*** for details.

The following options are available with meta forestplot but are not shown in the dialog box:

nooverall suppresses the row corresponding to the overall effect size in the forest plot.

olabel(*string*) modifies the default overall effect-size label under the _id column, which, by default, is Overall.

Remarks and examples

Remarks are presented under the following headings:

Overview
Using meta forestplot
Plot columns
Examples of using meta forestplot

Overview

Meta-analysis results are often presented using a forest plot (for example, Lewis and Ellis [1982]). A forest plot shows effect-size estimates and their confidence intervals for each study and, usually, the overall effect size from the meta-analysis (for example, Lewis and Clarke [2001]; Harris et al. [2016]; and Fisher 2016). Each study is represented by a square with the size of the square being proportional to the study weight; that is, larger squares correspond to larger (more precise) studies. The weights depend on the chosen meta-analysis model and method. Studies' CIs are plotted as whiskers extending from each side of the square and spanning the width of the CI. Heterogeneity measures such as the I^2 and H^2 statistics, homogeneity test, and the significance test of the overall effect sizes are also commonly reported.

A subgroup meta-analysis forest plot also shows group-specific results. Additionally, it reports a test of the between-group differences among the overall effect sizes. A cumulative meta-analysis forest plot shows the overall effect sizes and their CIs by accumulating the results from adding one study at a time to each subsequent analysis. Similarly, a leave-one-out meta-analysis forest plot shows the overall effect sizes and their CIs resulting from meta-analyses omitting one study at a time. By convention, group-specific and overall effect sizes are represented by diamonds centered on their estimated values with the diamond width corresponding to the CI length.

For more details about forest plots, see, for instance, Anzures-Cabrera and Higgins (2010). Also see Schriger et al. (2010) for an overview of their use in practice.

Using meta forestplot

`meta forestplot` produces meta-analysis forest plots. It provides a graphical representation of the results produced by `meta summarize` and thus supports most of its options such as those specifying a meta-analysis model and estimation method; see [META] **meta summarize**.

The default look of the forest plot produced by `meta forestplot` depends on the type of analysis. For basic meta-analysis, `meta forestplot` plots the study labels, effect sizes and their confidence intervals, and percentages of total weight given to each study. If `meta esize` was used to declare `meta data`, summary data are also plotted. That is,

```
. meta forestplot
```

is equivalent to typing

```
. meta forestplot _id _plot _esci _weight
```

after declaration by using `meta set` and to typing

```
. meta forestplot _id _data _plot _esci _weight
```

after declaration by using `meta esize`.

If multiple variables are specified in the subgroup() option,

```
. meta forestplot, subgroup(varlist)
```

is equivalent to typing

```
. meta forestplot _id _K _plot _esci _pvalue, subgroup(varlist)
```

For cumulative meta-analysis,

```
. meta forestplot, cumulative(varname)
```

is equivalent to typing

```
. meta forestplot _id _plot _esci _pvalue _order, cumulative(varname)
```

For leave-one-out meta-analysis,

```
. meta forestplot, leaveoneout
```

is equivalent to typing

```
. meta forestplot _id _plot _esci _pvalue, leaveoneout
```

You can also specify any of the supported columns with meta forestplot, including variables in your dataset. For example, you may include, say, variables x1 and x2, as columns in the forest plot by simply specifying them in the column list,

```
. meta forestplot _id x1 _plot _esci _weight x2
```

See *Plot columns* for details about the supported columns.

The CIs correspond to the confidence level as declared by meta set or meta esize. You can specify a different level in the level() option. Also, by default, the CIs are bound by using brackets. You can specify cibind(parentheses) to use parentheses instead.

As we mentioned earlier, you can produce forest plots for subgroup analysis by using the subgroup() option, for cumulative meta-analysis by using the cumulative() option, and for leave-one-out meta-analysis by using the leaveoneout option.

You can modify the default column supertitles, titles, formats, and so on by specifying the columnopts() option. You can repeat this option to modify the look of particular columns. If you want to apply the same formatting to multiple columns, you can specify these columns within columnopts(). See *Options* for the list of supported column options.

Options esrefline() and nullrefline() are used to draw vertical lines at the overall effect-size and no-effect values, respectively. For the leave-one-out forest plot, produced when you specify option leaveoneout, the overall effect-size line is drawn by default. You may suppress it by using option noesrefline. Suboptions favorsleft() and favorsright() of nullrefline() may be specified to annotate the sides (with respect to the no-effect line) of the plot favoring treatment or control.

Another option you may find useful is crop($\#_1$ $\#_2$). Sometimes, some of the smaller studies may have large standard errors that lead to CIs that are too wide to be displayed on the plot. You can "crop" such CIs by restricting their range. The restricted range will be indicated by the arrowheads at the corresponding ends of the CIs. You can crop both limits or only one of the limits. You can modify the default look of the arrowheads or CI lines, in general, by specifying ciopts().

You may sometimes want to show the overall effect-size estimates from multiple meta-analysis models (for example, common versus random), from different estimation methods (REML versus DL), or for specific values of moderators from a meta-regression. This may be accomplished via the `customoverall(`*#es #lb #ub* [`,`*customopts*]`)` option. This option may be repeated to display multiple diamonds depicting multiple custom-defined overall effect sizes.

You can specify many more options to customize the look of your forest plot such as modifying the look of text for column titles in `coltitleopts()` or the column format in `format()`; see *Syntax* for details.

`meta forestplot` uses the following default convention when displaying the results. The results from individual studies—individual effects sizes—are plotted as blue squares with areas proportional to study weights. The overall effect size is plotted as a green (or, more precisely, forest green using Stata's color convention) diamond with the width corresponding to its CI. The results of a single subgroup analysis—subgroup effect sizes—are plotted as red diamonds with the widths determined by the respective CIs. The results of multiple subgroup analyses are plotted as red circles with the CI lines. The cumulative meta-analysis results—cumulative overall effect sizes—are displayed as green circles with CI lines. Similarly, the leave-one-out meta-analysis results—overall effect size with one study omitted—are also displayed as green circles with CI lines.

Options `itemopts()`, `nomarkers`, and `markeropts()` control the look of study rows and markers, which represent individual effect sizes. These options are not relevant when individual studies are not reported such as with multiple subgroup analysis, cumulative meta-analysis, and leave-one-out meta-analysis.

Options `groupopts()`, `nogmarkers`, and `gmarkeropts()` control the look of subgroup rows and markers and are relevant only when subgroup analysis is performed by specifying the `subgroup()` option.

Options `overallopts()`, `noomarker`, and `omarkeropts()` control the look of overall rows and markers, which represent the overall effect sizes. These options are always applicable because the overall results are always displayed by default. With cumulative and leave-one-out meta-analysis, these options affect the displayed overall effect sizes.

Graphs created by `meta forestplot` cannot be combined with other Stata graphs using `graph combine`.

Plot columns

`meta forestplot` supports many columns that you can include in your forest plot; see the list of supported columns in *Syntax*. The default columns plotted for various analyses were described in *Using meta forestplot* above. Here we provide more details about some of the supported columns.

`meta forestplot` provides individual columns such as `_es` and `_se` and column shortcuts such as `_esse`. Column shortcuts are typically shortcuts for specifying multiple columns. For instance, when dealing with two-group comparison of binary or continuous outcomes, column `_data` is a shortcut for columns `_data1` and `_data2`, which themselves are shortcuts to individual summary-data columns. For a two-group comparison of continuous outcomes, `_data1` is a shortcut for columns `_n1`, `_mean1`, and `_sd1`, and `_data2` is a shortcut for `_n2`, `_mean2`, and `_sd2`. For a two-group comparison of binary outcomes, `_data1` corresponds to the treatment-group numbers of successes and failures, `_a` and `_b`, and `_data2` to the respective numbers in the control group, `_c` and `_d`. For estimating a single proportion, the case of one-sample binary data, columns `_data1` and `_data2` are not available. In this case, column `_data` corresponds to columns `_e` and `_n`, which are the number of successes and the study sample

size, respectively. Similarly, for correlation data, column _data corresponds to columns _r and _n, which are the correlation and the study sample size, respectively. Column _data and the corresponding summary-data columns are available only after declaration by using meta esize.

The other column shortcuts are _ci, _esci, and _esse. In addition to serving as shortcuts to the respective columns (_lb and _ub; _es, _lb, and _ub; and _es and _se), these shortcut columns have additional properties. For instance, when you specify _ci, the lower and upper CI bounds are separated with a comma, bounded in brackets, and share a title. That is,

```
. meta forestplot _ci
```

is similar to specifying

```
. meta forestplot _lb _ub,
> columnopts(_lb _ub, title(95% CI))
> columnopts(_lb, mask("[%6.2f,"))
> columnopts(_ub, mask("%6.2f]"))
```

Similarly, _esci additionally combines _es and _ci with the common column title, and _esse combines _es and _se and bounds the standard errors in parentheses. _ci, _esci, and _esse also apply other properties to improve the default look of the specified columns such as modifying the default column margins by specifying plotregion(margin()).

If you want to modify the individual columns of the shortcuts, you need to specify the corresponding column names in columnopts(). For instance, if we want to display the effect sizes of the _esci column with three decimal digits but continue using the default format for CIs, we can type

```
. meta forestplot _esci, columnopts(_es, format(%6.3f))
```

If we specify _esci instead of _es in columnopts(),

```
. meta forestplot _esci, columnopts(_esci, format(%6.3f))
```

both effect sizes and CIs will be displayed with three decimal digits. On the other hand, if we want to change the default title and supertitle for _esci, we should specify _esci in columnopts(),

```
. meta forestplot _esci, columnopts(_esci, supertitle("My ES") title("with my CI"))
```

Also see example 7 and example 8 for more examples of customizing the default look of columns.

Column _plot corresponds to the plot region that contains graphical representation of the effect sizes and their confidence intervals. You can modify the default look of the plot by specifying the *plot_options* in *Syntax*.

Column _es corresponds to the plotted effect sizes. For basic meta-analysis, this column displays the individual and overall effect sizes. For subgroup meta-analysis, it also displays subgroup-specific overall effect sizes. For cumulative meta-analysis, it displays the overall effect sizes corresponding to the accumulated studies. For leave-one-out meta-analysis, it displays the overall effect sizes corresponding to the meta-analyses omitting one study at a time.

Some of the columns such as _pvalue, _K, _size, and _order are available only with specific meta-analyses. _pvalue is available only with multiple subgroup analyses, with cumulative analysis, or with leave-one-out analysis; it displays the p-values of the significant tests of effect sizes. _K is available with multiple subgroup analyses and displays the number of studies within each subgroup. _order is available only with cumulative meta-analysis; it displays the values of the specified ordering variable.

You may also add variables in your dataset to the forest plot. For instance, if you want to display variables x1 and x2 in the second and last columns, you may type

```
. meta forestplot _id x1 _plot _esci _weight x2
```

Duplicate columns are ignored with meta forestplot. Also, column shortcuts take precedence. That is, if you specified both _es and _esci, the latter will be displayed.

Examples of using meta forestplot

In this section, we demonstrate some of the uses of meta forestplot. The examples are presented under the following headings:

Example 1: Forest plot for two-group comparison of binary outcomes
Example 2: Subgroup-analysis forest plot
Example 3: Cumulative forest plot
Example 4: Leave-one-out forest plot
Example 5: Forest plot for precomputed effect sizes
Example 6: Multiple subgroup-analyses forest plot
Example 7: Modifying columns' order and cropping confidence intervals
Example 8: Applying transformations and changing titles and supertitles
Example 9: Changing columns' formatting
Example 10: Changing axis range and adding center study markers
Example 11: Prediction intervals and sides favoring control or treatment
Example 12: Adding custom columns and overall effect sizes
Example 13: Forest plot for meta-analysis of a single proportion
Example 14: Increasing plot-region margin
Example 15: Prediction intervals with subgroup analysis and eliminating space in the _esci column
Example 16: Modifying default text for heterogeneity statistics and statistical tests

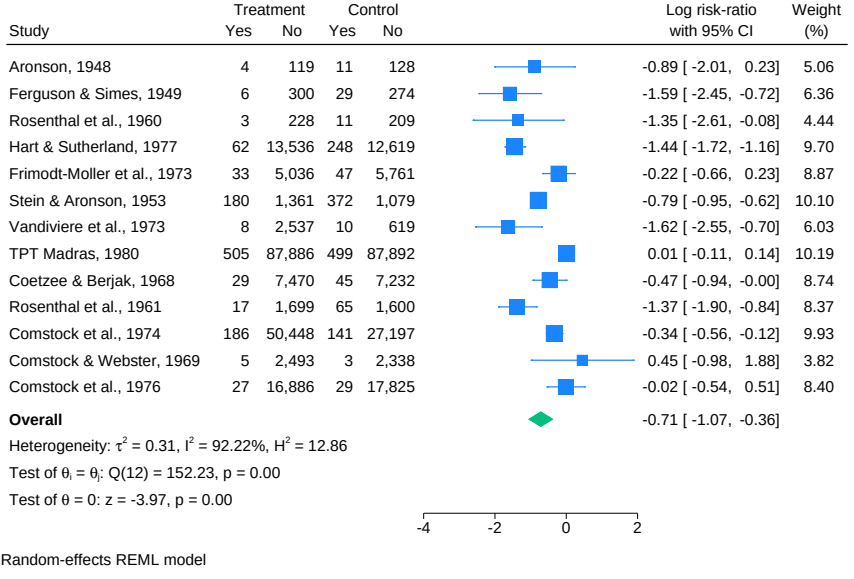
▷ Example 1: Forest plot for two-group comparison of binary outcomes

Consider the dataset from Colditz et al. (1994) of clinical trials that studies the efficacy of a Bacillus Calmette-Guérin (BCG) vaccine in the prevention of tuberculosis (TB). This dataset was introduced in *Efficacy of BCG vaccine against tuberculosis (bcg.dta)* of [META] **meta**. In this section, we use its declared version and focus on the demonstration of various options of meta forest.

```
. use https://www.stata-press.com/data/r18/bcgset
(Efficacy of BCG vaccine against tuberculosis; set with -meta esize-)
```

Let's construct a basic forest plot by simply typing

```
. meta forestplot
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
        Study label: studylbl
```



By default, the basic forest plot displays the study labels (column `_id`), the summary data (`_data`), graphical representation of the individual and overall effect sizes and their CIs (`_plot`), the corresponding values of the effect sizes and CIs (`_esci`), and the percentages of total weight for each study (`_weight`). You can also customize the columns on the forest plot; see example 7 and example 12.

In the graph, each study corresponds to a blue square centered at the point estimate of the effect size with a horizontal line (whiskers) extending on either side of the square. The centers of the squares (the values of study effect sizes) may be highlighted via the `insidemarker()` option; see example 10. The horizontal line depicts the CI. The area of the square is proportional to the corresponding study weight.

The overall effect size corresponds to the green diamond centered at the estimate of the overall effect size. The width of the diamond corresponds to the width of the overall CI. Note that the height of the diamond is irrelevant. It is customary in meta-analysis forest plots to display an overall effect size as a diamond filled inside with color. This, however, may overemphasize the actual area of the diamond whereas only the width of it matters. If desired, you may suppress the fill color by specifying the `omarkeropts(mfcolor(none))` option.

Under the diamond, three lines are reported. The first line contains heterogeneity measures I^2, H^2, and $\hat{\tau}^2$. The second line displays the homogeneity test based on the Q statistic. The third line displays the test of the overall effect size being equal to zero. These lines may be suppressed by specifying options `noohetstats`, `noohomtest`, and `noosigtest`. Alternatively, the default text in these lines may be modified via options `ohetstatstext()`, `ohomtesttext()`, and `osigtesttext()`, respectively; see example 16. See [META] **meta summarize** for a substantive interpretation of these results.

Some forest plots show vertical lines at the no-effect and overall effect-size values. These may be added to the plot via options `nullrefline()` and `esrefline()`, respectively; see example 5. Also, you may sometimes want to plot custom-defined overall effect sizes such as based on multiple meta-analysis models. This may be accomplished via the `customoverall()`; see example 12.

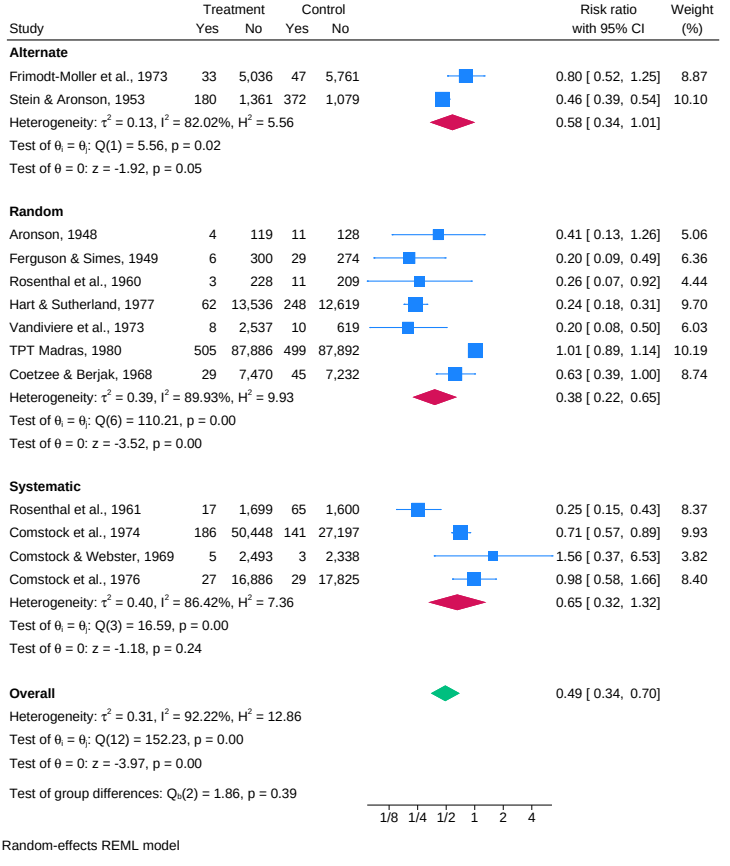
`meta forestplot` provides a quick way to assess between-study heterogeneity visually. In the absence of heterogeneity, we would expect to see that the middle points of the squares are close to the middle of the diamond and the CIs are overlapping. In these data, there is certainly evidence of some heterogeneity because the squares for some studies are far away from the diamond and there are studies with nonoverlapping CIs.

◁

▷ Example 2: Subgroup-analysis forest plot

Continuing with example 1, let's now perform a subgroup meta-analysis based on the method of treatment allocation recorded in variable `alloc`. We specify `subgroup(alloc)` and also use the `eform` option to display exponentiated results, risk ratios (RRs) instead of log risk-ratios in our example.

```
. meta forestplot, subgroup(alloc) eform
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
        Study label: studylbl
```



In addition to the overall results, the forest plot shows the results of meta-analysis for each of the three groups. With subgroup meta-analysis, each group gets its own red diamond marker that represents the group-specific overall effect size. Just like with the overall diamond, only the widths (not the heights) of the group-specific diamonds are relevant on the plot. Similarly to the overall marker, you can specify the `gmarkeropts(mfcolor(none))` option to suppress the fill color for the group-specific diamonds.

Heterogeneity measures, homogeneity tests, and significance tests are reported at the bottom (below the group-specific diamond marker) within each group. These provide information regarding the heterogeneity among the studies within each group and the statistical significance of the group-specific overall effect size. They may be suppressed with options `noghetstats`, `nogwhomtests`, and `nogsigtests`, respectively. Alternatively, you may specify options `ghetstats#text()`, `gwhomtest#text()`, and `gsigtest#text()` to modify the default text reported within the #th subgroup (# can be 1, 2, or 3 in this case); see example 16.

A test of between-group differences based on the Q_b statistic is reported at the bottom. This test investigates the difference between the group-specific overall effect sizes. It may be suppressed with `nogbhomtests`. Alternatively, the default text for this test may be modified using option `gbhomtest#text()`; see example 16.

You may also specify multiple variables in `subgroup()`, in which case a separate subgroup analysis is performed for each variable; see example 6 for details.

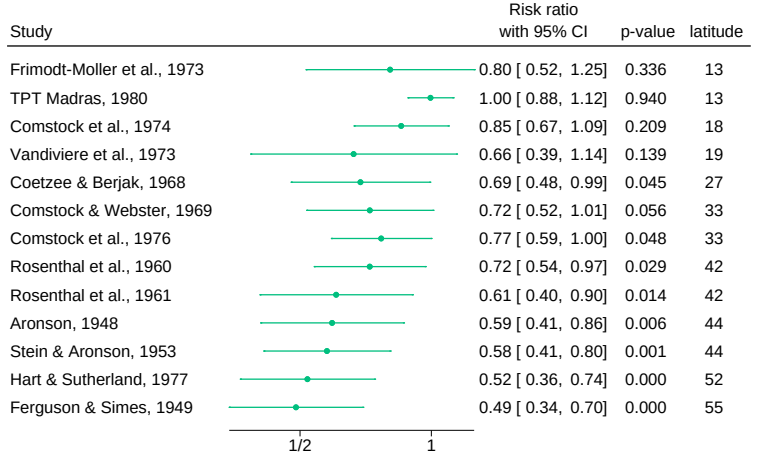
◁

▷ Example 3: Cumulative forest plot

Continuing with example 1, we now perform a cumulative meta-analysis in the ascending order of variable `latitude`. You can specify suboption `descending` within the `cumulative()` option to request a descending order.

We also specify `rr`, which is a synonym of the `eform` option we used in example 2, to display the RRs instead of the default log risk-ratios.

```
. meta forestplot, cumulative(latitude) rr
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
        Study label: studylbl
```

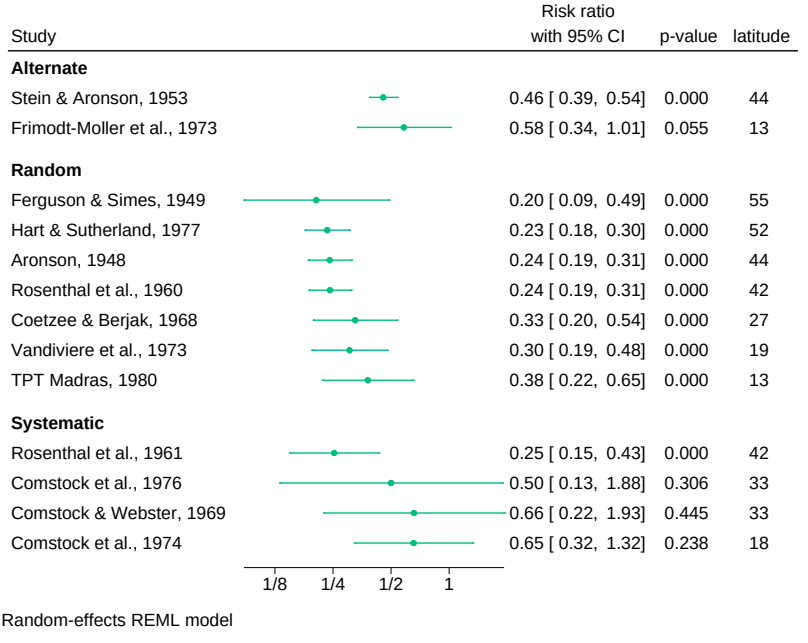


By default, the cumulative meta-analysis forest plot displays the study labels (`_id`), the plot of effect sizes and their CIs (`_plot`), the values of effect sizes and their CIs (`_esci`), the p-values (`_pvalue`) of the corresponding significance tests of the effect sizes, and the values of the order variable (`_order`).

The displayed effect sizes correspond to cumulative overall effect sizes or the overall effect sizes computed for each set of accumulated studies. To distinguish them from study-specific effect sizes, we plot them as unweighted circles using the same color, green, as the overall effect size in a standard meta-analysis forest plot. You can change the default style and color of the markers by specifying the `omarkeropts()` option. The corresponding CIs of the cumulative effect sizes are plotted as CI lines.

We may construct a cumulative forest plot stratified by a covariate by specifying `by()` within `cumulative()`. For example, let's stratify our cumulative analysis by the method of treatment allocation recorded in variable `alloc`.

```
. meta forestplot, cumulative(latitude, by(alloc) descending) rr
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
        Study label: studylbl
```

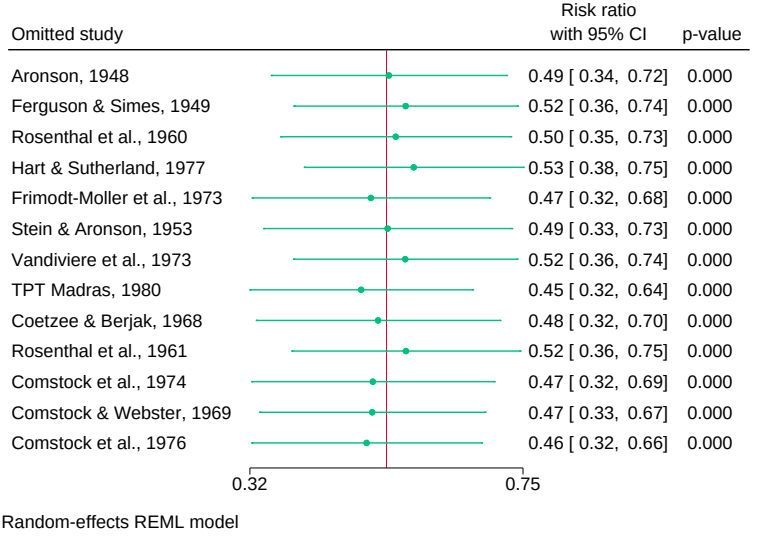


We specified that the analysis be conducted in the descending order of the `latitude` variable. The stratified forest plot shows the same columns as before but the cumulative analysis is performed separately for each group of `alloc`. A consistent pattern is observed across all three groups—RRs tend to increase as latitude decreases.

▷ Example 4: Leave-one-out forest plot

Continuing with example 1, we now perform a leave-one-out meta-analysis by specifying the `leaveoneout` option.

```
. meta forestplot, leaveoneout rr
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
        Study label: studylbl
```



By default, the leave-one-out meta-analysis forest plot displays the study labels (`_id`), the plot of effect sizes and their CIs (`_plot`), the values of effect sizes and their CIs (`_esci`), and the p-values (`_pvalue`) of the corresponding significance tests of the effect sizes.

For each study, the displayed effect size corresponds to an overall effect size computed from a meta-analysis excluding that study. Similarly to the case with cumulative forest plots, we will distinguish the overall effect sizes from study-specific effect sizes by plotting them as unweighted circles using the same color, green, as the overall effect size in a standard meta-analysis forest plot. You can change the default style and color of the markers by specifying the `omarkeropts()` option. The corresponding CIs of the overall effect sizes are plotted as CI lines.

By default, the leave-one-out forest plot displays a vertical line at the overall effect size based on the complete set of studies (with no omission) to facilitate the detection of influential studies. You may suppress this line by specifying option `noesrefline`.

All the overall effect sizes from the leave-one-out meta-analysis are close to the overall effect-size vertical line, and their CI lines intersect with the vertical red line based on all the studies, which means that there are no studies that substantially influence the results of our meta-analysis.

◁

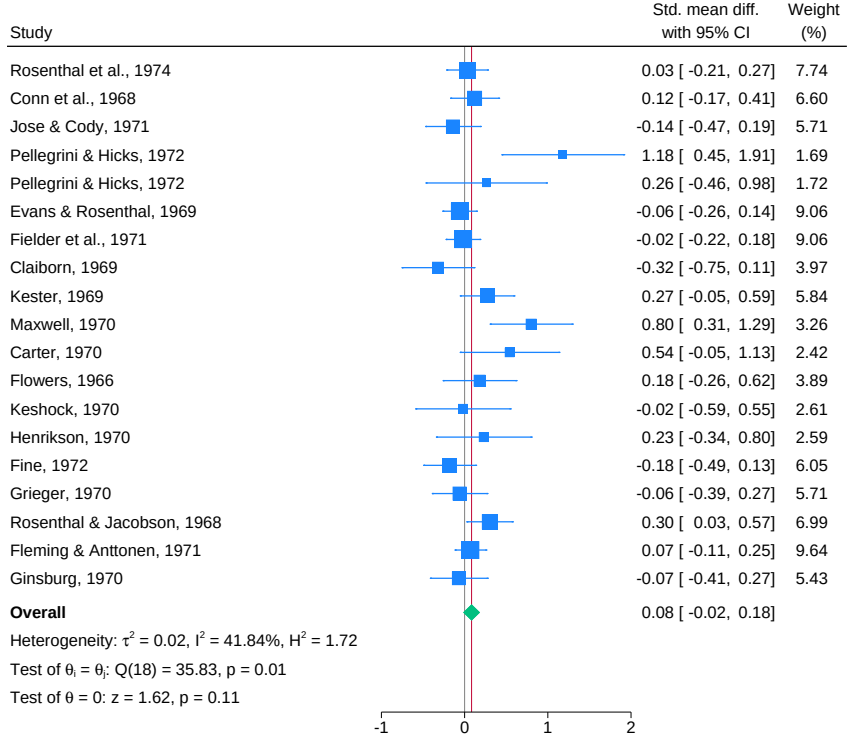
Example 5: Forest plot for precomputed effect sizes

Recall the pupil IQ data (Raudenbush and Bryk 1985; Raudenbush 1984) described in *Effects of teacher expectancy on pupil IQ (pupiliq.dta)* of [META] **meta**. Here we will use its declared version (declared with `meta set`) to construct a forest plot of precomputed effect sizes.

```
. use https://www.stata-press.com/data/r18/pupiliqset, clear
(Effects of teacher expectancy on pupil IQ; set with -meta set-)
```

We specify the `nullrefline` option to show the no-effect line at 0. Effect sizes with corresponding CIs that cross this line are not statistically significant at the 5% level. We also specify the `esrefline` option to draw a vertical line at the overall effect-size value. The default look of both lines may be modified by specifying options `nullrefline(`*line_options*`)` and `esrefline(`*line_options*`)`. See [G-3] ***line_options***.

```
. meta forestplot, nullrefline esrefline
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
        Study label: studylbl
```

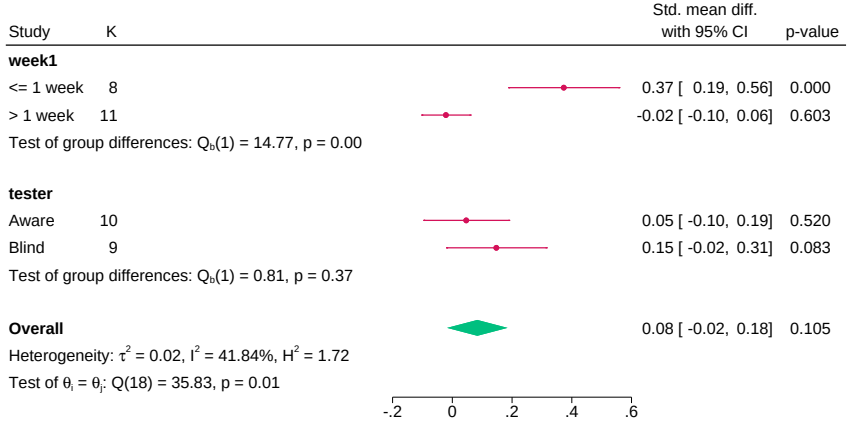


When `meta` data are declared by using `meta set` (that is, when we are working with precomputed effect sizes), the `_data` column is not available. If desired, you may plot the values of effect sizes and their standard errors by specifying the `_esse` column. Other components of the graph are interpreted as in example 1.

▷ Example 6: Multiple subgroup-analyses forest plot

Continuing with example 5, we will conduct multiple subgroup analyses and summarize their results on a forest plot. We specify variables `tester` and `week1` in `subgroup()` as follows:

```
. meta forestplot, subgroup(week1 tester)
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
        Study label: studylbl
```



By default, the forest plot displays the study labels (`_id`), the number of studies within each group (`_K`), the plot of effect sizes and their CIs (`_plot`), the values of effect sizes and their CIs (`_esci`), and the p-values (`_pvalue`) of the corresponding significance tests.

To keep the output compact, the forest plot does not report individual studies, only the number of studies in each group. The between-group homogeneity test based on the Q_b is reported for each subgroup analysis. For example, for subgroup analysis based on variable `week1`, there are two groups, `<= 1 week` and `> 1 week`. The test investigates whether the overall effect sizes corresponding to these two groups are the same. The results of this test are identical to those we would have obtained if we had specified `subgroup(week1)`. You may specify option `nogbhomtests` to suppress these tests. Alternatively, you may modify the default text for the between-group homogeneity tests using option `gbhomtest#text()` (# can be equal to 1 or 2 in this example); see example 16.

Just like with cumulative meta-analysis in example 3, `meta forestplot` uses unweighted circles and CI lines to display the overall group-specific effect sizes and their CIs. But here the circles are displayed in red—the same color used to display the group-specific diamonds in a single-variable subgroup analysis (see example 2).

◁

▷ Example 7: Modifying columns' order and cropping confidence intervals

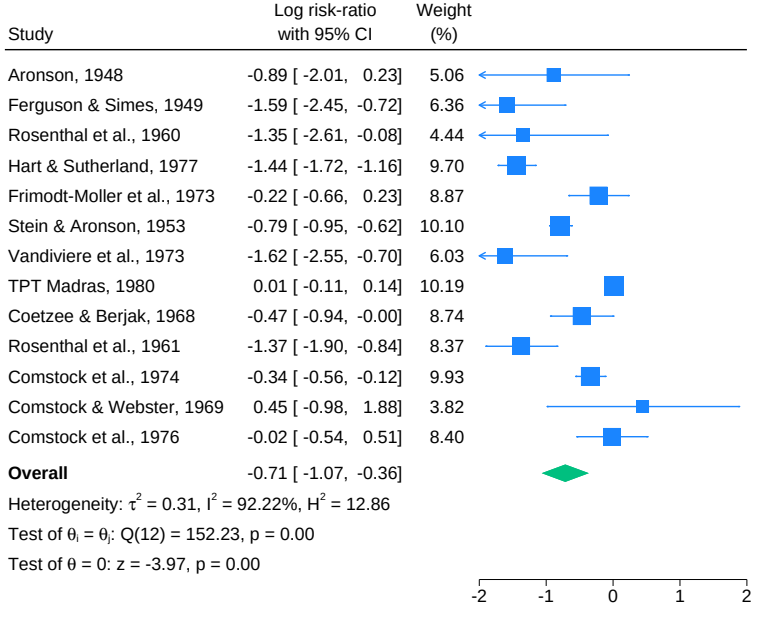
For this and the following examples, let's return to our BCG dataset from example 1.

```
. use https://www.stata-press.com/data/r18/bcgset, clear
(Efficacy of BCG vaccine against tuberculosis; set with -meta esize-)
. meta update, nometashow
```

We used `meta update` to suppress the meta setting information displayed by `meta forest` for the rest of our meta-analysis.

We can choose which columns to display and the order in which to display them in the forest plot by specifying the corresponding column names in the desired order. In the code below, we display the study labels first, followed by the effect sizes and their CIs, then weights, and finally the plot. We also use the `crop(-2 .)` option to restrict the range of the CIs at a lower limit of −2.

```
. meta forestplot _id _esci _weight _plot, crop(-2 .)
```



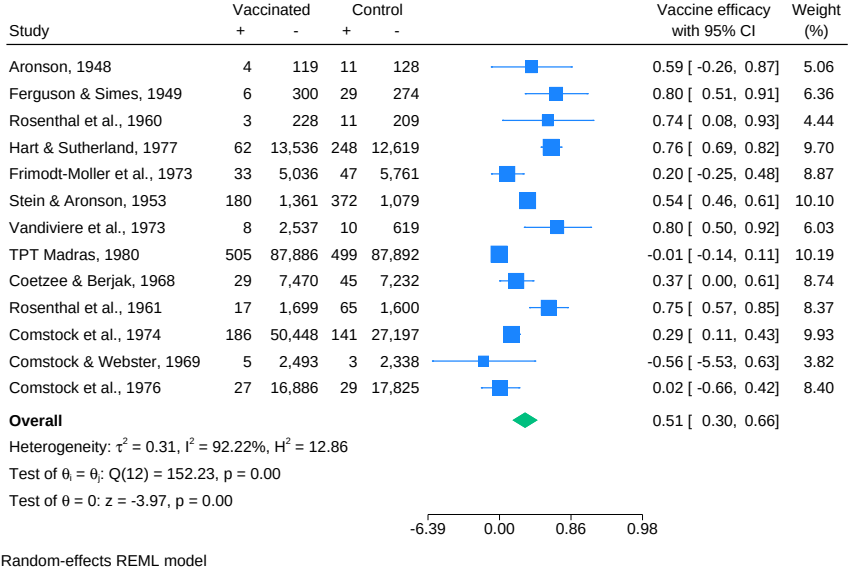
CIs that extend beyond the lower limit of −2 are identified with an arrow head at the cropped endpoint. ◁

Example 8: Applying transformations and changing titles and supertitles

Continuing with the BCG dataset from example 7, we demonstrate how to override the default title and supertitle for the _data column. The summary data we have correspond to a 2×2 table with cells _a, _b, _c, and _d. Cells _a and _b may be referred to as _data1, and cells _c and _d may be referred to as _data2.

We override the supertitle for the _data1 column to display "Vaccinated" and the titles for each cell to display either a "+" or a "−" as follows. We also use the transform() option to report vaccine efficacies instead of log risk-ratios. Vaccine efficacy is defined as 1 − RR and may be requested by specifying transform(efficacy).

```
. meta forestplot, transform(Vaccine efficacy: efficacy)
> columnopts(_data1, supertitle(Vaccinated))
> columnopts(_a _c, title(+)) columnopts(_b _d, title(-))
```

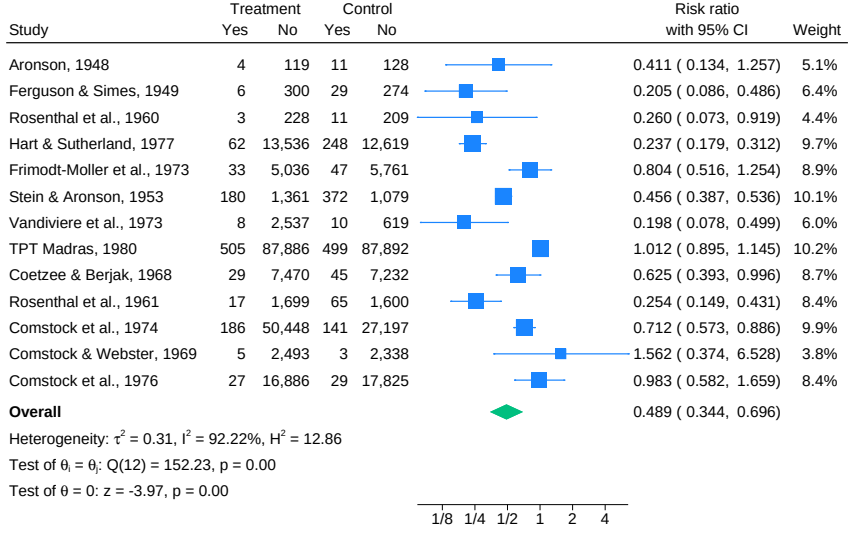


By specifying transform(Vaccine efficacy: efficacy), we also provided a meaningful label, "Vaccine efficacy", to be used for the transformed effect sizes. The overall vaccine efficacy is 0.51, which can be interpreted as a reduction of 51% in the risk of having tuberculosis among the vaccinated group.

Example 9: Changing columns' formatting

Following example 8, we now demonstrate how to override the default formats for the _esci and _weight columns. For the _esci column, we also specify that the CIs be displayed inside parentheses instead of the default brackets. For the _weight column, we specify that the plotted weights be adorned with a % sign and modify the title and supertitle accordingly.

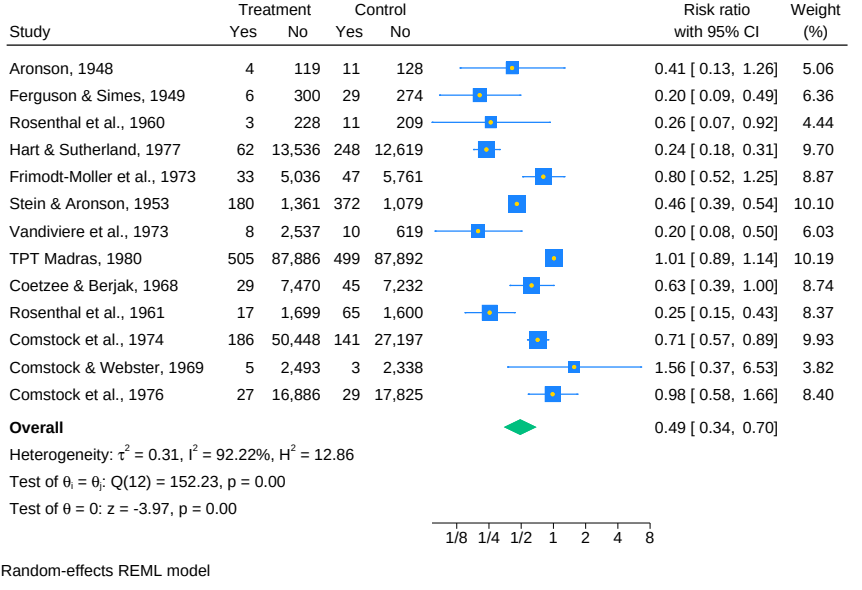
```
. meta forestplot, eform cibind(parentheses)
> columnopts(_esci, format(%6.3f))
> columnopts(_weight, mask(%6.1f%%) title(Weight) supertitle(""))
```



Example 10: Changing axis range and adding center study markers

In this example, we specify the `xscale(range(.125 8))` and `xlabel(#7)` options to specify that the x-axis range be symmetric (on the risk-ratio scale) about the no-effect value of 1 and that 7 tick marks be shown on the axis.

```
. meta forest, eform xscale(range(.125 8)) xlabel(#7) insidemarker
```

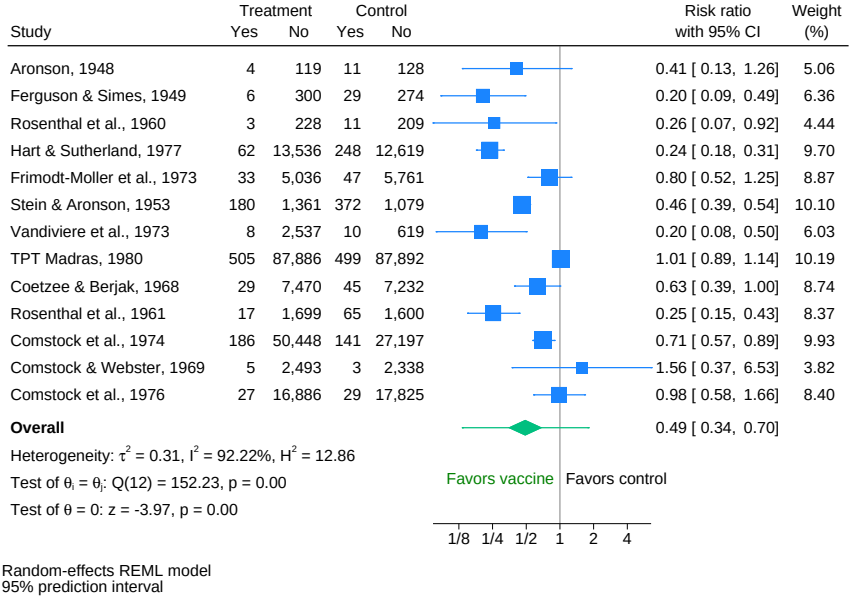


We also used the `insidemarker` option to insert a marker (yellow circle) at the center of the study markers (blue squares) to indicate the study-specific effect sizes. The default attributes of the inserted markers may be modified by specifying `insidemarker(`*marker_options*`)`; see [G-3] ***marker_options***.

Example 11: Prediction intervals and sides favoring control or treatment

Below, we specify the `favorsleft()` and `favorsright()` suboptions of the `nullrefline()` option to annotate the sides of the plot (with respect to the no-effect line) favoring the treatment or control. We will also specify the `predinterval` option to draw the prediction interval for the overall effect size.

```
. meta forest, eform predinterval
> nullrefline(
> favorsleft("Favors vaccine", color(green))
> favorsright("Favors control")
> )
```



In our example, the effect sizes that are falling on the "Favors vaccine" side (left side) reported that the treatment (vaccine) reduced the risk of tuberculosis. The default placement of the labels may be modified using the Graph Editor; see [G-1] **Graph Editor**.

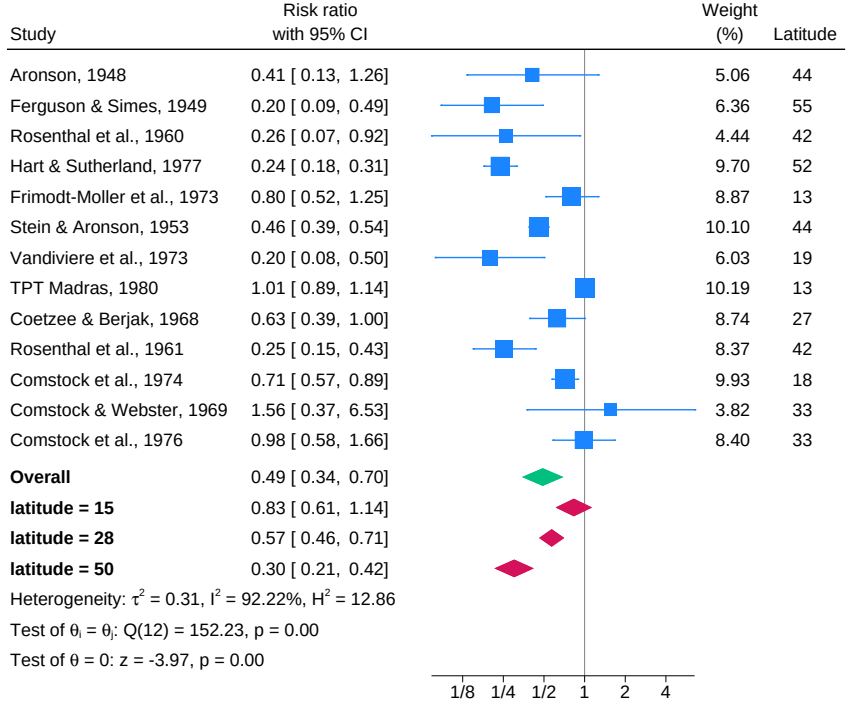
The prediction interval, represented by the green whiskers extending from the overall diamond, provides a plausible range for the effect size in a future, new study.

Example 12: Adding custom columns and overall effect sizes

Consider example 2 of [META] **meta regress postestimation**. We will use the results of the `margins` command from that example to display overall effect sizes at the specified latitudes on the forest plot. This may be done by specifying multiple `customoverall()` options, as we show below.

We also add the `latitude` variable to the forest plot (as the last column) to show study effect sizes as a function of that variable. And we swap the `_esci` and `_plot` columns compared with the default forest plot.

```
. local col mcolor("stred")
. meta forest _id _esci _plot _weight latitude, nullrefline
> columnopts(latitude, title("Latitude"))
> customoverall(-.184 -.495  .127, label("{bf:latitude = 15}") `col')
> customoverall(-.562 -.776 -.348, label("{bf:latitude = 28}") `col')
> customoverall(-1.20 -1.54 -.867, label("{bf:latitude = 50}") `col')
> rr
```

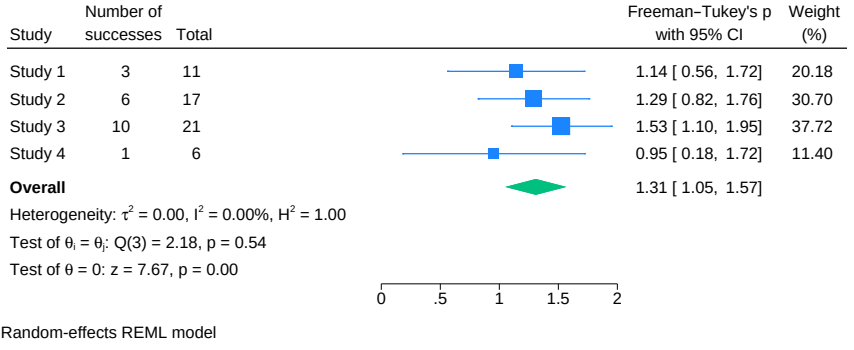


The latitude-specific overall effect sizes from the meta-regression model are shown as red diamonds (`stred` is the red associated with the `stcolor` scheme). In the `customoverall()` options, we specified the values of log risk-ratios, effect sizes in the estimation metric. But because we used the `rr` option, `meta forestplot` displayed the overall diamonds as risk ratios. For example, the mean risk ratio for studies conducted at `latitude` $= 50$ is roughly 0.30 with a CI of $[0.2,\ 0.4]$.

▷ Example 13: Forest plot for meta-analysis of a single proportion

Continuing from the `meta esize ndeaths pensize` setting in example 4 of [META] **meta data**, we will construct a forest plot to summarize our meta-analysis.

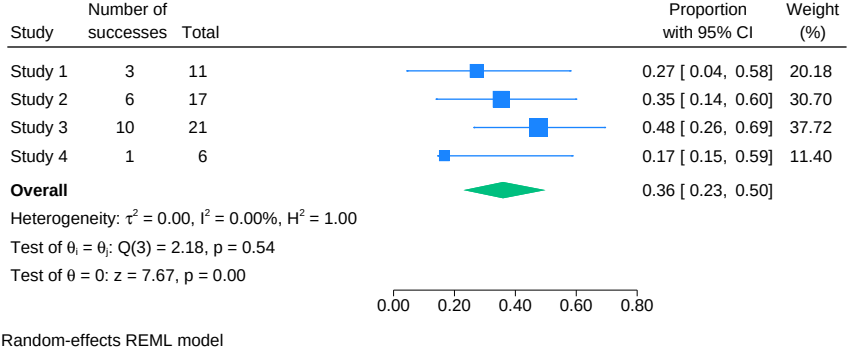
```
. meta forestplot
```



By default, the data displayed on the forest plot for pooling proportions are very similar to those displayed on a forest plot for two-sample binary data; see example 1. The only difference here is the summary data columns. Here `_data` corresponds to the number of events/successes (column `_e`, labeled as `Number of successes` on the forest plot) and the study sample size (column `_n`, labeled as `Total`). The displayed effect sizes are Freeman–Tukey-transformed proportions.

Below, we report our results as proportions using the `proportion` option. When the effect sizes are the Freeman–Tukey-transformed proportions, this option is equivalent to specifying option `transform(invftukey, hmean)`.

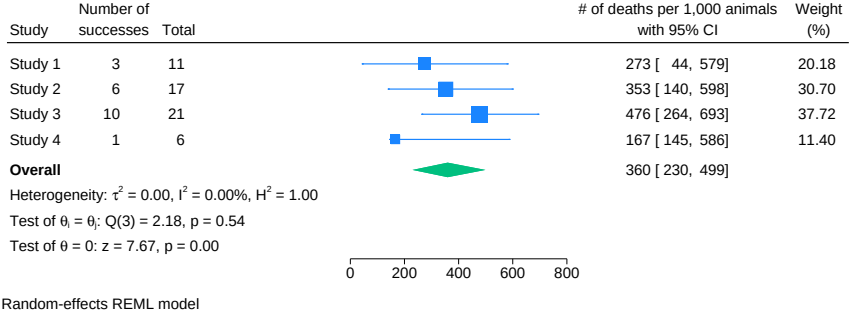
```
. meta forestplot, proportion
```



One unique characteristic of forest plots based on Freeman–Tukey-transformed proportions is that when you back-transform the effect sizes and their CIs (to report proportions), the back-transformed CIs are no longer symmetric. This is different from two-sample binary data with log odds-ratios or log risk-ratios as effect sizes. When you back-transform (exponentiate) these effect sizes to report odds ratios or risk ratios, the axis labels are also exponentiated to maintain the graphical representation of the CIs as symmetric. This is not possible for the one-sample case when the effect size is `ftukeyprop` because the back-transformation (the inverse Freeman–Tukey function) depends on sample size n; see *Inverse Freeman–Tukey transformation* in *Methods and formulas* in [META] **meta summarize**. The sample size varies between the studies, making it impossible to apply one transformation to the axis labels to make all study CIs symmetric. Therefore, if option `proportion` or `transform(invftukey)` is specified, no transformation is applied to the x-axis labels, and the plotted CIs for proportions will no longer be symmetric.

Next, we will use the `scale(1000)` suboption of `transform(invftukey)` to report our results as the number of deaths per 1,000 animals. We will also report the effect sizes and their CIs as integers using the `format(%3.0f)` suboption of `columnopts(_esci)`.

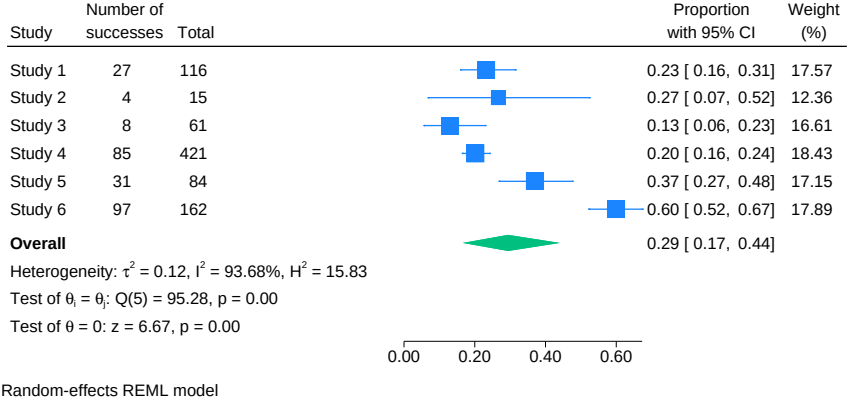
```
. meta forestplot,
> transform("# of deaths per 1,000 animals": invftukey, scale(1000))
> xlabel(, format(%3.0f)) columnopts(_esci, format(%3.0f))
```



Example 14: Increasing plot-region margin

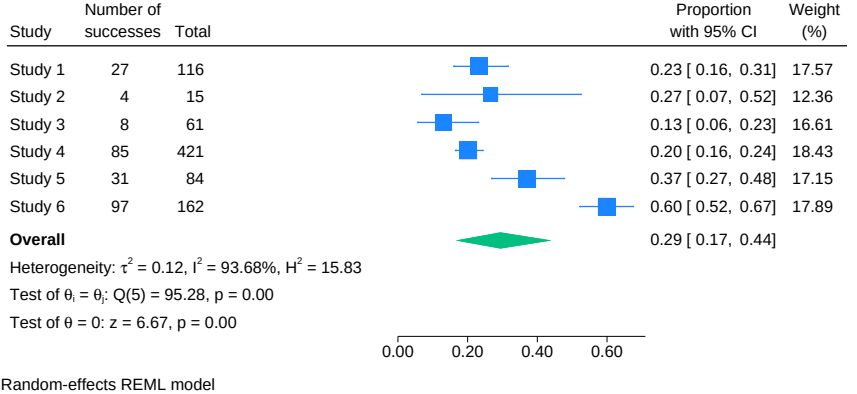
Continuing with example 8 of [META] **meta esize**, we will construct a forest plot to summarize our meta-analysis. It is quite common with forest plots of proportions for some study CIs in the `_plot` column to be very close to the `_esci` column (see `Study 6` in our example).

```
. meta forestplot, proportion
```



Below, we increase the margin between the plot region of column `_plot` and that of column `_esci` using the `columnopts(_plot, plotregion(margin(right)))` option.

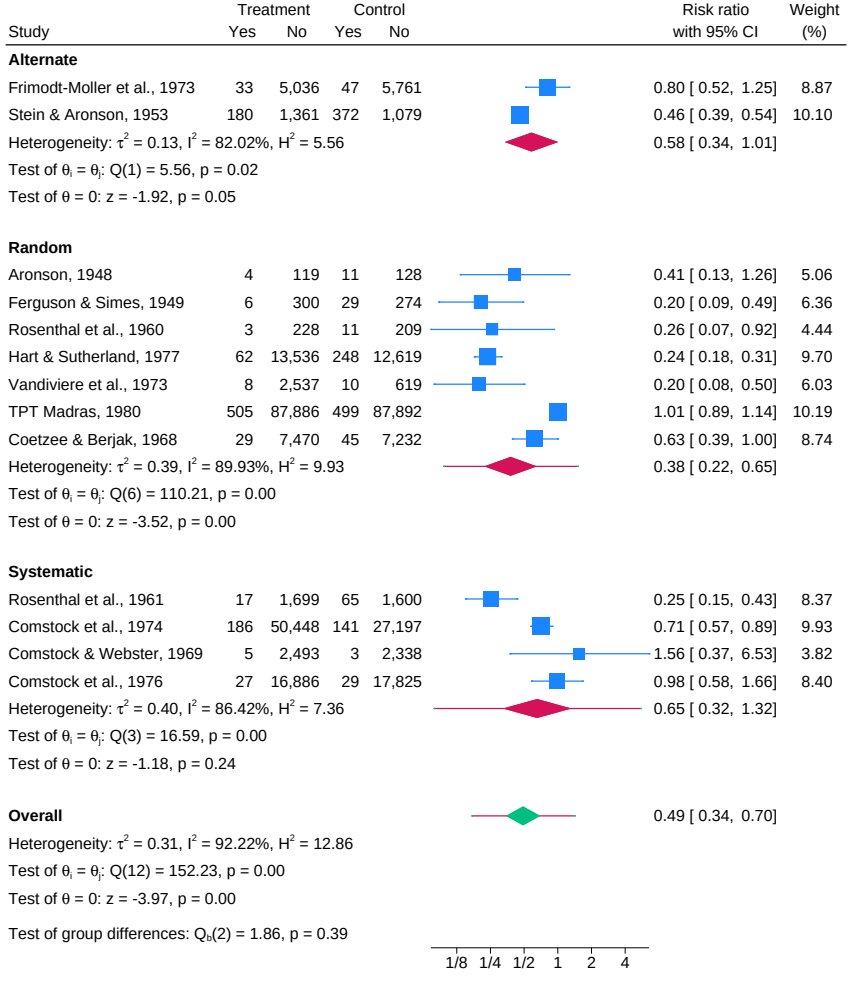
```
. meta forestplot, proportion columnopts(_plot, plotregion(margin(right)))
```



Example 15: Prediction intervals with subgroup analysis and eliminating space in the _esci column

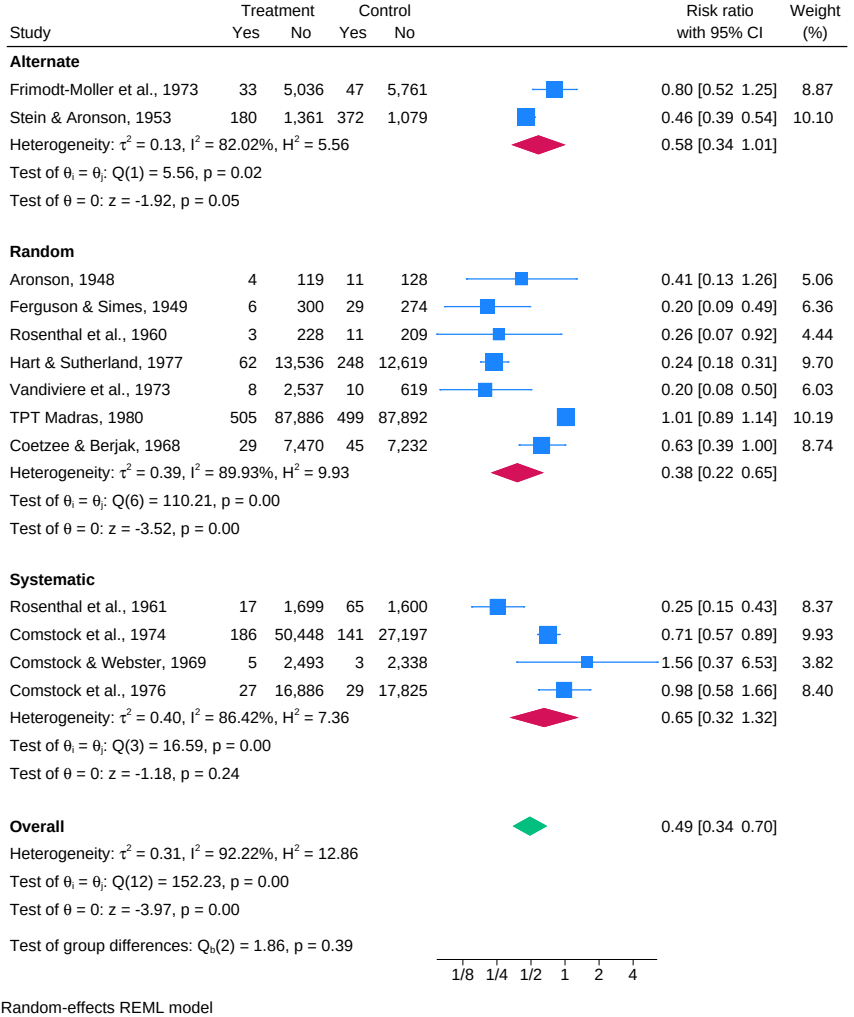
Continuing with example 2, we will add a 90% prediction interval within each subgroup. Notice that a predication interval is defined only when there are at least three studies; therefore, it is not computable for the first subgroup (`Alternate`).

```
. meta forest, subgroup(alloc) rr predinterval(90, lcolor(stred))
```



Next, we will eliminate the space in the _esci column right after the left bracket of the effect-size CI. This is done by removing the default binding of the CIs using option cibind(none) and specifying our own custom binding for columns _lb and _ub as follows:

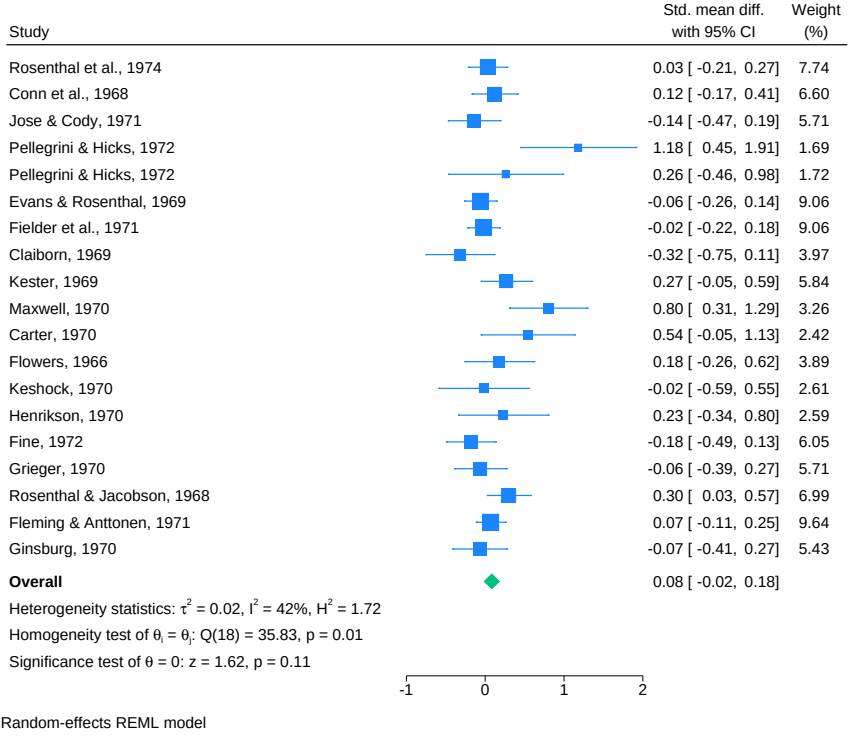
```
. meta forest, subgroup(alloc) rr
> columnopts(_lb, mask("[%4.2f"))
> columnopts(_ub, mask("%4.2f]")) cibind(none)
```



Example 16: Modifying default text for heterogeneity statistics and statistical tests

Continuing with example 5, we will modify the default text reported in the three lines under `Overall` using options `ohetstatstext()` (for the first line), `ohomtesttext()` (for the second line), and `osigtesttext()` (for the third line). We will be slightly more descriptive about the type of information reported in each line and report the I^2 statistic without decimal points.

```
. use https://www.stata-press.com/data/r18/pupiliqset, clear
(Effects of teacher expectancy on pupil IQ; set with -meta set-)
. local hstats "Heterogeneity statistics:"
. local htest "Homogeneity test of {&theta}{sub:i} = {&theta}{sub:j}:"
. local stest "Significance test of {&theta} = 0:"
. meta forest,
> ohetstatstext("`hstats' {&tau}{sup:2} = 0.02, I{sup:2} = 42%, H{sup:2} = 1.72")
> ohomtesttext("`htest' Q(18) = 35.83, p = 0.01")
> osigtesttext("`stest' z = 1.62, p = 0.11")
```

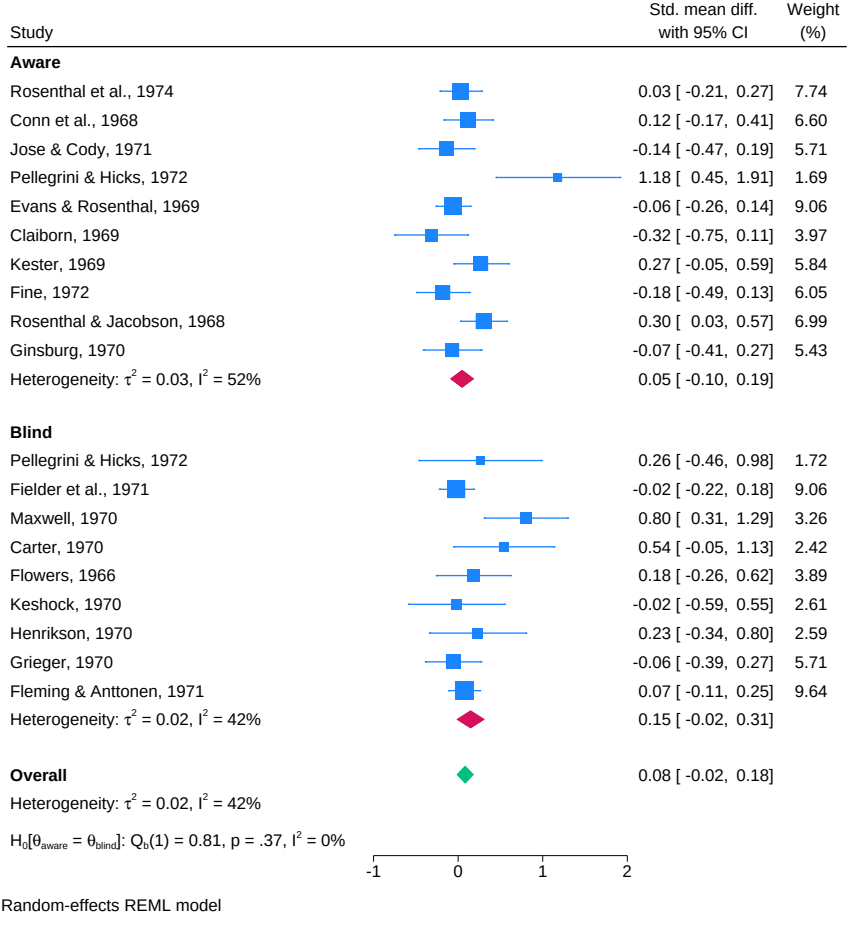


Next, we will construct a subgroup forest plot based on variable `tester` (`aware` versus `blind`). See example 2 for a detailed description of the subgroup forest plot.

We will suppress the within-group homogeneity tests (option `nogwhomtests`) and the tests of significance for the group-specific overall effect sizes (option `nogsigtests`). We also use options `noohomtest` and `noosigtest` to suppress the same information for the overall analysis. We will report only τ^2 and I^2 in the overall heterogeneity statistics (option `ohetstatstext()`) and in the group-specific heterogeneity statistics (by repeating the `ghetstats#text()` option for each subgroup). Finally, we use option `gbhomtest1text()` to modify the description of the between-group homogeneity test and label it as

$H_0: \theta_{\text{aware}} = \theta_{\text{blind}}$ and report the I^2 statistic corresponding to the Q_b test statistic. The I^2 statistic is computed as follows: $I^2 = 100 \times \max\{0, 1 - (L-1)/Q_b\}$, where L is the number of subgroups ($L = 2$ in this example).

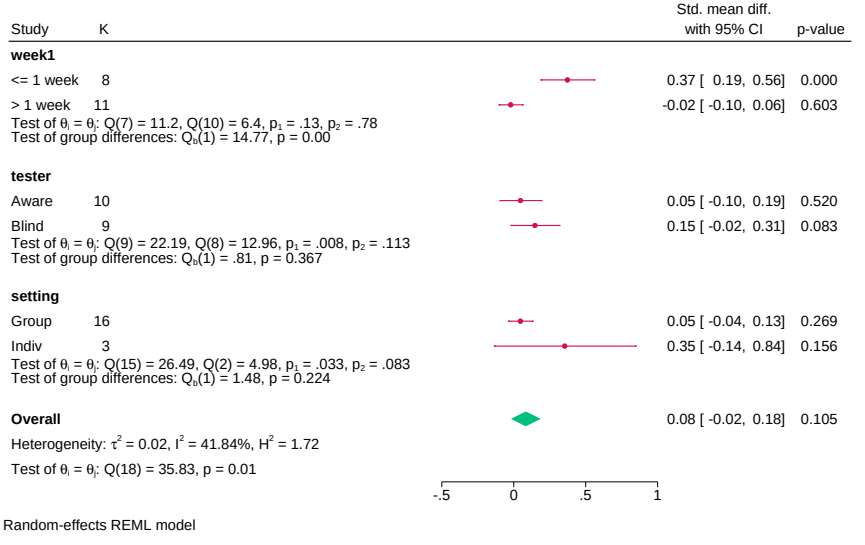
```
. local H0txt "H{sub:0}[{&theta}{sub:aware} = {&theta}{sub:blind}]:"
. local H0stats "Q{sub:b}(1) = 0.81, p = .37, I{sup:2} = 0% "
. meta forest, subgroup(tester) nogsigtests noosigtest nogwhomtests noohomtest
> ghetstats1text("Heterogeneity: {&tau}{sup:2} = 0.03, I{sup:2} = 52%")
> ghetstats2text("Heterogeneity: {&tau}{sup:2} = 0.02, I{sup:2} = 42%")
> ohetstatstext("Heterogeneity: {&tau}{sup:2} = 0.02, I{sup:2} = 42%")
> gbhomtest1text("`H0txt' `H0stats'")
```



Finally, we will construct a multiple subgroup-analyses forest plot based on variables `week1`, `tester`, and `setting`. See example 6 for the interpretation of this type of forest plot. By default, only information regarding the between-group homogeneity tests is reported for each variable. We will use option `gbhomtest#text()` (corresponding to the #th variable in `subgroup()`) to display the same default information regarding the between-group homogeneity test, but we now add an additional line reporting the within-group homogeneity tests for the groups defined by each variable. This is done by specifying

two strings within the gbhomtest#text() option, one for each line. The within-group homogeneity test information may be obtained from the second table in the output of meta summarize, subgroup(week1 tester setting).

```
. local Qdesc "Test of {&theta}{sub:i} = {&theta}{sub:j}:"
. local Qbdesc "Test of group differences: Q{sub:b}(1) ="
. meta forest, subgroup(week1 tester setting)
> gbhomtest1text( "‘Qdesc’ Q(7) = 11.2, Q(10) = 6.4, p{sub:1} = .13, p{sub:2} = .78"
>      "‘Qbdesc’ 14.77, p = 0.00")
> gbhomtest2text("‘Qdesc’ Q(9) = 22.19, Q(8) = 12.96, p{sub:1} = .008, p{sub:2} = .113"
>      "‘Qbdesc’ .81, p = 0.367")
> gbhomtest3text("‘Qdesc’ Q(15) = 26.49, Q(2) = 4.98, p{sub:1} = .033, p{sub:2} = .083"
>      "‘Qbdesc’ 1.48, p = 0.224")
```



Methods and formulas

Methods and formulas for the statistics reported by meta forestplot are given in *Methods and formulas* of [META] **meta summarize**.

References

Anzures-Cabrera, J., and J. P. T. Higgins. 2010. Graphical displays for meta-analysis: An overview with suggestions for practice. *Research Synthesis Methods* 1: 66–80. https://doi.org/10.1002/jrsm.6.

Colditz, G. A., T. F. Brewer, C. S. Berkey, M. E. Wilson, E. Burdick, H. V. Fineberg, and F. Mosteller. 1994. Efficacy of BCG vaccine in the prevention of tuberculosis: Meta-analysis of the published literature. *Journal of the American Medical Association* 271: 698–702. https://doi.org/10.1001/jama.1994.03510330076038.

Fisher, D. J. 2016. “Two-stage individual participant data meta-analysis and generalized forest plots”. In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 280–307. 2nd ed. College Station, TX: Stata Press.

Harris, R. J., M. J. Bradburn, J. J. Deeks, R. M. Harbord, D. G. Altman, and J. A. C. Sterne. 2016. “metan: Fixed- and random-effects meta-analysis”. In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 29–54. 2nd ed. College Station, TX: Stata Press.

Lewis, J. A., and S. H. Ellis. 1982. A statistical appraisal of post-infarction beta-blocker trials. *Primary Cardiology* Suppl. 1: 31–37.

Lewis, S., and M. Clarke. 2001. Forest plots: Trying to see the wood and the trees. *BMJ* 322: 1479–1480. https://doi.org/10.1136/bmj.322.7300.1479.

Raudenbush, S. W. 1984. Magnitude of teacher expectancy effects on pupil IQ as a function of the credibility of expectancy induction: A synthesis of findings from 18 experiments. *Journal of Educational Psychology* 76: 85–97. https://doi.org/10.1037/0022-0663.76.1.85.

Raudenbush, S. W., and A. S. Bryk. 1985. Empirical Bayes meta-analysis. *Journal of Educational Statistics* 10: 75–98. https://doi.org/10.2307/1164836.

Schriger, D. L., D. G. Altman, J. A. Vetter, T. Heafner, and D. Moher. 2010. Forest plots in reports of systematic reviews: A cross-sectional study reviewing current practice. *International Journal of Epidemiology* 39: 421–429. https://doi.org/10.1093/ije/dyp370.

Also see

[META] **meta data** — Declare meta-analysis data

[META] **meta galbraithplot** — Galbraith plots

[META] **meta labbeplot** — L'Abbé plots

[META] **meta summarize** — Summarize meta-analysis data+

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta summarize — Summarize meta-analysis data[+]

[+]This command includes features that are part of StataNow.

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
References | Also see

Description

`meta summarize` summarizes `meta` data. It reports individual effect sizes and the overall effect size (ES), their confidence intervals (CIs), heterogeneity statistics, and more. `meta summarize` can perform random-effects (RE), common-effect (CE), and fixed-effects (FE) meta-analyses. It can also perform subgroup, cumulative, and sensitivity meta-analyses. For graphical display of meta-analysis summaries, see [META] **meta forestplot**.

Quick start

Perform meta-analysis and summarize `meta` data, which were declared by either `meta set` or `meta esize`

```
meta summarize
```

Same as above, but summarize meta-analysis results using the empirical Bayes RE method instead of the declared method

```
meta summarize, random(ebayes)
```

Same as above, but report transformed effect sizes and CIs using the hyperbolic tangent function

```
meta summarize, random(ebayes) transform(tanh)
```

Perform subgroup meta-analysis based on the categorical variable `x1`

```
meta summarize, subgroup(x1)
```

Perform subgroup analysis based on the categorical variables `x1`, `x2`, and `x3`

```
meta summarize, subgroup(x1 x2 x3)
```

Perform cumulative meta-analysis (CMA), where studies are included in the CMA based on the ascending order of observations in variable `x4`

```
meta summarize, cumulative(x4)
```

Same as above, but stratify the results of the CMA based on groups of the categorical variable `x5`

```
meta summarize, cumulative(x4, by(x5))
```

Perform leave-one-out meta-analysis

```
meta summarize, leaveoneout
```

Perform sensitivity meta-analysis by assuming a fixed value of 0.2 for the between-study heterogeneity parameter τ^2, assuming that the declared model is RE

```
meta summarize, tau2(.2)
```

Menu

Statistics > Meta-analysis

Syntax

Meta-analysis as declared with meta set or meta esize

```
meta summarize [if] [in] [, options reopts]
```

Random-effects meta-analysis

```
meta summarize [if] [in], random[(remethod)] [options reopts]
```

Common-effect meta-analysis

```
meta summarize [if] [in], common[(cefemethod)] [options]
```

Fixed-effects meta-analysis

```
meta summarize [if] [in], fixed[(cefemethod)] [options]
```

options	Description
Main	
subgroup(*varlist*)	subgroup meta-analysis for each variable in *varlist*
cumulative(*cumulspec*)	cumulative meta-analysis
leaveoneout	leave-one-out meta-analysis
Options	
level(#)	set confidence level; default is as declared for meta-analysis
citype(*citype*)	specify the type of study CI (for meta-analysis of a single proportion)
proportion	report proportions (for meta-analysis of a single proportion)
prevalence	synonym for proportion but labels the effect sizes as Prevalence in the output
+correlation	report correlations (for meta-analysis of correlations)
eform_option	report exponentiated results
transform(*transfspec*)	report transformed results
sort(*varlist*[, ...])	sort studies according to *varlist*
tdistribution	report t test instead of z test for the overall effect size
nostudies	suppress output for individual studies
noheader	suppress output header
[no]metashow	display or suppress meta settings in the output
display_options	control column formats
Maximization	
maximize_options	control the maximization process; seldom used

+These features are part of StataNow.

collect is allowed; see **[U] 11.1.10 Prefix commands**.

remethod	Description
reml	restricted maximum likelihood; the default
mle	maximum likelihood
ebayes	empirical Bayes
dlaird	DerSimonian–Laird
sjonkman	Sidik–Jonkman
hedges	Hedges
hschmidt	Hunter–Schmidt

cefemethod	Description
mhaenszel	Mantel–Haenszel
invvariance	inverse variance
ivariance	synonym for invvariance

reopts	Description
tau2(#)	sensitivity meta-analysis using a fixed value of between-study variance τ^2
i2(#)	sensitivity meta-analysis using a fixed value of heterogeneity statistic I^2
predinterval[(#)]	report prediction interval for the overall effect size
se(*seadj*)	adjust standard error of the overall effect size

Options

Main

Options random(), common(), and fixed(), when specified with meta summarize, temporarily override the global model declared by meta set or meta esize during the computation. Options random(), common(), and fixed() may not be combined. If these options are omitted, the declared meta-analysis model is assumed; see *Declaring a meta-analysis model* in [META] **meta data**. Also see *Meta-analysis models* in [META] **Intro**.

random and random(*remethod*) specify that a random-effects model be assumed for meta-analysis; see *Random-effects model* in [META] **Intro**.

remethod specifies the type of estimator for the between-study variance τ^2. *remethod* is one of reml, mle, ebayes, dlaird, sjonkman, hedges, or hschmidt. random is a synonym for random(reml). See *Options* in [META] **meta esize** for more information.

common and common(*cefemethod*) specify that a common-effect model be assumed for meta-analysis; see *Common-effect ("fixed-effect") model* in [META] **Intro**. Also see the discussion in [META] **meta data** about common-effect versus fixed-effects models.

common implies common(mhaenszel) for effect sizes lnoratio, lnrratio, and rdiff and common(invvariance) for all other effect sizes. common(mhaenszel) is supported only with effect sizes lnoratio, lnrratio, and rdiff.

cefemethod is one of mhaenszel or invvariance (synonym ivariance). See *Options* in [META] **meta esize** for more information.

`fixed` and `fixed(`*cefemethod*`)` specify that a fixed-effects model be assumed for meta-analysis; see *Fixed-effects model* in [META] **Intro**. Also see the discussion in [META] **meta data** about fixed-effects versus common-effect models.

`fixed` implies `fixed(mhaenszel)` for effect sizes `lnoratio`, `lnrratio`, and `rdiff` and `fixed(invvariance)` for all other effect sizes. `fixed(mhaenszel)` is supported only with effect sizes `lnoratio`, `lnrratio`, and `rdiff`.

cefemethod is one of `mhaenszel` or `invvariance` (synonym `ivariance`); see *Options* in [META] **meta esize** for more information.

`subgroup(`*varlist*`)` specifies that a subgroup meta-analysis (subgroup analysis) be performed for each variable in *varlist*. Subgroup analysis performs meta-analysis separately for each variable in *varlist* and for each group as defined by that variable. The specified meta-analysis model is assumed for each subgroup. This analysis is useful when the results of all studies are too heterogeneous to be combined into one estimate but the results are similar within certain groups of studies. The specified variables can be numeric or string variables. When multiple variables are specified, only the subgroup results are displayed; that is, the results from individual studies are suppressed for brevity. This option may not be combined with `cumulative()` or `leaveoneout`.

`cumulative(`*ordervar*`[ , ascending|descending by(`*byvar*`) ])` performs a cumulative meta-analysis (CMA). CMA performs multiple meta-analyses and accumulates the results by adding one study at a time to each subsequent analysis. It is useful for monitoring the results of the studies as new studies become available. The studies enter the CMA based on the ordered values of variable *ordervar*. *ordervar* must be a numeric variable. By default, ascending order is assumed unless the suboption `descending` is specified; only one of `ascending` or `descending` is allowed. The `by(`*byvar*`)` option specifies that the CMA be stratified by variable *byvar*. This option may not be combined with `subgroup()` or `leaveoneout`.

`leaveoneout` performs a leave-one-out meta-analysis. For each study, the corresponding leave-one-out meta-analysis is a meta-analysis of all the studies except that study. It is useful for assessing the effect of a single study on the meta-analysis results and for identifying outliers if they exist. This option may not be combined with `subgroup()` or `cumulative()`.

reopts are `tau2(#)`, `i2(#)`, `predinterval[ (#) ]`, and `se(khartung[ , truncated ])`. These options are used with random-effects meta-analysis.

`tau2(#)` specifies the value of the between-study variance parameter, τ^2, to use for the random-effects meta-analysis. This option is useful for exploring the sensitivity of the results to different levels of between-study heterogeneity. Only one of `tau2()` or `i2()` may be specified. This option is not allowed in combination with `subgroup()`, `cumulative()`, or `leaveoneout`.

`i2(#)` specifies the value of the heterogeneity statistic I^2 (as a percentage) to use for the random-effects meta-analysis. This option is useful for exploring the sensitivity of the results to different levels of between-study heterogeneity. Only one of `i2()` or `tau2()` may be specified. This option is not allowed in combination with `subgroup()`, `cumulative()`, or `leaveoneout`.

`predinterval` and `predinterval(#)` specify that the 95% or #% prediction interval be reported for the overall effect size in addition to the confidence interval. # specifies the confidence level of the prediction interval. The prediction interval provides plausible ranges for the effect size in a future, new study. This option is not allowed in combination with `subgroup()` when specified with more than one variable, `cumulative()`, or `leaveoneout`.

`se(`*seadj*`)` specifies that the adjustment *seadj* be applied to the standard error of the overall effect size. Additionally, the test of significance of the overall effect size is based on a Student's t distribution instead of the normal distribution.

seadj is `khartung`[`, truncated`]. Adjustment `khartung` specifies that the Knapp–Hartung adjustment (Hartung and Knapp 2001a, 2001b; Knapp and Hartung 2003), also known as the Sidik–Jonkman adjustment (Sidik and Jonkman 2002), be applied to the standard error of the overall effect size. `hknapp` and `sjonkman` are synonyms for `khartung`. `truncated` specifies that the truncated Knapp–Hartung adjustment (Knapp and Hartung 2003), also known as the modified Knapp–Hartung adjustment, be used.

Options

`level(`#`)` specifies the confidence level, as a percentage, for confidence intervals. The default is as declared for the meta-analysis session; see *Declaring a confidence level for meta-analysis* in [META] **meta data**. Also see option `level()` in [META] **meta set**.

`citype(`*citype*`)` specifies the type of CI to be reported for meta-analysis of a single proportion. *citype* is one of `wald` (the default), `exact`, `wilson`, `agresti`, or `jeffreys`. For more details, see *Binomial proportion* in [R] **ci**. This option affects only individual study CIs and not the CI for the overall effect size. Thus, it may not be combined with options `cumulative()`, `leaveoneout`, and `subgroup()` with more than one variable.

`proportion` reports results as proportions for meta-analysis of a single proportion. By default, the results are displayed in the metric declared with `meta esize`, such as Freeman–Tukey-transformed proportions or logit-transformed proportions. `proportion` is a synonym for `transform(invftukey, hmean)` when the effect size is `esize(ftukeyprop)` or `transform(invlogit)` when the effect size is `esize(logitprop)`. This option affects how results are displayed, not how they are estimated or stored.

`prevalence` is a synonym for `proportion` but labels the effect sizes as `Prevalence` instead of `Proportion` in the output.

`correlation` is part of StataNow. It reports results as correlations for meta-analysis of correlations. By default, the results are displayed in the metric declared with `meta esize`, such as Fisher's z-transformed correlations. `correlation` is a synonym for `transform(corr)` when the effect size is `esize(fisherz)`. This option affects how results are displayed, not how they are estimated or stored.

eform_option is one of `eform`, `eform(`*string*`)`, `or`, or `rr`. It reports exponentiated effect sizes and transforms their respective confidence intervals, whenever applicable. By default, the results are displayed in the metric declared with `meta set` or `meta esize` such as log odds-ratios and log risk-ratios. *eform_option* affects how results are displayed, not how they are estimated and stored. *eform_option* is not available with two-sample continuous data, one-sample binary data, and correlation data.

`eform(`*string*`)` labels the exponentiated effect sizes as *string*; the other options use default labels. The default label is specific to the chosen effect size. For example, option `eform` uses `Odds ratio` when used with log odds-ratios declared with `meta esize` or `Risk ratio` when used with the declared log risk-ratios. Option `or` is a synonym for `eform` when log odds-ratio is declared, and option `rr` is a synonym for `eform` when log risk-ratio is declared. If option `eslabel(`*eslab*`)` is specified during declaration, then `eform` will use the `exp(`*eslab*`)` label or, if *eslab* is too long, the `exp(ES)` label.

`transform(`[*label*`:`] *transf_name*`)` reports transformed effect sizes and CIs. *transf_name* is one of `corr`, `efficacy`, `exp`, `invlogit`, `tanh`, or `invftukey`[`,` *invftopts*]. When *label* is specified, the transformed effect sizes are labeled as *label* instead of using the default label. This option may not be combined with *eform_option*.

`corr` transforms effect sizes (and CIs) specified as Fisher's z values into correlations and, by default, labels them as `Correlation`; that is, `transform(corr)` is a synonym for `transform(Correlation: tanh)`.

`efficacy` transforms the effect sizes and CIs using the $1 - $ `exp()` function (or more precisely, the `-expm1()` function) and labels them as `Efficacy`. This transformation is used, for example, when the effect sizes are log risk-ratios so that the transformed effect sizes can be interpreted as treatment efficacies, $1 -$ risk ratios.

`exp` exponentiates effect sizes and CIs and, by default, labels them as `exp(ES)`. This transformation is used, for example, when the effect sizes are log risk-ratios, log odds-ratios, and log hazard-ratios so that the transformed effect sizes can be interpreted as risk ratios, odds ratios, and hazard ratios. If the declared effect sizes are log odds-ratios or log risk-ratios, the default label is `Odds ratio` or `Risk ratio`, respectively.

`invlogit` transforms the effect sizes and CIs using the inverse-logit function, `invlogit()`, and, by default, labels them as `invlogit(ES)`. This transformation is used, for example, when the effect sizes are logit of proportions so that the transformed effect sizes can be interpreted as proportions.

`tanh` applies the hyperbolic tangent transformation, `tanh()`, to the effect sizes and CIs and, by default, labels them as `tanh(ES)`. This transformation is used, for example, when the effect sizes are Fisher's z values so that the transformed effect sizes can be interpreted as correlations.

`invftukey`[`,` *invftopts*] is relevant to meta-analysis of a single proportion. It applies the inverse Freeman–Tukey double arcsine transformation to the effect sizes and CIs and, by default, labels them as `Proportion`. This transformation is used only when pooling proportions (prevalences) with the default effect size `esize(ftukeyprop)`. See *Inverse Freeman–Tukey transformation* for more details.

invftopts are `hmean`, `gmean`, `amean`, `ivariance`, and `scale()`.

`hmean` specifies that the harmonic mean of the within-study sample sizes be used to back-transform the overall effect size.

`gmean` specifies that the geometric mean of the within-study sample sizes be used to back-transform the overall effect size.

`amean` specifies that the arithmetic mean of the within-study sample sizes be used to back-transform the overall effect size.

`ivariance` specifies that the inverse of the variance of the overall effect size be used to back-transform the overall effect size.

`scale(`#`)` scales the study proportions, the overall proportion, and their CIs by #. This option is relevant when the proportions are very small, in which case it might be preferable to report them as the number of successes per, say, 1,000 or 10,000 observations. # must be an integer greater than 1.

`sort(`*varlist*[`,` `ascending` | `descending`]`)` sorts the studies in ascending or descending order based on values of the variables in *varlist*. This option is useful if you want to sort the studies in the output by effect sizes, `sort(_meta_es)`, or by precision, `sort(_meta_se)`. By default, ascend-

ing order is assumed unless the suboption descending is specified; only one of ascending or descending is allowed. *varlist* may contain string and numeric variables. This option is not allowed with cumulative(). When sort() is not specified, the order of the studies in the output is based on the ascending values of variable _meta_id, which is equivalent to sort(_meta_id).

tdistribution reports a t test instead of a z test for the overall effect size. This option may not be combined with option subgroup(), cumulative(), leaveoneout, or se().

nostudies (synonym nostudy) suppresses the display of information such as effect sizes and their CIs for individual studies from the output table.

noheader suppresses the output header.

metashow and nometashow display or suppress the meta setting information. By default, this information is displayed at the top of the output. You can also specify nometashow with meta update to suppress the meta setting output for the entire meta-analysis session.

display_options: cformat(%*fmt*), pformat(%*fmt*), and sformat(%*fmt*); see [R] **Estimation options**. The defaults are cformat(%9.3f), pformat(%5.3f), and sformat(%8.2f).

wgtformat(%*fmt*) specifies how to format the weight column in the output table. The default is wgtformat(%5.2f). The maximum format width is 5.

ordformat(%*fmt*) specifies the format for the values of the order variable, specified in cumulative(*ordervar*). The default is ordformat(%9.0g). The maximum format width is 9.

Maximization

maximize_options: iterate(#), tolerance(#), nrtolerance(#), nonrtolerance (see [R] **Maximize**), from(#), and showtrace. These options control the iterative estimation of the between-study variance parameter, τ^2, with random-effects methods reml, mle, and ebayes. These options are seldom used.

from(#) specifies the initial value for τ^2 during estimation. By default, the initial value for τ^2 is the noniterative Hedges estimator.

showtrace displays the iteration log that contains the estimated parameter τ^2, its relative difference with the value from the previous iteration, and the scaled gradient.

Remarks and examples

Remarks are presented under the following headings:

Introduction
Examples of using meta summarize

Introduction

Meta-analysis helps answer research questions based on the results of multiple studies. Does exercise prolong life? Does lack of sleep increase the risk of cancer? Does daylight saving save energy? Or does performing the duck-face technique while taking a selfie increase the number of likes on Facebook? These (except perhaps the last one) and many other research questions have been investigated by multiple studies. These studies may have reported conflicting results: some may have shown effects in one direction, some in the opposite, and others may have shown none that are statistically significant. Meta-analysis uses quantitative methods to explore these conflicting results and, whenever possible, provide a unified conclusion based on the results of the individual studies.

Meta-analysis combines the results of similar multiple studies into a single result. Studies typically report some measures of outcomes, or effect sizes, and their precision (standard errors or CIs). Meta-analysis combines the individual effects sizes to provide various meta-analysis summaries. The main summaries are the overall effect size and its precision. Other meta-analysis summaries include the test of significance of the overall effect size, between-study heterogeneity summaries such as the I^2 statistic, and the test of homogeneity between studies. The `meta summarize` command reports such summaries.

Estimating the overall effect size, θ, and its precision based on the results of multiple studies is at the heart of meta-analysis. There are various methods for estimating θ, which depend on the research goals and model assumptions about the studies. The estimate of the overall (combined) ES is computed as the weighted average of the study-specific effect sizes, with larger weights given to more precise (larger) studies:

$$\hat{\theta} = \frac{\sum_{j=1}^{K} w_j \hat{\theta}_j}{\sum_{j=1}^{K} w_j}$$

The weights are determined by the chosen meta-analysis model, estimation method, and potentially the type of effect size; see *Methods and formulas* for details. (In [META] **Intro**, we used θ_{pop} to denote the population parameter of interest. For simplicity, here and in the rest of the documentation, we will use θ.)

As we described in *Meta-analysis models* in [META] **Intro**, the choice of a meta-analysis model is important not only for estimation but also for interpretation of $\hat{\theta}$. `meta summarize` supports random-effects (`random`), fixed-effects (`fixed`), and common-effect (`common`) meta-analysis models. Each meta-analysis model provides various estimation methods such as the random-effects REML method, `random(reml)`, and fixed-effects Mantel–Haenszel method, `fixed(mhaenszel)`. The default model and method are as declared with `meta set` or `meta esize`; see *Declaring a meta-analysis model* in [META] **meta data**. Note that the Mantel–Haenszel method is available only with effect sizes `lnoratio`, `lnrratio`, and `rdiff` declared by using `meta esize`; see [META] **meta esize**.

For random-effects models, you can perform sensitivity meta-analysis to explore the impact of different levels of heterogeneity on the results. You can use the `tau2(#)` option to specify different fixed values for the between-study variance τ^2. Or you can fix the percentage of variation in the effect sizes because of heterogeneity by specifying the values for the I^2 statistic in the `i2(#)` option. With random-effects models, you can also compute prediction intervals for $\hat{\theta}$, `predinterval(#)`, and use the alternative standard-error estimators, `se()`.

You can perform subgroup analysis, `subgroup()`, CMA, `cumulative()`, or leave-one-out meta-analysis, `leaveoneout`; see *Subgroup meta-analysis*, *Cumulative meta-analysis*, and *Leave-one-out meta-analysis* in [META] **Intro**. Also see *Subgroup meta-analysis*, *Cumulative meta-analysis*, and *Leave-one-out meta-analysis* in *Methods and formulas* below.

You can sort the studies based on variables of interest via option `sort()`. For example, use `sort(_meta_es)` or `sort(_meta_weight)` to display the results based on the ascending order of the study effect sizes or study weights, respectively.

You can specify the desired confidence level with `level()`; report exponentiated results by specifying `eform`; report a t test, instead of a z test, for the overall effect size by specifying `tdistribution`; and more.

In the next section, we demonstrate various usages of `meta summarize`.

Examples of using meta summarize

Recall the pupil IQ data (Raudenbush and Bryk 1985; Raudenbush 1984) described in *Effects of teacher expectancy on pupil IQ (pupiliq.dta)* of [META] **meta**. Here we will use its declared version and will focus on the demonstration of various options of `meta summarize` and explanation of its output.

```
. use https://www.stata-press.com/data/r18/pupiliqset
(Effects of teacher expectancy on pupil IQ; set with -meta set-)
. keep in 1/10
(9 observations deleted)
. meta query, short
-> meta set stdmdiff se , studylabel(studylbl) eslabel(Std. mean diff.)
  Effect-size label: Std. mean diff.
   Effect-size type: Generic
        Effect size: stdmdiff
          Std. err.: se
              Model: Random effects
             Method: REML
```

For brevity, we consider only the first 10 studies. We use `meta query, short` to remind us about the main settings of the declaration step. Our data were declared by using `meta set` with variables `stdmdiff` and `se` specifying the effect sizes and their standard errors, respectively. The declared meta-analysis model is the default random-effects model with the REML estimation method.

Examples are presented under the following headings:

Example 1: Default random-effects meta-analysis
Example 2: DerSimonian–Laird random-effects method
Example 3: Fixed-effects meta-analysis
Example 4: Common-effect meta-analysis
Example 5: Knapp–Hartung standard-error adjustment
Example 6: Prediction interval
Example 7: Sensitivity meta-analysis
Example 8: Other options: CI level, t distribution, sort, eform
Example 9: Subgroup meta-analysis
Example 10: Meta-analysis of correlations (StataNow)
Example 11: Meta-analysis of a single proportion and the transform() option
Example 12: Cumulative meta-analysis
Example 13: Leave-one-out meta-analysis

▷ Example 1: Default random-effects meta-analysis

We type `meta summarize` to obtain a standard meta-analysis summary.

```
. meta summarize
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
        Study label: studylbl
Meta-analysis summary                          Number of studies =     10
Random-effects model                           Heterogeneity:
Method: REML                                              tau2 =  0.0754
                                                       I2 (%) =   74.98
                                                           H2 =    4.00
                Effect size: Std. mean diff.
------------------------------------------------------------------------------
                       Study |     Effect size    [95% conf. interval]  % weight
-----------------------------+------------------------------------------------
  Rosenthal et al., 1974     |           0.030      -0.215       0.275      12.39
       Conn et al., 1968     |           0.120      -0.168       0.408      11.62
       Jose & Cody, 1971     |          -0.140      -0.467       0.187      10.92
Pellegrini & Hicks, 1972     |           1.180       0.449       1.911       5.25
Pellegrini & Hicks, 1972     |           0.260      -0.463       0.983       5.33
 Evans & Rosenthal, 1969     |          -0.060      -0.262       0.142      13.11
    Fielder et al., 1971     |          -0.020      -0.222       0.182      13.11
          Claiborn, 1969     |          -0.320      -0.751       0.111       9.11
            Kester, 1969     |           0.270      -0.051       0.591      11.02
            Maxwell, 1970    |           0.800       0.308       1.292       8.15
-----------------------------+------------------------------------------------
                       theta |           0.134      -0.075       0.342
------------------------------------------------------------------------------
Test of theta = 0: z = 1.26                  Prob > |z| = 0.2085
Test of homogeneity: Q = chi2(9) = 26.21      Prob > Q = 0.0019
```

As with other `meta` commands, a short information about meta settings is displayed directly following the `meta summarize` command. It can be suppressed with the `nometashow` option; see example 2.

Next, the header reports the information about the meta-analysis model and method, the number of studies (10), and several heterogeneity statistics. The output table reports the effect sizes and their 95% CIs for individual studies and the estimate of the overall, combined ES, labeled as `theta`, and its 95% CI. The test of significance of the overall effect size and the homogeneity test are reported at the bottom of the table.

Because our declared effect-size label, `Std. mean diff.`, was too long to fit as the column header, `meta summarize` used the generic column label `Effect size` but displayed the specified label in the table legend.

The mean effect size in our example is 0.134 with the 95% CI of $[-0.075, 0.342]$. This estimate is computed as the weighted average of the study-specific effect sizes, with the weights representing precision of the studies. The percentages of the total weight for each study are reported in the `% weight` column. The more precise the study is, the larger its weight percentage. For example, studies 6 and 7, with labels `Evans & Rosenthal, 1969` and `Fielder et al., 1971`, have the largest weight percentage among the studies of about 13% (each). Thus, their effect-size estimates, -0.06 and -0.02, have the largest weights in the weighted-average estimate.

The 95% CI for the overall estimate and the test of H_0: $\theta = 0$ with the z-test statistic of 1.26 and the p-value of 0.2085 suggest that θ is not statistically significantly different from 0. We should be careful, however, with our conclusions in the presence of between-study heterogeneity.

The heterogeneity statistic I^2, reported in the header, is about 75%, which means that 75% of the variability in the effect-size estimates is because of the between-study differences rather than the sampling variation. According to Higgins et al. (2003), this value of I^2 corresponds to "large heterogeneity". (The authors suggest that $I^2 = 25\%$ should indicate "small heterogeneity", $I^2 = 50\%$ indicate "medium heterogeneity", and $I^2 = 75\%$ indicate "large heterogeneity".) The between-study variance τ^2 is estimated to be 0.0754. The homogeneity test of H_0: $\theta_1 = \theta_2 = \cdots = \theta_{10}$ reports the Q test statistic of 26.21 with a p-value of 0.0019.

When there are few studies, which is typical in meta-analysis, the homogeneity test is known to have low power, which means that it may not detect clinically significant heterogeneity (Hedges and Pigott 2001). Thus, you should use caution when interpreting nonsignificant results as "no heterogeneity". In fact, many experts (for example, Berman and Parker [2002]) recommend using a 10% significance level instead of the classical 5% level to determine statistical significance when using this test. On the other hand, when there are many studies, this test is known to have excessive power, which means that it tends to detect heterogeneity that is clinically insignificant (Hardy and Thompson 1998).

In our example, the p-value of the homogeneity test is $0.0019 < 0.05 < 0.1$, so there is definitely statistical evidence of the between-study heterogeneity. See example 9 for one way to account for the heterogeneity.

◁

▷ Example 2: DerSimonian–Laird random-effects method

Continuing with example 1, let's use the DerSimonian–Laird random-effects method instead of the default (declared) REML method. Let's also suppress the meta setting information displayed at the top of the command output by using the `nometashow` option.

```
. meta summarize, random(dlaird) nometashow
Meta-analysis summary                          Number of studies =     10
Random-effects model                           Heterogeneity:
Method: DerSimonian-Laird                                   tau2 =  0.0481
                                                         I2 (%) =   65.66
                                                             H2 =    2.91

                Effect size: Std. mean diff.
```

Study	Effect size	[95% conf.	interval]	% weight
Rosenthal et al., 1974	0.030	-0.215	0.275	13.00
Conn et al., 1968	0.120	-0.168	0.408	11.88
Jose & Cody, 1971	-0.140	-0.467	0.187	10.90
Pellegrini & Hicks, 1972	1.180	0.449	1.911	4.42
Pellegrini & Hicks, 1972	0.260	-0.463	0.983	4.49
Evans & Rosenthal, 1969	-0.060	-0.262	0.142	14.11
Fielder et al., 1971	-0.020	-0.222	0.182	14.11
Claiborn, 1969	-0.320	-0.751	0.111	8.58
Kester, 1969	0.270	-0.051	0.591	11.04
Maxwell, 1970	0.800	0.308	1.292	7.45
theta	0.117	-0.061	0.296	

```
Test of theta = 0: z = 1.29                    Prob > |z| = 0.1967
Test of homogeneity: Q = chi2(9) = 26.21         Prob > Q = 0.0019
```

The results are now based on the DerSimonian–Laird method, and the header is updated to reflect this. This method is one of the many random-effects methods for estimating the between-study variance τ^2. Its estimate is 0.0481. In random-effects models, the weights depend on τ^2 and thus will differ across

different random-effects methods. The mean effect-size estimate under the DerSimonian–Laird method is 0.117 with the 95% CI of [−0.061, 0.296]. This estimate is similar to the 0.134 estimate we obtained in example 1. We also arrive at the same inferential conclusion of no statistical significance of the mean effect size as in the previous example.

To shorten the output, let's suppress the meta setting information from the output of `meta summarize` for all remaining examples. We can use `meta update` to update our current meta settings.

```
. quietly meta update, nometashow
```

We specified the `nometashow` option with `meta update` to suppress the display of the meta setting information in all `meta` commands; see *Modifying default meta settings* in [META] **meta data**.

Example 3: Fixed-effects meta-analysis

In example 1, we assumed a random-effects meta-analysis model. We can use the `fixed` option to specify a fixed-effects meta-analysis model.

```
. meta summarize, fixed
Meta-analysis summary                           Number of studies =     10
Fixed-effects model                             Heterogeneity:
Method: Inverse-variance                                  I2 (%) =   65.66
                                                          H2 =     2.91

                Effect size: Std. mean diff.
```

Study	Effect size	[95% conf.	interval]	% weight
Rosenthal et al., 1974	0.030	-0.215	0.275	15.13
Conn et al., 1968	0.120	-0.168	0.408	10.94
Jose & Cody, 1971	-0.140	-0.467	0.187	8.48
Pellegrini & Hicks, 1972	1.180	0.449	1.911	1.70
Pellegrini & Hicks, 1972	0.260	-0.463	0.983	1.74
Evans & Rosenthal, 1969	-0.060	-0.262	0.142	22.29
Fielder et al., 1971	-0.020	-0.222	0.182	22.29
Claiborn, 1969	-0.320	-0.751	0.111	4.89
Kester, 1969	0.270	-0.051	0.591	8.79
Maxwell, 1970	0.800	0.308	1.292	3.75
theta	0.051	-0.045	0.146	

```
Test of theta = 0: z = 1.04                     Prob > |z| = 0.2974
Test of homogeneity: Q = chi2(9) = 26.21          Prob > Q = 0.0019
```

As reported in the header, `fixed` implied the inverse-variance estimation method. The between-group variance parameter is not estimated with fixed-effects models, so the heterogeneity summary does not report `tau2`. Under this model, the mean effect-size estimate is 0.051 with the 95% CI of [−0.045, 0.146]. As we explain in *Comparison between the models and interpretation of their results* in [META] **Intro**, in a fixed-effects model, `theta` estimates the weighted average of the true study-specific standardized mean differences. Our interpretation is also limited to these 10 studies that we observed in our meta-analysis. That is, the weighted average of the standardized mean differences of these 10 studies is not statistically significantly different from 0.

▷ Example 4: Common-effect meta-analysis

From example 1 and example 3, we determined that there is substantial between-study variability in these data. Thus, a common-effect model, which assumes that all study-specific effects are the same, is not reasonable for these data. But we will demonstrate it for illustration purposes.

```
. meta summarize, common
Meta-analysis summary                          Number of studies =     10
Common-effect model
Method: Inverse-variance
                Effect size: Std. mean diff.
```

Study	Effect size	[95% conf.	interval]	% weight
Rosenthal et al., 1974	0.030	-0.215	0.275	15.13
Conn et al., 1968	0.120	-0.168	0.408	10.94
Jose & Cody, 1971	-0.140	-0.467	0.187	8.48
Pellegrini & Hicks, 1972	1.180	0.449	1.911	1.70
Pellegrini & Hicks, 1972	0.260	-0.463	0.983	1.74
Evans & Rosenthal, 1969	-0.060	-0.262	0.142	22.29
Fielder et al., 1971	-0.020	-0.222	0.182	22.29
Claiborn, 1969	-0.320	-0.751	0.111	4.89
Kester, 1969	0.270	-0.051	0.591	8.79
Maxwell, 1970	0.800	0.308	1.292	3.75
theta	0.051	-0.045	0.146	

```
Test of theta = 0: z = 1.04                  Prob > |z| = 0.2974
```

We use the `common` option to specify a common-effect model. Because this model implies no heterogeneity, the corresponding summaries and the homogeneity test are not reported for this model. As we point out in *Comparison between the models and interpretation of their results* in [META] **Intro**, a common-effect model is computationally the same as a fixed-effects model. So we obtain the exact same results as in example 3. However, the interpretation of our results is different. Here `theta` estimates a single effect, which is common to all studies. Although the two models produce the same results, to encourage proper interpretation, we provide both options, `common` and `fixed`, to distinguish between these models; see *Declaring a meta-analysis model* in [META] **meta data** for details.

◁

▷ Example 5: Knapp–Hartung standard-error adjustment

Let's return to our random-effects model from example 1. For random-effects models, `meta summarize` provides several additional options, which we explore in the next three examples.

The Knapp–Hartung adjustment (also known as the Sidik–Jonkman adjustment) to the standard error of the overall effect size (Knapp and Hartung 2003 and Hartung and Knapp 2001a, 2001b) is sometimes used in practice. We can specify it with the `se(khartung)` option. We also specify the `nostudies` option to suppress the output from individual studies because it is unaffected by the `se(khartung)` option.

```
. meta summarize, se(khartung) nostudies
Meta-analysis summary                                   Number of studies =     10
Random-effects model                                    Heterogeneity:
Method: REML                                                        tau2 =  0.0754
SE adjustment: Knapp-Hartung                                     I2 (%) =   74.98
                                                                    H2 =    4.00

        theta: Overall Std. mean diff.
------------------------------------------------------------------------------
             |    Estimate    Std. err.     t    P>|t|    [95% conf. interval]
-------------+----------------------------------------------------------------
       theta |    .1335309    .1215065    1.10   0.300   -.1413358    .4083976
------------------------------------------------------------------------------
Test of homogeneity: Q = chi2(9) = 26.21                  Prob > Q = 0.0019
```

Without the individual studies, the output table is slightly different. The test of significance is now reported in the output table instead of at the bottom of the output table.

The estimate `theta` is the same as in example 1, 0.134, but it is reported with more digits in this table. The confidence intervals and the test of significance are different. In addition to making an adjustment to the standard error, Knapp and Hartung also use a Student's t distribution as a sampling distribution instead of the normal distribution. Thus, the t statistic is reported in the output table instead of the z statistic. Regardless, we still conclude that our overall effect size is not statistically significant.

Another standard error adjustment, also used in practice, is the so-called truncated or modified Knapp–Hartung adjustment; see *Methods and formulas* for details. This adjustment can be specified with the `se(khartung, truncated)` option.

```
. meta summarize, se(khartung, truncated)
  (output omitted)
```

◁

▷ Example 6: Prediction interval

Recall from *Random-effects model* in [META] **Intro** that a random-effects model implies that the observed studies in a meta-analysis represent a sample from a larger population of similar studies. What if we want to estimate the plausible ranges for the overall effect size in a new, future study? We cannot use the confidence interval for the overall effect size because it does not incorporate the uncertainty in estimating the between-study variance, which is important if we want to predict an effect in a new study. We can compute the prediction interval.

```
. meta summarize, predinterval(90) nostudies
Meta-analysis summary                                   Number of studies =     10
Random-effects model                                    Heterogeneity:
Method: REML                                                        tau2 =  0.0754
                                                                 I2 (%) =   74.98
                                                                    H2 =    4.00

        theta: Overall Std. mean diff.
------------------------------------------------------------------------------
             |    Estimate    Std. err.     z    P>|z|    [95% conf. interval]
-------------+----------------------------------------------------------------
       theta |    .1335309    .1061617    1.26   0.208   -.0745422    .3416041
------------------------------------------------------------------------------
90% prediction interval for theta: [-0.414, 0.681]
Test of homogeneity: Q = chi2(9) = 26.21                  Prob > Q = 0.0019
```

We specified predinterval(90) to compute the 90% prediction interval for the mean effect size; use predinterval to compute the 95% interval. Following example 5, we also used nostudies to suppress individual studies.

The 90% prediction interval, reported at the bottom of the table, is $[-0.414, 0.681]$. The prediction interval will be wider than the confidence interval because it additionally accounts for the uncertainty in the between-study variability.

◁

▷ Example 7: Sensitivity meta-analysis

For random-effects models, we can perform sensitivity analysis to explore various levels of heterogeneity between studies. Let's see how our results change for different values of the between-study variance τ^2 and the heterogeneity statistic I^2.

Let's compute the results assuming that τ^2 equals 0.25.

```
. meta summarize, tau2(0.25) nostudies
Sensitivity meta-analysis summary                   Number of studies =     10
Random-effects model                                Heterogeneity:
Method: User-specified tau2                                    tau2 =  0.2500
                                                             I2 (%) =   90.86
                                                                 H2 =   10.94

        theta: Overall Std. mean diff.
------------------------------------------------------------------------------
             |    Estimate    Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       theta |     .173588     .171407     1.01   0.311    -.1623636     .5095395
------------------------------------------------------------------------------
Test of homogeneity: Q = chi2(9) = 26.21                  Prob > Q = 0.0019
```

Our estimate of the mean effect size is 0.174 with the 95% CI of $[-0.162, 0.51]$ compared with 0.134 with the 95% CI of $[-0.075, 0.342]$ from example 1.

The specified value of τ^2 corresponds to an I^2 of about 91%. Let's now compute the results assuming I^2 of 10%.

```
. meta summarize, i2(10) nostudies
Sensitivity meta-analysis summary                   Number of studies =     10
Random-effects model                                Heterogeneity:
Method: User-specified I2                                      tau2 =  0.0028
                                                             I2 (%) =   10.00
                                                                 H2 =    1.11

        theta: Overall Std. mean diff.
------------------------------------------------------------------------------
             |    Estimate    Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       theta |    .0589369    .0527232     1.12   0.264    -.0443987     .1622724
------------------------------------------------------------------------------
Test of homogeneity: Q = chi2(9) = 26.21                  Prob > Q = 0.0019
```

The estimate of the mean effect size is 0.059 with the 95% CI of $[-0.044, 0.162]$. The corresponding τ^2 value is 0.0028.

In both cases above, the mean effect size is not statistically significant.

◁

▷ Example 8: Other options: CI level, t distribution, sort, eform

`meta summarize` provides other options such as `level()` to temporarily change the declared confidence level and `tdistribution` to use a Student's t distribution as the sampling distribution instead of the default normal distribution.

```
. meta summarize, level(90) tdistribution
Meta-analysis summary                           Number of studies =     10
Random-effects model                            Heterogeneity:
Method: REML                                                tau2 =  0.0754
                                                         I2 (%) =   74.98
                                                             H2 =    4.00

                Effect size: Std. mean diff.

------------------------------------------------------------------------------
                       Study |  Effect size  [90% conf. interval]  % weight
-----------------------------+------------------------------------------------
  Rosenthal et al., 1974     |       0.030      -0.176       0.236      12.39
       Conn et al., 1968     |       0.120      -0.122       0.362      11.62
       Jose & Cody, 1971     |      -0.140      -0.415       0.135      10.92
Pellegrini & Hicks, 1972     |       1.180       0.566       1.794       5.25
Pellegrini & Hicks, 1972     |       0.260      -0.347       0.867       5.33
 Evans & Rosenthal, 1969     |      -0.060      -0.229       0.109      13.11
    Fielder et al., 1971     |      -0.020      -0.189       0.149      13.11
          Claiborn, 1969     |      -0.320      -0.682       0.042       9.11
            Kester, 1969     |       0.270       0.000       0.540      11.02
           Maxwell, 1970     |       0.800       0.387       1.213       8.15
-----------------------------+------------------------------------------------
                       theta |       0.134      -0.061       0.328
------------------------------------------------------------------------------
Test of theta = 0: t(9) = 1.26                   Prob > |t| = 0.2401
Test of homogeneity: Q = chi2(9) = 26.21           Prob > Q = 0.0019
```

Notice that all CIs, including those for the individual studies, now correspond to the 90% confidence level, compared with example 1. Also, the significance test now uses the Student's t distribution with 9 degrees of freedom, but the conclusion remains the same—the mean effect size is not statistically significant.

You may also find `meta summarize`'s option `eform` useful when dealing with the effect sizes in the log-transformed metric such as log odds-ratios or log risk-ratios. By default, `meta summarize` reports results in the declared metric, which should be chosen such that the sampling distributions of the effect sizes are well approximated by normal distributions. It may be more convenient, however, to display the final results in the original metric. When you specify the `eform` option, it reports the exponentiated results and the corresponding CIs. Note that the significance tests and other summary measures are still computed based on the nonexponentiated results.

It does not make sense to exponentiate standardized mean differences in our example, but we will do this just to demonstrate the option.

We will also use the sort() option to sort our results based on the descending order of study weights, with larger, more precise studies appearing first.

```
. meta summarize, eform sort(_meta_weight, descending)
Meta-analysis summary                          Number of studies =     10
Random-effects model                           Heterogeneity:
Method: REML                                               tau2 =  0.0754
                                                        I2 (%) =   74.98
                                                            H2 =    4.00

                    exp(ES): exp(Std. mean diff.)
------------------------------------------------------------------------------
                       Study |     exp(ES)    [95% conf. interval]  % weight
-----------------------------+------------------------------------------------
 Evans & Rosenthal, 1969     |        0.942       0.770       1.152      13.11
    Fielder et al., 1971     |        0.980       0.801       1.199      13.11
  Rosenthal et al., 1974     |        1.030       0.807       1.317      12.39
       Conn et al., 1968     |        1.127       0.845       1.504      11.62
            Kester, 1969     |        1.310       0.950       1.807      11.02
       Jose & Cody, 1971     |        0.869       0.627       1.206      10.92
          Claiborn, 1969     |        0.726       0.472       1.118       9.11
           Maxwell, 1970     |        2.226       1.361       3.640       8.15
Pellegrini & Hicks, 1972     |        1.297       0.629       2.673       5.33
Pellegrini & Hicks, 1972     |        3.254       1.567       6.760       5.25
-----------------------------+------------------------------------------------
                exp(theta)   |        1.143       0.928       1.407
------------------------------------------------------------------------------
Sorted by: _meta_weight
Test of theta = 0: z = 1.26                    Prob > |z| = 0.2085
Test of homogeneity: Q = chi2(9) = 26.21         Prob > Q = 0.0019
```

meta summarize, eform reports exponentiated effect sizes and their corresponding CIs. It labels the effect-size column as exp(ES), but you can change this label to *string* by specifying eform(*string*).

Note that the eform option worked in our example because meta set declared our precomputed effect sizes as generic. They could have been log odds-ratios, in which case eform would make perfect sense. However, if you use meta esize to compute the standardized mean differences (for example, Hedges's g) and try to use eform with meta summarize, you will receive an error message because meta summarize knows that exponentiation is not appropriate with effect sizes that correspond to continuous data. With effect sizes lnoratio (or lnorpeto) and lnrratio computed by meta esize, you can also use the respective options or and rr, which are synonyms for eform in those cases. These options (and eform) will label your results as Odds ratio (Peto's OR) and Risk ratio.

◁

▷ Example 9: Subgroup meta-analysis

In example 1 and example 3, we identified the presence of substantial heterogeneity between the observed studies. Sometimes, the heterogeneity can be explained by some study-level covariates, also known as moderators. With categorical moderators, we can perform subgroup analysis, which performs meta-analysis separately for each category of each moderator.

We have binary variable week1, which records whether teachers had prior contact with students for more than 1 week or for 1 week or less. Let's use this variable as the moderator in our subgroup analysis. We specify the variable week1 in the subgroup() option.

```
. meta summarize, subgroup(week1)
Subgroup meta-analysis summary                    Number of studies =     10
Random-effects model
Method: REML
Group: week1
             Effect size: Std. mean diff.
```

Study	Effect size	[95% conf.	interval]	% weight
Group: <= 1 week				
Pellegrini & Hicks, 1972	1.180	0.449	1.911	5.25
Pellegrini & Hicks, 1972	0.260	-0.463	0.983	5.33
Kester, 1969	0.270	-0.051	0.591	11.02
Maxwell, 1970	0.800	0.308	1.292	8.15
theta	0.581	0.174	0.989	
Group: > 1 week				
Rosenthal et al., 1974	0.030	-0.215	0.275	12.39
Conn et al., 1968	0.120	-0.168	0.408	11.62
Jose & Cody, 1971	-0.140	-0.467	0.187	10.92
Evans & Rosenthal, 1969	-0.060	-0.262	0.142	13.11
Fielder et al., 1971	-0.020	-0.222	0.182	13.11
Claiborn, 1969	-0.320	-0.751	0.111	9.11
theta	-0.033	-0.137	0.071	
Overall				
theta	0.134	-0.075	0.342	

Heterogeneity summary

Group	df	Q	P > Q	tau2	% I2	H2
<= 1 week	3	7.14	0.068	0.095	57.03	2.33
> 1 week	5	3.53	0.618	0.000	0.00	1.00
Overall	9	26.21	0.002	0.075	74.98	4.00

Test of group differences: Q_b = chi2(1) = 8.18 Prob > Q_b = 0.004

We now have two output tables. Our main table now reports results from individual studies separately for each group, in addition to the group-specific overall effect size. The overall effect size computed using all studies is reported at the bottom under `Overall`.

The second table reports the group-specific and overall heterogeneity summaries. The test of group differences is reported at the bottom of this table.

The estimated `theta` for the group with contact `<= 1 week` is 0.581 with the 95% CI of $[0.174, 0.989]$. The mean effect size in this group is statistically significant at the 5% level. The estimated `theta` for the group with contact `> 1 week` is -0.033 with the 95% CI of $[-0.137, 0.071]$. The mean effect size in this group is not statistically significant at the 5% level.

If we look at the heterogeneity summaries, the `<= 1 week` group still has some unexplained between-study heterogeneity with an estimated I^2 of 57% and a p-value of the homogeneity test of $0.068 < 0.1$. There does not appear to be any between-study heterogeneity in the `> 1 week` group: I^2 is essentially 0%, and the homogeneity test p-value is 0.618.

We should interpret our results with caution because each subgroup analysis used a few studies, with the <= 1 week group having only 4 studies.

We can specify multiple variables in the subgroup() option. Let's also include variable tester in our subgroup analysis.

```
. meta summarize, subgroup(week1 tester)
Subgroup meta-analysis summary                    Number of studies =     10
Random-effects model
Method: REML
Group: week1 tester
```

Group	No. of studies	Std. mean diff.	[95% conf.	interval]	p-value
week1					
<= 1 week	4	0.581	0.174	0.989	0.005
> 1 week	6	-0.033	-0.137	0.071	0.535
tester					
Aware	7	0.059	-0.129	0.247	0.535
Blind	3	0.316	-0.206	0.837	0.235
Overall					
theta	10	0.134	-0.075	0.342	0.208

Heterogeneity summary

Group	df	Q	P > Q	tau2	% I2	H2
week1						
<= 1 week	3	7.14	0.068	0.095	57.03	2.33
> 1 week	5	3.53	0.618	0.000	0.00	1.00
tester						
Aware	6	16.35	0.012	0.035	59.07	2.44
Blind	2	9.31	0.009	0.154	75.14	4.02
Overall	9	26.21	0.002	0.075	74.98	4.00

Tests of group differences

	df	Q_b	P > Q_b
week1	1	8.18	0.004
tester	1	0.82	0.365

With more than one variable in subgroup(), meta summarize reports three output tables. To conserve space, the main table does not report individual studies but reports the number of studies in each group. It also reports the p-values of the corresponding significance tests of the overall effect sizes in each group.

The heterogeneity table reports the group summaries for each variable, in addition to the overall summaries. The new table reports the results of tests of subgroup differences for each variable.

The studies appear to be homogeneous across the levels of the tester variable.

◁

▷ Example 10: Meta-analysis of correlations (StataNow)

Continuing with the dataset in example 5 of [META] **meta data**, we wish to produce a meta-analysis summary and compute the overall correlation between conscientiousness and medication adherence.

```
. use https://www.stata-press.com/data/r18/adherence
(Conscientiousness and medication adherence)
. describe rho n studylbl
```

Variable name	Storage type	Display format	Value label	Variable label
rho	double	%9.0g		* Correlation coefficient
n	int	%9.0g		Sample size of the study
studylbl	str26	%26s		Study label

The correlation coefficient `rho` is measured on the natural scale ($-1 \leq r \leq 1$). Many meta-analysts (for example, Borenstein and Hedges [2019]) recommend working with the Fisher's z-transformed correlations instead of the raw correlations:

$$z = \frac{1}{2}\log\left(\frac{1+\texttt{rho}}{1-\texttt{rho}}\right) = \texttt{atanh(rho)} \sim N\left(0, \frac{1}{\texttt{n}-3}\right)$$

If the underlying data are bivariate normal, the variance of z equals $1/(\texttt{n}-3)$ and depends only on the within-study sample size and not on the correlation parameter itself. Below, we use the first specification of `meta esize` in example 5 of [META] **meta data** to compute Fisher's z-transformed correlations:

```
. meta esize rho n, fisherz studylabel(studylbl) nometashow
```

The meta-analysis summary may be obtained as follows:

```
. meta summarize
```

```
Meta-analysis summary                          Number of studies =     16
Random-effects model                           Heterogeneity:
Method: REML                                              tau2 =  0.0081
                                                       I2 (%) =   61.73
                                                           H2 =    2.61
```

Study	Fisher's z	[95% conf.	interval]	% weight
Axelsson et al. (2009)	0.189	-0.001	0.380	5.68
Axelsson et al. (2011)	0.163	0.092	0.235	10.54
Bruce et al. (2010)	0.354	0.082	0.626	3.64
Christensen et al. (1999)	0.332	0.139	0.524	5.62
Christensen & Smith (1995)	0.277	0.041	0.513	4.41
Cohen et al. (2004)	0.000	-0.249	0.249	4.11
Dobbels et al. (2005)	0.177	0.027	0.327	7.14
Ediger et al. (2007)	0.050	-0.059	0.159	8.89
Insel et al. (2006)	0.266	0.002	0.530	3.79
Jerant et al. (2011)	0.010	-0.061	0.081	10.58
Moran et al. (1997)	-0.090	-0.359	0.179	3.69
O'Cleirigh et al. (2007)	0.388	0.179	0.597	5.11
Penedo et al. (2003)	0.000	-0.184	0.184	5.87
Quine et al. (2012)	0.151	0.066	0.236	9.98
Stilley et al. (2004)	0.245	0.087	0.402	6.84
Wiebe & Christensen (1997)	0.040	-0.209	0.289	4.11
theta	0.150	0.088	0.212	

```
Test of theta = 0: z = 4.75                          Prob > |z| = 0.0000
Test of homogeneity: Q = chi2(15) = 38.16              Prob > Q = 0.0009
```

The overall Fisher's z value (transformed correlation coefficient) across the 16 studies is estimated to be 0.150 using the REML RE meta-analysis model.

The interpretation of the results, however, is easier in the natural correlation-coefficient metric, which we can compute using the inverse transformation:

$$\texttt{rho} = \frac{\exp(2z)-1}{\exp(2z)+1} = \texttt{tanh}(z)$$

Thus, you may obtain the value of the correlation coefficient and its CI by typing

```
. display tanh(r(theta))
.14880413
. display "[" tanh(r(ci_lb)) ", " tanh(r(ci_ub)) "]"
[.08783366, .20866384]
```

More conveniently, you can use the `correlation` option to report correlations. This option applies the hyperbolic tangent (`tanh()`) transformation to the Fisher's z-values and labels the resulting effect sizes as `Correlation`. Notice that specifying `correlation` is equivalent to specifying `transform(corr)` or `transform(Correlation: tanh)`.

```
. meta summarize, correlation
Meta-analysis summary                           Number of studies =     16
Random-effects model                            Heterogeneity:
Method: REML                                                tau2 =  0.0081
                                                         I2 (%) =   61.73
                                                             H2 =    2.61

------------------------------------------------------------------------------
                       Study |  Correlation  [95% conf. interval]  % weight
-----------------------------+------------------------------------------------
      Axelsson et al. (2009) |        0.187       -0.001        0.362      5.68
      Axelsson et al. (2011) |        0.162        0.091        0.231     10.54
         Bruce et al. (2010) |        0.340        0.082        0.555      3.64
   Christensen et al. (1999) |        0.320        0.139        0.481      5.62
Christensen & Smith (1995)   |        0.270        0.041        0.472      4.41
         Cohen et al. (2004) |        0.000       -0.244        0.244      4.11
       Dobbels et al. (2005) |        0.175        0.027        0.316      7.14
        Ediger et al. (2007) |        0.050       -0.059        0.158      8.89
         Insel et al. (2006) |        0.260        0.002        0.486      3.79
        Jerant et al. (2011) |        0.010       -0.061        0.081     10.58
         Moran et al. (1997) |       -0.090       -0.345        0.177      3.69
  O'Cleirigh et al. (2007)   |        0.370        0.178        0.535      5.11
       Penedo et al. (2003)  |        0.000       -0.182        0.182      5.87
        Quine et al. (2012)  |        0.150        0.066        0.232      9.98
      Stilley et al. (2004)  |        0.240        0.087        0.382      6.84
Wiebe & Christensen (1997)   |        0.040       -0.206        0.281      4.11
-----------------------------+------------------------------------------------
                tanh(theta)  |        0.149        0.088        0.209
------------------------------------------------------------------------------
Test of theta = 0: z = 4.75                          Prob > |z| = 0.0000
Test of homogeneity: Q = chi2(15) = 38.16              Prob > Q = 0.0009
```

The overall correlation value is 0.149 with a CI of $[0.088, 0.209]$.

◁

▷ Example 11: Meta-analysis of a single proportion and the transform() option

Continuing from the `meta esize ndeaths pensize` setting in example 4 of [META] **meta data**, we produce a meta-analysis summary and compute the overall proportion as follows:

```
. meta summarize, proportion
  Effect-size label: Freeman-Tukey's p
        Effect size: _meta_es
          Std. err.: _meta_se
Meta-analysis summary                     Number of studies =      4
Random-effects model                      Heterogeneity:
Method: REML                                          tau2 =  0.0000
                                                   I2 (%) =    0.00
                                                      H2 =    1.00

------------------------------------------------------------------------------
       Study |  Proportion   [95% conf. interval]  % weight
-------------+----------------------------------------------------------------
     Study 1 |       0.273       0.044       0.579     20.18
     Study 2 |       0.353       0.140       0.598     30.70
     Study 3 |       0.476       0.264       0.693     37.72
     Study 4 |       0.167       0.145       0.586     11.40
-------------+----------------------------------------------------------------
 invftukey(t |       0.360       0.230       0.499
------------------------------------------------------------------------------
Test of theta = 0: z = 7.67                  Prob > |z| = 0.0000
Test of homogeneity: Q = chi2(3) = 2.18       Prob > Q = 0.5368
```

The overall proportion is estimated to be 0.360 with a CI of $[0.230, 0.499]$.

The `proportion` option was used to report proportions instead of the Freeman–Tukey-transformed proportions. This option is equivalent to `transform(invftukey, hmean)`, where `hmean` specifies that the harmonic mean of the study-specific sample sizes be used as n_θ to back-transform the overall effect size [see (4) in *Inverse Freeman–Tukey transformation* for details]. Instead of the harmonic mean, Barendregt et al. (2013) suggested to use the inverse of the variance of the overall Freeman–Tukey-transformed proportion as an estimate of n_θ. This may be requested via `transform(invftukey, ivariance)`.

```
. meta summarize, transform(invftukey, ivariance)
  Effect-size label: Freeman-Tukey's p
        Effect size: _meta_es
          Std. err.: _meta_se
Meta-analysis summary                     Number of studies =      4
Random-effects model                      Heterogeneity:
Method: REML                                          tau2 =  0.0000
                                                   I2 (%) =    0.00
                                                      H2 =    1.00

------------------------------------------------------------------------------
       Study |  Proportion   [95% conf. interval]  % weight
-------------+----------------------------------------------------------------
     Study 1 |       0.273       0.044       0.579     20.18
     Study 2 |       0.353       0.140       0.598     30.70
     Study 3 |       0.476       0.264       0.693     37.72
     Study 4 |       0.167       0.145       0.586     11.40
-------------+----------------------------------------------------------------
 invftukey(t |       0.369       0.247       0.499
------------------------------------------------------------------------------
Note: Method ivariance is used to compute overall proportion.
Test of theta = 0: z = 8.89                  Prob > |z| = 0.0000
Test of homogeneity: Q = chi2(3) = 2.18       Prob > Q = 0.5368
```

Finally, the CIs for the Freeman–Tukey-transformed proportions are the standard normal-based Wald intervals. These are stored in system variables _meta_cil and _meta_ciu. The CIs displayed in the table above are the corresponding back-transformed (using transform(invftukey)) confidence intervals in the proportion metric, and these are stored in _meta_cil_transf and _meta_ciu_transf.

When you report proportions either via the proportion or transform() option, you can use the citype() option to display other types of CIs for the study proportions. Below, we display Wilson CIs for the study proportions.

```
. meta summarize, transform(invftukey, ivariance) citype(wilson)
  Effect-size label: Freeman-Tukey's p
        Effect size: _meta_es
          Std. err.: _meta_se
Meta-analysis summary                     Number of studies =      4
Random-effects model                      Heterogeneity:
Method: REML                                          tau2 =  0.0000
                                                   I2 (%) =    0.00
                                                       H2 =    1.00
```

Study	Proportion	Wilson [95% conf.	interval]	% weight
Study 1	0.273	0.097	0.566	20.18
Study 2	0.353	0.173	0.587	30.70
Study 3	0.476	0.283	0.676	37.72
Study 4	0.167	0.030	0.564	11.40
invftukey(theta)	0.369	0.247	0.499	

```
Note: Method ivariance is used to compute overall proportion.
Note: Wilson CIs are reported only for individual studies.
Test of theta = 0: z = 8.89                  Prob > |z| = 0.0000
Test of homogeneity: Q = chi2(3) = 2.18        Prob > Q = 0.5368
```

The citype() option applies to the CIs of individual studies only and not to the CI of the overall proportion.

◁

▷ Example 12: Cumulative meta-analysis

CMA (Lau et al. 1992 ; Sterne 2016) performs multiple meta-analyses by accumulating studies one at a time. The studies are first ordered with respect to a variable of interest, the ordering variable. Meta-analysis summaries are then computed for the first study, for the first two studies, for the first three studies, and so on. The last meta-analysis will correspond to the standard meta-analysis using all studies.

CMA is useful, for instance, for identifying the point in time of the potential change in the direction or significance of the effect size when the ordering variable is time. You can use the cumulative() option to perform CMA.

For demonstration purposes, let's continue with the dataset in example 1 and use year as our ordering variable.

```
. meta summarize, cumulative(year)
Cumulative meta-analysis summary                    Number of studies =     10
Random-effects model
Method: REML
Order variable: year
               theta: Overall Std. mean diff.
```

Study	theta	[95% conf.	interval]	p-value	year
Conn et al., 1968	0.120	-0.168	0.408	0.414	1968
Evans & Rosent~1969	-0.001	-0.166	0.165	0.995	1969
Claiborn, 1969	-0.042	-0.201	0.117	0.605	1969
Kester, 1969	0.022	-0.177	0.221	0.830	1969
Maxwell, 1970	0.140	-0.178	0.459	0.389	1970
Jose & Cody, 1971	0.089	-0.177	0.355	0.510	1971
Fielder et al., 1~1	0.064	-0.141	0.270	0.539	1971
Pellegrini & H~1972	0.161	-0.117	0.438	0.257	1972
Pellegrini & H~1972	0.161	-0.090	0.413	0.208	1972
Rosenthal et.., 1~4	0.134	-0.075	0.342	0.208	1974

The output table reports the overall effect size and its CIs for each cumulative analysis. The p-value column contains the p-values of the significance tests of the overall effect sizes from these analyses. The last column displays the values of the ordering variable.

In our example, no particular trend is apparent.

We can perform stratified CMA by specifying a categorical variable in cumulative()'s option by(). To demonstrate, we also specify cumulative()'s option descending to list results in descending order of year.

```
. meta summarize, cumulative(year, by(week1) descending)
Stratified cumulative meta-analysis summary         Number of studies =     10
Random-effects model
Method: REML
Order variable: year (descending)
Stratum: week1
               theta: Overall Std. mean diff.
```

Study	theta	[95% conf.	interval]	p-value	year
Group: <= 1 week					
Pellegrini & H~1972	0.260	-0.463	0.983	0.481	1972
Pellegrini & H~1972	0.718	-0.183	1.620	0.118	1972
Maxwell, 1970	0.755	0.320	1.190	0.001	1970
Kester, 1969	0.581	0.174	0.989	0.005	1969
Group: > 1 week					
Rosenthal et.., 1~4	0.030	-0.215	0.275	0.810	1974
Fielder et al., 1~1	0.000	-0.156	0.156	0.998	1971
Jose & Cody, 1971	-0.026	-0.166	0.115	0.720	1971
Claiborn, 1969	-0.054	-0.188	0.080	0.429	1969
Evans & Rosent~1969	-0.056	-0.167	0.056	0.326	1969
Conn et al., 1968	-0.033	-0.137	0.071	0.535	1968

CMA is performed separately for each group of week1.

Also see *Cumulative meta-analysis* in [META] **meta**.

Example 13: Leave-one-out meta-analysis

For each study in the meta-analysis, the corresponding leave-one-out meta-analysis will omit that study and perform a meta-analysis on the remaining set of studies ($k-1$ studies). It is useful for exploring the influence of a single study on the overall effect size estimate.

Continuing with example 1, we will use option `leaveoneout` to perform a leave-one-out meta-analysis and sort our results according to variable `se` so that larger studies appear first.

```
. meta summarize, leaveoneout sort(se)
Leave-one-out meta-analysis summary          Number of studies =     10
Random-effects model
Method: REML
                    theta: Overall Std. mean diff.
```

Omitted study	theta	[95% conf.	interval]	p-value
Evans & Rosenthal, 1969	0.172	-0.073	0.418	0.169
Fielder et al., 1971	0.168	-0.081	0.418	0.186
Rosenthal et al., 1974	0.161	-0.090	0.413	0.208
Conn et al., 1968	0.149	-0.102	0.400	0.244
Kester, 1969	0.127	-0.115	0.368	0.304
Jose & Cody, 1971	0.174	-0.060	0.408	0.146
Claiborn, 1969	0.175	-0.036	0.386	0.105
Maxwell, 1970	0.021	-0.076	0.119	0.665
Pellegrini & Hicks, 1972	0.132	-0.095	0.358	0.254
Pellegrini & Hicks, 1972	0.057	-0.090	0.204	0.446
theta	0.134	-0.075	0.342	0.208

```
Sorted by: se
```

The output table reports the overall effect size and its CIs for each leave-one-out analysis. In this example, the first row reports the overall effect size estimate based on all the studies excluding the `Evans & Rosenthal, 1969` study ($10 - 1 = 9$ studies). The `p-value` column contains the p-values of the significance tests of the overall effect sizes from these analyses. The last row displays the results based on all 10 studies. It seems that the `Maxwell, 1970` study has a relatively large influence because the 95% CI from the meta-analysis excluding that study, $[-0.076, 0.119]$, does not contain the overall effect size estimate based on all studies, 0.134.

Stored results

meta summarize stores the following in r():

Scalars

r(N)	number of observations
r(theta)	overall effect size
r(se)	standard error of overall effect size
r(ci_lb)	lower CI bound for overall effect size
r(ci_ub)	upper CI bound for overall effect size
r(tau2)	between-study variance
r(I2)	I^2 heterogeneity statistic (not for CE model)
r(H2)	H^2 heterogeneity statistic (not for CE model)
r(z)	z statistic for test of significance of overall effect size (when se() not specified)
r(t)	t statistic for test of significance of overall effect size (when se() specified)
r(df)	degrees of freedom for t distribution
r(p)	p-value for test of significance of overall effect size
r(Q)	Cochran's Q heterogeneity test statistic (not for CE model)
r(df_Q)	degrees of freedom for heterogeneity test
r(p_Q)	p-value for heterogeneity test
r(Q_b)	Cochran's Q statistic for test of group differences (for subgroup() with one variable)
r(df_Q_b)	degrees of freedom for test of group differences
r(p_Q_b)	p-value for test of group differences
r(seadj)	standard error adjustment
r(level)	confidence level for CIs
r(pi_lb)	lower bound of prediction interval
r(pi_ub)	upper bound of prediction interval
r(pilevel)	confidence level for prediction interval
r(converged)	1 if converged, 0 otherwise (with iterative random-effects methods)

Macros

r(model)	meta-analysis model
r(method)	meta-analysis estimation method
r(citype)	type of CI used in option citype() for meta-analysis of a single proportion
r(subgroupvars)	names of subgroup-analysis variables
r(ordervar)	name of order variable used in option cumulative()
r(byvar)	name of variable used in suboption by() within option cumulative()
r(direction)	ascending or descending
r(seadjtype)	type of standard error adjustment

Matrices

r(esgroup)	ESs and CIs from subgroup analysis
r(hetgroup)	heterogeneity summary from subgroup analysis
r(diffgroup)	results for tests of group differences from subgroup analysis
r(cumul)	results from cumulative meta-analysis
r(leaveoneout)	results from leave-one-out meta-analysis
r(pi_info)	prediction intervals from subgroup analysis

meta summarize also creates a system variable, _meta_weight, which contains study weights. When the transform() option is specified, meta summarize creates system variables _meta_es_transf, _meta_cil_transf, and _meta_ciu_transf, which contain the transformed effect sizes and lower and upper bounds of the corresponding transformed CIs.

Also see *Stored results* in [META] **meta set** and *Stored results* in [META] **meta esize** for other system variables.

Methods and formulas

Methods and formulas are presented under the following headings:

Fixed-effects and common-effect methods for combining study estimates
  Inverse-variance method
  Mantel–Haenszel method for two-group comparison of binary outcomes
  Peto's method for odds ratios
Random-effects methods for combining study estimates
  Iterative methods
  Noniterative methods
  Knapp–Hartung standard-error adjustment
  Prediction intervals
Confidence intervals and significance test
Heterogeneity measures
Inverse Freeman–Tukey transformation
Homogeneity test
Subgroup meta-analysis
  Fixed-effects model
  Random-effects model
Cumulative meta-analysis
Leave-one-out meta-analysis

The formulas and methods below are based on Veroniki et al. (2016), Viechtbauer et al. (2015), Borenstein et al. (2009), Schwarzer, Carpenter, and Rücker (2015), Kontopantelis and Reeves (2016), Fisher (2016), and Bradburn, Deeks, and Altman (2016).

Fixed-effects and common-effect methods for combining study estimates

Consider the data from K independent studies. Let $\hat{\theta}_j$ be the estimate of the population effect size θ_j reported by the jth study and $\hat{\sigma}_j^2$ be the corresponding estimate of the within-study variance, which is equal to the squared standard error of $\hat{\theta}_j$. $\hat{\theta}_j$ is one of Hedges's g_j, Cohen's d_j, $\ln\left(\widehat{\text{OR}}_j\right)$, $\ln\left(\widehat{\text{RR}}_j\right)$, and so on, as defined in *Methods and formulas* of [META] **meta esize**, or a generic (precomputed) effect size as declared by [META] **meta set**.

Consider a fixed-effects model (Hedges and Vevea 1998; Rice, Higgins, and Lumley 2018) from *Meta-analysis models* in [META] **Intro**,

$$\hat{\theta}_j = \theta_j + \epsilon_j \qquad \epsilon_j \sim N\left(0, \hat{\sigma}_j^2\right)$$

where $\hat{\sigma}_j^2$'s are treated as known values that do not require estimation. Under the assumption that $\theta_1 = \theta_2 = \cdots = \theta_K = \theta$, the above fixed-effects model simplifies to a common-effect model (Hedges 1982; Rosenthal and Rubin 1982):

$$\hat{\theta}_j = \theta + \epsilon_j \qquad \epsilon_j \sim N\left(0, \hat{\sigma}_j^2\right)$$

The estimation methods we describe below are the same for the two models, but the interpretation of the estimates is different; see *Comparison between the models and interpretation of their results* in [META] **Intro**. The two models estimate different population parameters. A common-effect model estimates the common effect $\theta_{\text{pop}} = \theta$, whereas a fixed-effects model estimates a weighted average of the study-specific effects $\hat{\theta}_j$'s,

$$\theta_{\text{pop}} = \text{Ave}(\theta_j) = \frac{\sum_{j=1}^{K} W_j \theta_j}{\sum_{j=1}^{K} W_j}$$

where W_j's represent true, unknown weights, which are defined in Rice, Higgins, and Lumley (2018, eq. 3). For simplicity, in what follows, we will use θ to mean θ_{pop}.

Inverse-variance method

Under the inverse-variance method, the MLE of θ is

$$\hat{\theta}_{\text{IV}} = \frac{\sum_{j=1}^{K} \hat{\theta}_j/\hat{\sigma}_j^2}{\sum_{j=1}^{K} 1/\hat{\sigma}_j^2} = \frac{\sum_{j=1}^{K} w_j\hat{\theta}_j}{\sum_{j=1}^{K} w_j}$$

where the weight $w_j = 1/\hat{\sigma}_j^2$ is used to estimate the true weight W_j. The inverse-variance method takes its name from the weights being the reciprocal of the effect-size variances.

The variance estimate of $\hat{\theta}_{\text{IV}}$

$$\widehat{\text{Var}}\left(\hat{\theta}_{\text{IV}}\right) = \frac{1}{w_{\cdot}}$$

where $w_{\cdot} = \sum_{j=1}^{K} w_j$.

Mantel–Haenszel method for two-group comparison of binary outcomes

For meta-analysis that compares two binary outcomes, the Mantel–Haenszel method can be used to combine odds ratios (OR), risk ratios (RR), and risk differences (RD) instead of the inverse-variance method. The classical Mantel–Haenszel method (Mantel and Haenszel 1959) is used for OR, and its extension by Greenland and Robins (1985) is used for RR and RD. The Mantel–Haenszel method may be preferable with sparse data (Emerson 1994). This is the default pooling method in `meta esize` for the effect sizes mentioned above with fixed-effects and common-effect models.

Consider the following 2×2 table for the jth study.

group	event	no event	size
treatment	a_j	b_j	$n_{1j} = a_j + b_j$
control	c_j	d_j	$n_{2j} = c_j + d_j$

The sample size for the jth study is denoted by $n_j = n_{1j} + n_{2j}$.

For the overall risk difference, the formula is

$$\hat{\theta}_{\text{MH}} = \frac{\sum_{j=1}^{K} w_j^{(\text{MH})} \times \hat{\theta}_j}{\sum_{j=1}^{K} w_j^{(\text{MH})}}$$

where $\hat{\theta}_j$ is $\widehat{\text{RD}}$ from the jth study.

Unlike the inverse-variance method, with log odds-ratios and log risk-ratios, the Mantel–Haenszel method combines the individual effect sizes in the original metric and then takes the log to obtain the final overall log odds-ratio or log risk-ratio estimate,

$$\hat{\theta}_{\text{MH}} = \ln\left\{\frac{\sum_{j=1}^{K} w_j^{(\text{MH})} \times \exp(\hat{\theta}_j)}{\sum_{j=1}^{K} w_j^{(\text{MH})}}\right\}$$

where $\hat{\theta}_j$ is $\ln(\widehat{\text{OR}})$ or $\ln\left(\widehat{\text{RR}}\right)$ from the jth study.

The MH weights are defined as follows. In the formula for the overall risk difference, the weight assigned to each study is

$$w_j^{(\text{MH})} = \frac{n_{1j}n_{2j}}{n_j}$$

For the overall log risk-ratio, the jth weight is given by

$$w_j^{(\text{MH})} = \frac{n_{1j}c_j}{n_j}$$

And for the overall log odds-ratio, the jth weight is given by

$$w_j^{(\text{MH})} = \frac{b_j c_j}{n_j}$$

An estimator of the variance of the overall risk difference $\hat{\theta}_{\text{MH}} = \widehat{\text{RD}}_{\text{MH}}$ (Greenland and Robins 1985) is

$$\widehat{\text{Var}}\left(\widehat{\text{RD}}_{\text{MH}}\right) = \frac{\sum_{j=1}^{K}\left(a_j b_j n_{2j}^3 + c_j d_j n_{1j}^3\right)/n_{1j}n_{2j}n_j^2}{\left(\sum_{j=1}^{K} n_{1j}n_{2j}/n_j\right)^2}$$

An estimator of the variance of the overall log risk-ratio $\hat{\theta}_{\text{MH}} = \ln(\widehat{\text{RR}}_{\text{MH}})$ (Greenland and Robins 1985) is

$$\widehat{\text{Var}}\left\{\ln(\widehat{\text{RR}}_{\text{MH}})\right\} = \frac{\sum_{j=1}^{K}\left\{n_{1j}n_{2j}\left(a_j + c_j\right) - a_j c_j n_j\right\}/n_j^2}{\left(\sum_{j=1}^{K} a_j n_{2j}/n_j\right) \times \left(\sum_{j=1}^{K} c_j n_{1j}/n_j\right)}$$

And an estimator of the variance of the overall log odds-ratio $\hat{\theta}_{\text{MH}} = \ln(\widehat{\text{OR}}_{\text{MH}})$ (Robins, Breslow, and Greenland 1986; Robins, Greenland, and Breslow 1986) is

$$\widehat{\text{Var}}\left\{\ln(\widehat{\text{OR}}_{\text{MH}})\right\} = \frac{\sum_{j=1}^{K} P_j R_j}{2\left(\sum_{j=1}^{K} R_j\right)^2} + \frac{\sum_{j=1}^{K}\left(P_j S_j + Q_j R_j\right)}{2\sum_{j=1}^{K} R_j \sum_{j=1}^{K} S_j} + \frac{\sum_{j=1}^{K} Q_j S_j}{2\left(\sum_{j=1}^{K} S_j\right)^2}$$

where

$$P_j = \frac{a_j + d_j}{n_j}, \quad Q_j = \frac{b_j + c_j}{n_j}, \quad R_j = \frac{a_j d_j}{n_j}, \quad \text{and } S_j = \frac{b_j c_j}{n_j}$$

Greenland and Robins (1985) and Robins, Breslow, and Greenland (1986) demonstrate consistency of all the above variance estimators in the two cases they call a sparse-data limiting model, in which the number of 2×2 tables (studies) increases but the cell sizes remain fixed, and a large-strata limiting model, in which the number of studies remains fixed but individual cell sizes increase.

Peto's method for odds ratios

An alternative to the Mantel–Haenszel method for combining odds ratios is the Peto's method (Peto et al. 1977 ; Yusuf et al. 1985). It is based on the inverse-variance method but uses an alternate way to compute the odds ratios (and consequently the log odds-ratio).

Let $\ln\left(\widehat{\text{OR}}_j^{\text{Peto}}\right)$ be Peto's log odds-ratio for the jth study as defined in *Odds ratio* in [META] **meta esize**. Then, Peto's overall log odds-ratio is defined following the inverse-variance method as follows,

$$\hat{\theta}_{\text{Peto}} = \ln\left(\widehat{\text{OR}}^{\text{Peto}}\right) = \frac{\sum_{j=1}^{K} w_j \ln\left(\widehat{\text{OR}}_j^{\text{Peto}}\right)}{\sum_{j=1}^{K} w_j}$$

where $w_j = 1/\hat{\sigma}_j^2 = \text{Var}(a_j)$ and $\text{Var}(a_j)$ is as defined in *Methods and formulas* of [META] **meta esize** of [META] **meta esize**.

The variance estimate is

$$\widehat{\text{Var}}\left\{\ln\left(\widehat{\text{OR}}^{\text{Peto}}\right)\right\} = \frac{1}{\sum_{j=1}^{K} w_j}$$

Random-effects methods for combining study estimates

Suppose that the observed study-specific effect sizes represent a random sample from a population of effect sizes that is normally distributed with mean θ and variance τ^2.

Consider a random-effects model (Hedges 1983; DerSimonian and Laird 1986) from *Meta-analysis models* in [META] **Intro**,

$$\hat{\theta}_j = \theta_j + \epsilon_j = \theta + u_j + \epsilon_j$$

where ϵ_j and u_j are assumed to be independent with $\epsilon_j \sim N\left(0, \hat{\sigma}_j^2\right)$ and $u_j \sim N\left(0, \tau^2\right)$.

The overall effect $E(\hat{\theta}_j) = \theta$ is estimated as the weighted average,

$$\hat{\theta}^* = \frac{\sum_{j=1}^{K} w_j^* \hat{\theta}_j}{\sum_{j=1}^{K} w_j^*} \tag{1}$$

where $w_j^* = 1/\left(\hat{\sigma}_j^2 + \hat{\tau}^2\right)$. The variance of $\hat{\theta}^*$ is estimated by

$$\widehat{\text{Var}}\left(\hat{\theta}^*\right) = \frac{1}{w_\cdot^*}$$

where $w_\cdot^* = \sum_{j=1}^{K} w_j^*$.

Different estimators of the between-study variance, τ^2, lead to different estimators of θ. `meta summarize` supports seven estimation methods of τ^2. Three methods are iterative: the maximum likelihood (ML) estimator (Hardy and Thompson 1996); the restricted maximum-likelihood (REML) estimator (Raudenbush 2009); and the empirical Bayes (EB) estimator (Morris 1983; Berkey et al. 1995), also known as the Paule–Mandel estimator (Paule and Mandel 1982). Four methods are noniterative (have a closed-form expression): DerSimonian–Laird (DL) estimator (DerSimonian and Laird 1986); Hedges estimator (HE) (Hedges 1983; Hedges and Olkin 1985), also known as the Cochran estimator or variance-component estimator; Hunter–Schmidt (HS) estimator (Schmidt and Hunter 2015); and Sidik–Jonkman (SJ) estimator (Sidik and Jonkman 2005).

The formulas for and properties of these estimators have been discussed at length in Veroniki et al. (2016). Expressions for these estimators are given in the more general context of meta-regression in *Methods and formulas* of [META] **meta regress**. Below, we provide the simplified expressions when no covariates (moderators) are included in the regression model. The simplified expressions were obtained by replacing the **X** matrix with $K \times 1$ column vector of 1s.

Iterative methods

The ML method (Hardy and Thompson 1996; Thompson and Sharp 1999) computes the MLE of τ^2 by maximizing the following log-likelihood function,

$$\ln L_{\text{ML}}\left(\tau^2\right) = -\frac{K}{2}\ln(2\pi) - \frac{1}{2}\sum_{j=1}^{K}\ln\left(\hat{\sigma}_j^2 + \tau^2\right) - \frac{1}{2}\sum_{j=1}^{K}\frac{\left(\hat{\theta}_j - \hat{\theta}^*\right)^2}{\hat{\sigma}_j^2 + \tau^2}$$

with respect to τ^2, where $\hat{\theta}^*$ is defined in (1) and is based on the current value of $\hat{\tau}^2$.

The ML method is asymptotically efficient but may produce biased results in small samples. The REML method estimates τ^2 by accounting for the uncertainty in the estimation of θ, which leads to nearly an unbiased estimate of τ^2.

The REML log-likelihood function is

$$\ln L_{\text{REML}}\left(\tau^2\right) = \ln L_{\text{ML}}\left(\tau^2\right) - \frac{1}{2}\ln\left\{\sum_{j=1}^{K}\left(\hat{\sigma}_j^2 + \tau^2\right)^{-1}\right\} + \frac{\ln(2\pi)}{2}$$

The EB estimator and a description of the iterative process for each estimator in this section is presented in the *Methods and formulas* of [META] **meta regress**.

Noniterative methods

The methods in this section do not make any assumptions about the distribution of the random effects. They also do not require any iteration.

The most popular noniterative estimation method is the DL method. This is a method of moment estimator for τ^2, and it is defined as follows,

$$\hat{\tau}^2_{\text{DL}} = \frac{Q - (K-1)}{\sum_{j=1}^{K} w_j - \sum_{j=1}^{K} w_j^2 / \sum_{j=1}^{K} w_j}$$

where $Q = \sum_{j=1}^{K} w_j\left(\hat{\theta}_j - \hat{\theta}_{\text{IV}}\right)^2$ and $w_j = 1/\hat{\sigma}_j^2$.

Because $\hat{\tau}^2_{\text{DL}}$ is negative when $Q < K-1$, it is truncated at 0 in practice, and thus $\max\left(0, \hat{\tau}^2_{\text{DL}}\right)$ is used to estimate the between-study variance:

$$\hat{\tau}^2_{\text{DL}} = \max\left\{0, \frac{\sum_{j=1}^{K} w_j\left(\hat{\theta}_j - \hat{\theta}_{\text{IV}}\right)^2 - (K-1)}{\sum_{j=1}^{K} w_j - \sum_{j=1}^{K} w_j^2 / \sum_{j=1}^{K} w_j}\right\}$$

The HE estimator is another method of moment estimator defined as follows,

$$\hat{\tau}^2_{\text{HE}} = \max\left\{0, \frac{1}{K-1}\sum_{j=1}^{K}\left(\hat{\theta}_j - \overline{\theta}\right)^2 - \frac{1}{K}\sum_{j=1}^{K}\hat{\sigma}_j^2\right\}$$

where $\overline{\theta} = \left(\sum_{j=1}^{K}\hat{\theta}_j\right)/K$.

The HS estimator is given by

$$\hat{\tau}^2_{\text{HS}} = \max\left\{0, \frac{Q-K}{\sum_{j=1}^{K} w_j}\right\}$$

For the SJ estimator, consider an initial estimate of τ^2, given by

$$\hat{\tau}^2_0 = \frac{\sum_{j=1}^{K}\left(\hat{\theta}_j - \overline{\theta}\right)^2}{K}$$

Then, the estimator is defined as

$$\hat{\tau}^2_{\text{SJ}} = \frac{\sum_{j=1}^{K} w_j^{\text{SJ}}\left(\hat{\theta}_j - \hat{\theta}^{\text{SJ}}\right)^2}{K-1}$$

where $w_j^{\text{SJ}} = \hat{\tau}^2_0/\left(\hat{\sigma}^2_j + \hat{\tau}^2_0\right)$ and $\hat{\theta}^{\text{SJ}} = \sum_{j=1}^{K} w_j^{\text{SJ}}\hat{\theta}_j/\sum_{j=1}^{K} w_j^{\text{SJ}}$.

Knapp–Hartung standard-error adjustment

Hartung and Knapp (2001a) and Sidik and Jonkman (2002) proposed an adjustment to the variance of $\hat{\theta}^*$ to account for the uncertainty in estimating τ^2, which is used in the expression for weights. They proposed to multiply $\widehat{\text{Var}}(\hat{\theta}^*) = 1/w^*_{\cdot}$ by the following quadratic form,

$$q_{\text{KH}} = \frac{1}{K-1}\sum_{j=1}^{K} w_j^*\left(\hat{\theta}_j - \hat{\theta}^*\right)^2$$

or by $\max(1, q_{\text{KH}})$.

The variance estimator for $\hat{\theta}^*$ can then be defined as

$$\widehat{\text{Var}}_{\text{HK}}\left(\hat{\theta}^*\right) = \begin{cases} q_{\text{KH}} \times 1/w^*_{\cdot} & \text{with option } \texttt{se(khartung)} \\ \max(1, q_{\text{KH}}) \times 1/w^*_{\cdot} & \text{with option } \texttt{se(khartung, truncated)} \end{cases}$$

Hartung (1999) established that the statistic

$$\frac{\hat{\theta}^* - \theta}{\sqrt{\widehat{\text{Var}}_{\text{HK}}\left(\hat{\theta}^*\right)}}$$

has a Student's t distribution with $K-1$ degrees of freedom.

Correspondingly, the $(1-\alpha)\times 100\%$ CI for θ using the Knapp–Hartung standard error is

$$\hat{\theta}^* \pm t_{K-1,1-\alpha/2}\sqrt{\widehat{\text{Var}}_{\text{HK}}\left(\hat{\theta}^*\right)}$$

where $t_{K-1,1-\alpha/2}$ denotes the $1-\alpha/2$ quantile of the Student's t distribution with $K-1$ degrees of freedom.

The test statistic for the significance test of an overall effect, H_0: $\theta = 0$, is

$$\frac{\hat{\theta}^*}{\sqrt{\widehat{\text{Var}}_{\text{HK}}\left(\hat{\theta}^*\right)}}$$

and has the Student's t distribution with $K-1$ degrees of freedom.

Also see Sidik and Jonkman (2002, 2003) and Cornell et al. (2014) for more discussion about the Knapp–Hartung adjustment.

Prediction intervals

In a random-effects model, you can compute a prediction interval (Higgins, Thompson, and Spiegelhalter 2009) that estimates plausible ranges for θ in a future study. Compared with the CI, a prediction interval incorporates the uncertainty in estimating τ^2 in the computation.

A $(1-\alpha)\times 100\%$ prediction interval is defined as

$$\hat{\theta}^* \pm t_{K-2,1-\alpha/2}\sqrt{\widehat{\text{Var}}\left(\hat{\theta}^*\right) + \hat{\tau}^2}$$

where $t_{K-2,1-\alpha/2}$ denotes the $1-\alpha/2$ quantile of the Student's t distribution with $K-2$ degrees of freedom. This prediction interval may be specified with the `predinterval()` option.

Confidence intervals and significance test

Let $\hat{\theta}$ be any of the estimators considered in the previous sections such as $\hat{\theta}_{\text{IV}}$ or $\hat{\theta}^*$. The $(1-\alpha)\times 100\%$ confidence interval for θ is

$$\hat{\theta} \pm z_{1-\alpha/2}\sqrt{\widehat{\text{Var}}\left(\hat{\theta}\right)}$$

where $z_{1-\alpha/2}$ is the $(1-\alpha/2)$th quantile of the standard normal distribution.

We reject the hypothesis of no treatment effect H_0: $\theta = 0$ at level α, if

$$\frac{\left|\hat{\theta}\right|}{\sqrt{\widehat{\text{Var}}\left(\hat{\theta}\right)}} > z_{1-\alpha/2}$$

If the `tdistribution` option is specified, the $z_{1-\alpha/2}$ critical value is replaced with the $t_{K-1,1-\alpha/2}$ critical value in the above formulas.

Heterogeneity measures

The homogeneity test can be used to test whether the study-specific effects are the same; see *Homogeneity test*. But with a small number of studies, this test may have low power (Hedges and Pigott 2001). Also, it does not provide an estimate of the magnitude of the between-study heterogeneity. Some authors (for example, Higgins and Thompson [2002] and Higgins et al. [2003]) suggest examining the heterogeneity statistics rather than relying solely on the homogeneity test.

Higgins and Thompson (2002) proposed two heterogeneity measures: I^2 and H^2. We define them separately for random-effects and fixed-effects models.

For a random-effects model, the two heterogeneity measures are defined as follows:

$$I^2 = \frac{\hat{\tau}^2}{\hat{\tau}^2 + s^2} \times 100\% \tag{2}$$

and

$$H^2 = \frac{\hat{\tau}^2 + s^2}{s^2} \tag{3}$$

where

$$s^2 = \frac{K-1}{\sum_{j=1}^{K} w_j - \sum_{j=1}^{K} w_j^2 / \sum_{j=1}^{K} w_j}$$

is the within-study variance and $\hat{\tau}^2$ is an estimator of the between-study variance. The values of I^2 and H^2 will vary depending on which estimator of $\hat{\tau}^2$ is specified in the `random()` option.

For a fixed-effects model, the expressions for I^2 and H^2 are given by

$$I^2 = \left\{ \frac{Q - (K-1)}{Q} \right\} \times 100\%$$

and

$$H^2 = \frac{Q}{K-1}$$

where Q is defined in *Homogeneity test*.

The formulas above for I^2 and H^2 are equivalent to the corresponding formulas (2) and (3), when the DL method is used to estimate τ^2. I^2 is negative when $Q < (K-1)$ and is thus reset to zero in that case.

Inverse Freeman–Tukey transformation

For each study, let $\hat{p}_{\text{FT}}$ be the Freeman–Tukey-transformed proportion as defined in *Freeman–Tukey-transformed proportion* in *Methods and formulas* in [META] **meta esize**.

The inverse Freeman–Tukey transformation, which back-transforms $\hat{p}_{\text{FT}}$ to a proportion (option `transform(invftukey)`), is given by (Miller 1978)

$$\hat{p} = 0.5 \left\{ 1 - \text{sgn}\left(\cos \hat{p}_{\text{FT}}\right) \sqrt{1 - \left(\sin \hat{p}_{\text{FT}} + \frac{\sin \hat{p}_{\text{FT}} - \frac{1}{\sin \hat{p}_{\text{FT}}}}{n} \right)^2} \right\}$$

where sgn is the sign operator. The expression depends on the study sample size n, which is available for each study but not for the overall (pooled) effect size. To back-transform the overall effect size $\hat{\theta}$, where $\hat{\theta}$ is obtained by pooling the study-specific $\hat{p}_{\text{FT}}$'s, to obtain the overall proportion, Miller (1978) suggested to use n_θ, the harmonic mean (default) of the study-specific sample sizes, in place of n in the above formula. Other estimators for n_θ include the geometric mean, arithmetic mean, or the inverse of the variance of the overall effect size.

Because $0 \leq e \leq n$, each study's $\hat{p}_{\text{FT}}$ must be between $\texttt{asin}\{\sqrt{1/(n+1)}\}$ and $\texttt{asin}\{\sqrt{n/(n+1)}\} + \pi/2$ [see (1) in [META] **meta esize**]. Thus, the above back-transformation is valid only if $\texttt{asin}\{\sqrt{1(n_\theta+1)}\} \leq \hat{\theta} \leq \texttt{asin}\{\sqrt{n_\theta/(n_\theta+1)}\} + (\pi/2)$. Therefore, in practice, the overall proportion, $\hat{p}_{\text{ov}}$, is computed as follows:

$$\hat{p}_{\text{ov}} = \begin{cases} 0 & \text{if } \hat{\theta} < \texttt{asin}\left(\sqrt{\frac{1}{n_\theta+1}}\right) \\ 1 & \text{if } \hat{\theta} > \texttt{asin}\left(\sqrt{\frac{n_\theta}{n_\theta+1}}\right) + \frac{\pi}{2} \\ 0.5\left\{1 - \text{sgn}\left(\cos\hat{\theta}\right)\sqrt{1 - \left(\sin\hat{\theta} + \frac{\sin\hat{\theta} - \frac{1}{\sin\hat{\theta}}}{n_\theta}\right)^2}\right\} & \text{otherwise} \end{cases} \tag{4}$$

Because $\hat{\theta}$ can be bounded away from 0 whenever $\hat{\theta} > \texttt{asin}\{\sqrt{1/(n_\theta+1)}\}$, the test statistic for H_0: $\theta = 0$ is adjusted as follows:

$$\frac{\left|\hat{\theta} - \texttt{asin}\left(\sqrt{\frac{1}{n_\theta+1}}\right)\right|}{\sqrt{\widehat{\text{Var}}\left(\hat{\theta}\right)}}$$

Homogeneity test

Consider a test of H_0: $\theta_1 = \theta_2 = \cdots = \theta_K = \theta$, known as the homogeneity test, that evaluates whether the effect sizes are the same across the studies. It uses the following test statistic,

$$Q = \sum_{j=1}^{K} w_j\left(\hat{\theta}_j - \hat{\theta}\right)^2 = \sum_{j=1}^{K} w_j\hat{\theta}_j^2 - \frac{\left(\sum_{j=1}^{K} w_j\hat{\theta}_j\right)^2}{\sum_{j=1}^{K} w_j}$$

where $w_j = 1/\hat{\sigma}_j^2$, and $\hat{\theta}_j$ and $\hat{\theta}$ depend on the type of the effect size chosen.

Under the null hypothesis of homogeneity, Q follows a χ^2 distribution with $K-1$ degrees of freedom.

Hedges and Pigott (2001) showed that the test has low power when the number of studies (K) is small, which is typical in meta-analysis. This means that the null hypothesis of homogeneity is not rejected as often as it should be. Thus, for the homogeneity test, the meta-analysis literature (for example, Petitti [2001]; Berman and Parker [2002]; Sutton and Higgins [2008]) suggests using the significance level $\alpha = 0.1$ instead of the conventional $\alpha = 0.05$.

The homogeneity test checks for the potential presence of heterogeneity but does not estimate the magnitude of the heterogeneity. Thus, many authors (for example, Higgins and Thompson [2002]; Higgins et al. [2003]) suggest exploring the heterogeneity statistics rather than solely relying on the test. See *Heterogeneity measures*.

Subgroup meta-analysis

When the `subgroup(`*varname*`)` option is specified, we assume that the K studies are partitioned into L subgroups defined by *varname*. Estimates of the overall effect size and their corresponding standard errors are calculated for each of the L subgroups.

Let $\hat{\theta}_{jl}$ be the effect-size estimate from study j within subgroup l and $\hat{\sigma}^2_{jl}$ be the corresponding variance, where $l = 1, 2, \ldots, L$ and $j = 1, 2, \ldots, K_l$.

Below, we describe the formulas separately for fixed-effects and random-effects models. The formulas for the common-effect model are the same as for the fixed-effects model. When you specify a common-effect model with subgroup analysis, this model is assumed within each subgroup $l = 1, 2, \ldots, L$, but not for the entire sample of studies.

Fixed-effects model

In what follows, we assume the inverse-variance method, but the same principles apply to the Mantel–Haenszel method.

In subgroup analysis, a fixed-effects model may be formulated as

$$\hat{\theta}_{jl} = \theta_{jl} + \epsilon_{jl}, \quad \epsilon_{jl} \sim N\left(0, \hat{\sigma}^2_{jl}\right)$$

For the lth group, $\hat{\theta}_{\text{IV},l}$ is a weighted average of the effect sizes $\hat{\theta}_{jl}$ with weights $w_{jl} = 1/\hat{\sigma}^2_{jl}$:

$$\hat{\theta}_{\text{IV},l} = \frac{\sum_{j=1}^{K_l} w_{jl}\hat{\theta}_{jl}}{\sum_{j=1}^{K_l} w_{jl}}$$

The variance estimate of $\hat{\theta}_{\text{IV},l}$ is

$$\widehat{\text{Var}}\left(\hat{\theta}_{\text{IV},l}\right) = \frac{1}{w_{.l}}$$

where $w_{.l} = \sum_{j=1}^{K_l} w_{jl}$.

Other meta-analytic quantities such as I^2_l and Q_l may also be computed for the lth subgroup just as we described in the previous sections.

The Cochran's Q statistic can be extended to test for differences between the L subgroups:

$$Q_b = \sum_{l=1}^{L} w_{.l} \left(\hat{\theta}_{\text{IV},l} - \frac{\sum_{l=1}^{L} w_{.l} \hat{\theta}_{\text{IV},l}}{\sum_{l=1}^{L} w_{.l}} \right)^2$$

The subscript b in Q_b stands for "between" to emphasize that Q_b tests for "between-group" differences.

Under the null hypothesis of homogeneity between the subgroups ($\theta_{.1} = \theta_{.2} = \cdots = \theta_{.L} = \theta$), the statistic Q_b has a χ^2 distribution with $L - 1$ degrees of freedom.

Random-effects model

Consider a random-effects model with L subgroups and separate between-study variances τ_l^2:

$$\hat{\theta}_{jl} = \theta_{.l} + u_{jl} + \epsilon_{jl} \qquad \epsilon_{jl} \sim N\left(0, \hat{\sigma}_{jl}^2\right) \qquad u_{jl} \sim N\left(0, \tau_l^2\right)$$

The formulas for the random-effects model are the same as for the above fixed-effects model, except we replace the weights with the random-effects weights.

The estimate, $\hat{\theta}_l^*$, and its variance in the lth group are

$$\hat{\theta}_l^* = \frac{\sum_{j=1}^{K_l} w_{jl}^* \hat{\theta}_{jl}}{\sum_{j=1}^{K_l} w_{jl}^*}$$

$$\widehat{\text{Var}}\left(\hat{\theta}_l^*\right) = \frac{1}{w_{.l}^*}$$

where $w_{jl}^* = 1/(\hat{\sigma}_{jl}^2 + \hat{\tau}_l^2)$ and $w_{.l}^* = \sum_{j=1}^{K_l} w_{jl}^*$.

The Cochran's statistic for testing differences between the L subgroups is defined as

$$Q_b^* = \sum_{l=1}^{L} w_{.l}^* \left(\hat{\theta}_l^* - \frac{\sum_{l=1}^{L} w_{.l}^* \hat{\theta}_l^*}{\sum_{l=1}^{L} w_{.l}^*} \right)^2$$

Under the null hypothesis of homogeneity between the subgroups ($\theta_{.1} = \theta_{.2} = \cdots = \theta_{.L} = \theta$), Q_b^* has a χ^2 distribution with $L - 1$ degrees of freedom.

Also see Borenstein et al. (2009, chap. 19) and Schwarzer, Carpenter, and Rücker (2015).

Cumulative meta-analysis

To perform CMA, we first sort the studies in ascending order according to the values of the variable specified in the `cumulative()` option. If suboption `descending` is specified within the `cumulative()` option, the order is reversed. Mathematically, this corresponds to sorting the pairs $\left(\hat{\theta}_j, \hat{\sigma}_j^2\right)$ in the specified order. Let $\left(\hat{\theta}_j^s, \hat{\sigma}_j^{2,s}\right)$ denote the sorted pairs.

CMA estimates K overall effect sizes $\hat{\theta}^c_j$'s as follows,

$$\begin{aligned}
\hat{\theta}^c_1 &= \hat{\theta}^s_1 \\
\hat{\theta}^c_2 &= \text{MA}\left(\hat{\theta}^s_1, \hat{\theta}^s_2\right) \\
\hat{\theta}^c_3 &= \text{MA}\left(\hat{\theta}^s_1, \hat{\theta}^s_2, \hat{\theta}^s_3\right) \\
&\vdots \\
\hat{\theta}^c_j &= \text{MA}\left(\hat{\theta}^s_1, \hat{\theta}^s_2, \hat{\theta}^s_3, \ldots, \hat{\theta}^s_j\right) \\
&\vdots \\
\hat{\theta}^c_K &= \text{MA}\left(\hat{\theta}^s_1, \hat{\theta}^s_2, \hat{\theta}^s_3, \ldots, \hat{\theta}^s_K\right)
\end{aligned}$$

where $\text{MA}\left(\hat{\theta}^s_1, \hat{\theta}^s_2, \hat{\theta}^s_3, \ldots, \hat{\theta}^s_j\right)$ denotes a meta-analysis applied to the sorted studies 1 through j. Note that the meta-analysis also depends on the values $\hat{\sigma}^{2,s}_j$ but we omitted them from $\text{MA}()$ for notational convenience.

If suboption by(*byvar*) is specified within the `cumulative()` option, the above procedure is repeated for each subgroup defined by variable *byvar*.

Leave-one-out meta-analysis

Leave-one-out meta-analysis estimates K overall effect sizes $\hat{\theta}_{-j}$'s as follows,

$$\begin{aligned}
\hat{\theta}_{-1} &= \text{MA}\left(\hat{\theta}_2, \hat{\theta}_3, \ldots, \hat{\theta}_K\right) \\
\hat{\theta}_{-2} &= \text{MA}\left(\hat{\theta}_1, \hat{\theta}_3, \ldots, \hat{\theta}_K\right) \\
&\vdots \\
\hat{\theta}_{-j} &= \text{MA}\left(\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_{j-1}, \hat{\theta}_{j+1}, \ldots, \hat{\theta}_K\right) \\
&\vdots \\
\hat{\theta}_{-K} &= \text{MA}\left(\hat{\theta}_1, \hat{\theta}_2, \hat{\theta}_3, \ldots, \hat{\theta}_{K-1}\right)
\end{aligned}$$

where $\text{MA}\left(\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_{j-1}, \hat{\theta}_{j+1}, \ldots, \hat{\theta}_K\right)$ denotes a meta-analysis applied to all the studies except the jth study. Note that the meta-analysis also depends on the values $\hat{\sigma}^2_j$, but we omitted them from $\text{MA}()$ for notational convenience.

References

Barendregt, J. J., S. A. Doi, Y. Y. Lee, R. E. Norman, and T. Vos. 2013. Meta-analysis of prevalence. *Journal of Epidemiology and Community Health* 67: 974–978. https://doi.org/10.1136/jech-2013-203104.

Berkey, C. S., D. C. Hoaglin, F. Mosteller, and G. A. Colditz. 1995. A random-effects regression model for meta-analysis. *Statistics in Medicine* 14: 395–411. https://doi.org/10.1002/sim.4780140406.

Berman, N. G., and R. A. Parker. 2002. Meta-analysis: Neither quick nor easy. *BMC Medical Research Methodology* 2: art. 10. https://doi.org/10.1186/1471-2288-2-10.

Borenstein, M., and L. V. Hedges. 2019. "Effect sizes for meta-analysis". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 207–243. 3rd ed. New York: Russell Sage Foundation.

Borenstein, M., L. V. Hedges, J. P. T. Higgins, and H. R. Rothstein. 2009. *Introduction to Meta-Analysis*. Chichester, UK: Wiley.

Bradburn, M. J., J. J. Deeks, and D. G. Altman. 2016. "metan—A command for meta-analysis in Stata". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 4–28. 2nd ed. College Station, TX: Stata Press.

Cornell, J. E., C. D. Mulrow, A. R. Localio, C. B. Stack, A. R. Meibohm, E. Guallar, and S. N. Goodman. 2014. Random-effects meta-analysis of inconsistent effects: A time for change. *Annals of Internal Medicine* 160: 267–270. https://doi.org/10.7326/M13-2886.

DerSimonian, R., and N. M. Laird. 1986. Meta-analysis in clinical trials. *Controlled Clinical Trials* 7: 177–188. https://doi.org/10.1016/0197-2456(86)90046-2.

Emerson, J. D. 1994. Combining estimates of the odds ratio: The state of the art. *Statistical Methods in Medical Research* 3: 157–178. https://doi.org/10.1177/096228029400300204.

Fisher, D. J. 2016. "Two-stage individual participant data meta-analysis and generalized forest plots". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 280–307. 2nd ed. College Station, TX: Stata Press.

Greenland, S., and J. M. Robins. 1985. Estimation of a common effect parameter from sparse follow-up data. *Biometrics* 41: 55–68. https://doi.org/10.2307/2530643.

Hardy, R. J., and S. G. Thompson. 1996. A likelihood approach to meta-analysis with random effects. *Statistics in Medicine* 15: 619–629. https://doi.org/10.1002/(SICI)1097-0258(19960330)15:6<619::AID-SIM188>3.0.CO;2-A.

———. 1998. Detecting and describing heterogeneity in meta-analysis. *Statistics in Medicine* 17: 841–856. https://doi.org/10.1002/(sici)1097-0258(19980430)17:8<841::aid-sim781>3.0.co;2-d.

Hartung, J. 1999. An alternative method for meta-analysis. *Biometrical Journal* 41: 901–916. https://doi.org/10.1002/(SICI)1521-4036(199912)41:8<901::AID-BIMJ901>3.0.CO;2-W.

Hartung, J., and G. Knapp. 2001a. On tests of the overall treatment effect in meta-analysis with normally distributed responses. *Statistics in Medicine* 20: 1771–1782. https://doi.org/10.1002/sim.791.

———. 2001b. A refined method for the meta-analysis of controlled clinical trials with binary outcome. *Statistics in Medicine* 20: 3875–3889. https://doi.org/10.1002/sim.1009.

Hedges, L. V. 1982. Estimation of effect size from a series of independent experiments. *Psychological Bulletin* 92: 490–499. https://doi.org/10.1037/0033-2909.92.2.490.

———. 1983. A random effects model for effect sizes. *Psychological Bulletin* 93: 388–395. https://doi.org/10.1037/0033-2909.93.2.388.

Hedges, L. V., and I. Olkin. 1985. *Statistical Methods for Meta-Analysis*. Orlando, FL: Academic Press.

Hedges, L. V., and T. D. Pigott. 2001. The power of statistical tests in meta-analysis. *Psychological Methods* 6: 203–217. https://doi.org/10.1037/1082-989X.6.3.203.

Hedges, L. V., and J. L. Vevea. 1998. Fixed- and random-effects models in meta-analysis. *Psychological Methods* 3: 486–504. https://doi.org/10.1037/1082-989X.3.4.486.

Higgins, J. P. T., and S. G. Thompson. 2002. Quantifying heterogeneity in a meta-analysis. *Statistics in Medicine* 21: 1539–1558. https://doi.org/10.1002/sim.1186.

Higgins, J. P. T., S. G. Thompson, J. J. Deeks, and D. G. Altman. 2003. Measuring inconsistency in meta-analyses. *BMJ* 327: 557–560. https://doi.org/10.1136/bmj.327.7414.557.

Higgins, J. P. T., S. G. Thompson, and D. J. Spiegelhalter. 2009. A re-evaluation of random-effects meta-analysis. *Journal of the Royal Statistical Society*, A ser., 172: 137–159. https://doi.org/10.1111/j.1467-985X.2008.00552.x.

Knapp, G., and J. Hartung. 2003. Improved tests for a random effects meta-regression with a single covariate. *Statistics in Medicine* 22: 2693–2710. https://doi.org/10.1002/sim.1482.

Kontopantelis, E., and D. Reeves. 2016. "metaan: Random-effects meta-analysis". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 55–67. 2nd ed. College Station, TX: Stata Press.

Lau, J., E. M. Antman, J. Jimenez-Silva, B. Kupelnick, F. Mosteller, and T. C. Chalmers. 1992. Cumulative meta-analysis of therapeutic trials for myocardial infarction. *New England Journal of Medicine* 327: 248–254. https://doi.org/10.1056/NEJM199207233270406.

Mantel, N., and W. Haenszel. 1959. Statistical aspects of the analysis of data from retrospective studies of disease. *Journal of the National Cancer Institute* 22: 719–748. Reprinted in *Evolution of Epidemiologic Ideas: Annotated Readings on Concepts and Methods*, ed. S. Greenland, pp. 112–141. Newton Lower Falls, MA: Epidemiology Resources.

Miller, J. J. 1978. The inverse of the Freeman–Tukey double arcsine transformation. *American Statistician* 32: 138. https://doi.org/10.1080/00031305.1978.10479283.

Morris, C. N. 1983. Parametric empirical Bayes inference: Theory and applications. *Journal of the American Statistical Association* 78: 47–55. https://doi.org/10.2307/2287098.

Paule, R. C., and J. Mandel. 1982. Consensus values and weighting factors. *Journal of Research of the National Bureau of Standards* 87: 377–385. https://doi.org/10.6028/jres.087.022.

Petitti, D. B. 2001. Approaches to heterogeneity in meta-analysis. *Statistics in Medicine* 20: 3625–3633. https://doi.org/10.1002/sim.1091.

Peto, R., M. C. Pike, P. Armitage, N. E. Breslow, D. R. Cox, S. V. Howard, N. Mantel, K. McPherson, J. Peto, and P. G. Smith. 1977. Design and analysis of randomized clinical trials requiring prolonged observation of each patient. II. Analysis and examples. *British Journal of Cancer* 35: 1–39. https://doi.org/10.1038/bjc.1977.1.

Raudenbush, S. W. 1984. Magnitude of teacher expectancy effects on pupil IQ as a function of the credibility of expectancy induction: A synthesis of findings from 18 experiments. *Journal of Educational Psychology* 76: 85–97. https://doi.org/10.1037/0022-0663.76.1.85.

———. 2009. "Analyzing effect sizes: Random-effects models". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 295–316. 2nd ed. New York: Russell Sage Foundation.

Raudenbush, S. W., and A. S. Bryk. 1985. Empirical Bayes meta-analysis. *Journal of Educational Statistics* 10: 75–98. https://doi.org/10.2307/1164836.

Rice, K., J. P. T. Higgins, and T. S. Lumley. 2018. A re-evaluation of fixed effect(s) meta-analysis. *Journal of the Royal Statistical Society*, A ser., 181: 205–227. https://doi.org/10.1111/rssa.12275.

Robins, J. M., N. E. Breslow, and S. Greenland. 1986. Estimators of the Mantel–Haenszel variance consistent in both sparse data and large-strata limiting models. *Biometrics* 42: 311–323. https://doi.org/10.2307/2531052.

Robins, J. M., S. Greenland, and N. E. Breslow. 1986. A general estimator for the variance of the Mantel–Haenszel odds ratio. *American Journal of Epidemiology* 124: 719–723. https://doi.org/10.1093/oxfordjournals.aje.a114447.

Rosenthal, R., and D. B. Rubin. 1982. Comparing effect sizes of independent studies. *Psychological Bulletin* 92: 500–504. https://doi.org/10.1037/0033-2909.92.2.500.

Schmidt, F. L., and J. E. Hunter. 2015. *Methods of Meta-Analysis: Correcting Error and Bias in Research Findings*. 3rd ed. Thousand Oaks, CA: Sage. https://doi.org/10.4135/9781483398105.

Schwarzer, G., J. R. Carpenter, and G. Rücker. 2015. *Meta-Analysis with R*. New York: Springer. https://doi.org/10.1007/978-3-319-21416-0.

Sidik, K., and J. N. Jonkman. 2002. A simple confidence interval for meta-analysis. *Statistics in Medicine* 21: 3153–3159. https://doi.org/10.1002/sim.1262.

———. 2003. On constructing confidence intervals for a standardized mean difference in meta-analysis. *Communications in Statistics—Simulation and Computation* 32: 1191–1203. https://doi.org/10.1081/SAC-120023885.

———. 2005. A note on variance estimation in random effects meta-regression. *Journal of Biopharmaceutical Statistics* 15: 823–838. https://doi.org/10.1081/BIP-200067915.

Sterne, J. A. C. 2016. "Cumulative meta-analysis". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 68–77. 2nd ed. College Station, TX: Stata Press.

Sutton, A. J., and J. P. T. Higgins. 2008. Recent developments in meta-analysis. *Statistics in Medicine* 27: 625–650. https://doi.org/10.1002/sim.2934.

Thompson, S. G., and S. J. Sharp. 1999. Explaining heterogeneity in meta-analysis: A comparison of methods. *Statistics in Medicine* 18: 2693–2708. https://doi.org/10.1002/(sici)1097-0258(19991030)18:20<2693::aid-sim235>3.0.co;2-v.

Veroniki, A. A., D. Jackson, W. Viechtbauer, R. Bender, J. Bowden, G. Knapp, O. Kuss, J. P. T. Higgins, D. Langan, and G. Salanti. 2016. Methods to estimate the between-study variance and its uncertainty in meta-analysis. *Research Synthesis Methods* 7: 55–79. https://doi.org/10.1002/jrsm.1164.

Viechtbauer, W., J. A. López-López, J. Sánchez-Meca, and F. Marín-Martínez. 2015. A comparison of procedures to test for moderators in mixed-effects meta-regression models. *Psychological Methods* 20: 360–374. https://doi.org/10.1037/met0000023.

Yusuf, S., R. Peto, J. A. Lewis, R. Collins, and P. Sleight. 1985. Beta blockade during and after myocardial infarction: An overview of the randomized trials. *Progress in Cardiovascular Diseases* 27: 335–371. https://doi.org/10.1016/S0033-0620(85)80003-7.

Also see

[META] **meta data** — Declare meta-analysis data

[META] **meta forestplot** — Forest plots+

[META] **meta galbraithplot** — Galbraith plots

[META] **meta regress** — Meta-analysis regression

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta galbraithplot — Galbraith plots

Description Quick start Menu Syntax
Options Remarks and examples Stored results References
Also see

Description

`meta galbraithplot` produces Galbraith plots for a meta-analysis. These plots are useful for assessing heterogeneity of the studies and for detecting potential outliers. They may also be an alternative to forest plots for summarizing meta-analysis results when there are many studies.

Quick start

Produce a Galbraith plot after data are declared by using either `meta set` or `meta esize`

```
meta galbraithplot
```

Same as above, but request that the slope of the regression line, the standardized effect-sizes, and the study precisions be computed using a random-effects REML method instead of the default common-effect inverse-variance method

```
meta galbraithplot, random(reml)
```

Same as above, but suppress the CI bands

```
meta galbraithplot, random(reml) noci
```

Modify the default styles of the reference and regression lines

```
meta galbraithplot, rlopts(lcolor(red)) lineopts(lpattern(dash))
```

Menu

Statistics > Meta-analysis

Syntax

```
meta galbraithplot [ if ] [ in ] [ , options ]
```

options	Description
Main	
`random[(remethod)]`	random-effects meta-analysis
`common`	common-effect meta-analysis; implies inverse-variance method; the default
`fixed`	fixed-effects meta-analysis; implies inverse-variance method
`[no]regline`	display or suppress the regression line
`[no]ci`	display or suppress the confidence intervals
`level(#)`	set confidence level; default is as declared for meta-analysis
`[no]metashow`	display or suppress meta settings in the output
graph_options	affect rendition of overall Galbraith plot
`[no]lowercase`	lowercase (default) or display as is the first word of the effect-size label used in the *y*-axis title

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

`lowercase` and `nolowercase` do not appear in the dialog box.

remethod	Description
`reml`	restricted maximum likelihood; the default
`mle`	maximum likelihood
`ebayes`	empirical Bayes
`dlaird`	DerSimonian–Laird
`sjonkman`	Sidik–Jonkman
`hedges`	Hedges
`hschmidt`	Hunter–Schmidt

graph_options	Description
RL options	
`rlopts(`*line_options*`)`	affect rendition of the plotted reference line indicating no effect
Fitted line	
`lineopts(`*line_options*`)`	affect rendition of the plotted regression line
CI plot	
`ciopts(`*ciopts*`)`	affect rendition of the plotted CI band
Add plots	
`addplot(`*plot*`)`	add other plots to the Galbraith plot
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than `by()` documented in [G-3] ***twoway_options***

Options

Main

`random()`, `common`, and `fixed` specify a meta-analysis model to use when estimating the slope of the regression line in the Galbraith plot. These options also affect the standard error computation used in the standardization of the effect sizes. For historical reasons, the default is `common` based on the inverse-variance method, regardless of the global model declared by `meta set` or `meta esize`. Specify one of these options with `meta galbraithplot` to override this default.

`random` and `random(`*remethod*`)` specify that a random-effects model be assumed for the construction of the Galbraith plot; see *Random-effects model* in [META] **Intro**.

remethod specifies the type of estimator for the between-study variance τ^2. *remethod* is one of `reml`, `mle`, `ebayes`, `dlaird`, `sjonkman`, `hedges`, or `hschmidt`. `random` is a synonym for `random(reml)`. See *Options* in [META] **meta esize** for more information.

`common` specifies that a common-effect model be assumed for the construction of the Galbraith plot; see *Common-effect ("fixed-effect") model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about common-effect versus fixed-effects models.

`fixed` specifies that a fixed-effects model be assumed for the construction of the Galbraith plot; see *Fixed-effects model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about fixed-effects versus common-effect models. Galbraith plots for the common-effect and fixed-effects models are identical.

`regline` and `noregline` display or suppress the rendition of the regression line. The default, `regline`, is to display the regression line. Option `noregline` implies option `noci`.

`ci` and `noci` display or suppress confidence intervals. The default, `ci`, is to display them.

`level(#)` specifies the confidence level, as a percentage, for confidence intervals. The default is as declared for the meta-analysis session; see *Declaring a confidence level for meta-analysis* in [META] **meta data**. Also see option `level()` in [META] **meta set**.

`metashow` and `nometashow` display or suppress the meta setting information. By default, this information is displayed at the top of the output. You can also specify `nometashow` with `meta update` to suppress the meta setting output for the entire meta-analysis session.

RL options

`rlopts(`*line_options*`)` affects the rendition of the plotted reference (diagonal) line that indicates no effect of the intervention or treatment; see [G-3] ***line_options***.

Fitted line

`lineopts(`*line_options*`)` affects the rendition of the plotted regression line; see [G-3] ***line_options***.

CI plot

`ciopts(`*ciopts*`)` affects the rendition of the CI band in the Galbraith plot. *ciopts* are any options as defined in [G-2] **graph twoway rline** and option `recast(rline)` as described in [G-3] ***advanced_options***.

Add plots

addplot(*plot*) allows adding more graph twoway plots to the graph; see [G-3] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G-3] ***twoway_options***, excluding by(). These include options for titling the graph (see [G-3] ***title_options***) and for saving the graph to disk (see [G-3] ***saving_option***).

The following options are available with meta galbraithplot but are not shown in the dialog box:

lowercase and nolowercase lowercase or leave as is the first word of the effect-size label shown on the y-axis title. The y-axis title is "Standardized *eslabel*", where *eslabel* is the effect-size label defined by meta set or meta esize using the eslabel() option. By default, the command lowercases the first word of *eslabel* to follow Stata's sentence capitalization style. If the first word is a proper name or if you want to use the title capitalization style, you can specify option nolowercase to display *eslabel* as is.

Remarks and examples

The Galbraith plot (Galbraith 1988) is a scatterplot of the standardized effect size (z score) on the y axis against precision (inverse standard error) on the x axis for each study. It is mainly used to assess heterogeneity of the studies and detect potential outliers. It may also be an alternative to forest plots for summarizing meta-analysis results, especially when there are many studies (Anzures-Cabrera and Higgins 2010). The overall effect size is depicted as the slope of the regression line through the origin.

Heterogeneity (and potential outliers) may be investigated by looking at the variation of the studies around the regression line. To aid with that, the Galbraith plot additionally draws a $100(1-\alpha)\%$ confidence region represented by two lines drawn at the $\pm z_{1-\alpha/2}$ intercept values parallel to the regression line. In the absence of heterogeneity, $100(1-\alpha)\%$ of the studies should fall within that region. The plot also contains a reference line at $y = 0$, which indicates "no effect".

meta galbraithplot produces Galbraith plots. The plotted standardized effect size is determined automatically based on the declared effect size. Unlike other meta commands, for historical reasons, meta galbraithplot assumes a common-effect model with the inverse-variance method for the construction of the Galbraith plot.

Under the common-effect and fixed-effects models, the study precisions, $x_j = 1/\hat{\sigma}_j$, and the standardized effect sizes, $y_j = \hat{\theta}_j/\hat{\sigma}_j$, are used to estimate the slope, $\hat{\theta}_{\text{IV}}$, of the regression line through the origin

$$\hat{\theta}_{\text{IV}} = \frac{\sum_{j=1}^{K} x_j y_j}{\sum_{j=1}^{K} x_j^2}$$

If random() is specified, the expressions for x_j and y_j become $x_j = 1/\sqrt{\hat{\sigma}_j^2 + \hat{\tau}^2}$ and $y_j = \hat{\theta}_j/\sqrt{\hat{\sigma}_j^2 + \hat{\tau}^2}$, and the slope of the regression line is now equal to the overall effect size from the random-effects model, $\hat{\theta}^*$.

Two $100(1-\alpha)\%$ CI lines $y = \hat{\theta}x \pm z_{1-\alpha/2}$, which are parallel to the regression line, are added to the plot, where $\hat{\theta}$ is one of $\hat{\theta}_{\text{IV}}$ or $\hat{\theta}^*$ depending on the chosen model.

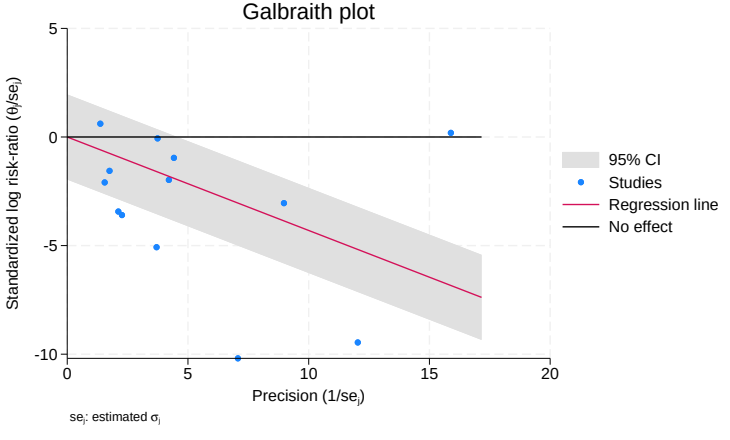
By default, the global CI level specified in `meta set` or `meta esize` is used to compute $z_{1-\alpha/2}$ but a different level may be selected via the `level()` option. The regression line (and consequently its CI bands) may be suppressed via the `noregline` option.

If you wish to only suppress the CI bands, then you may specify the `noci` option. You can also control the look of the plotted reference line, the regression line, and the CI bands by specifying the `rlopts()`, `lineopts()`, and `ciopts()` options, respectively.

▷ Example 1: Basic Galbraith plot

Consider the declared version of the BCG dataset, `bcgset.dta`, which we used in, for instance, example 1 of [META] **meta regress**. Let's produce the Galbraith plot for these data.

```
. use https://www.stata-press.com/data/r18/bcgset
(Efficacy of BCG vaccine against tuberculosis; set with -meta esize-)
. meta galbraithplot
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Common effect
             Method: Inverse-variance
```



The blue circles form a scatterplot of the study-specific standardized log risk-ratios against study precisions. Studies that are close to the y axis have low precision. Precision of studies increases as you move toward the right on the x axis.

The reference black line ($y = 0$) represents the "no-effect" line. That is, the log risks (or risks) in the treatment and control groups for the trials on the line are either the same or very similar. There are two trials that are on the line in our example: one is a large trial, and the other one is a small trial. The log risks for these trials are similar in the two groups, and the corresponding log risk-ratios are close to zero.

If a circle is above the reference line, the risk in the treatment group is higher than the risk in the control group for that study. Conversely, if a circle is below the line, the risk in the treatment group is lower than the risk in the control group. In our example, one trial is above the reference line, suggesting that the risk in the treatment group is higher, but this is an imprecise trial. The remaining trials are below the line, suggesting that the risk is lower in the treatment group.

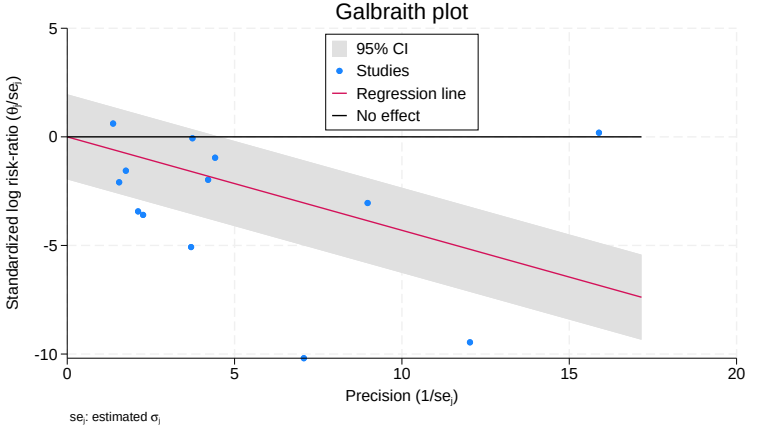
The red line is the regression line through the origin. The slope of this line equals the estimate of the overall effect size, which is the overall log risk-ratio in our example. Also, the slope of an imaginary line from the origin to an individual circle is equal to the effect size (log risk-ratio) estimate corresponding to that circle. This is because the slope is given by $y_j/x_j = (\hat{\theta}_j/\hat{\sigma}_j)/(1/\hat{\sigma}_j) = \hat{\theta}_j$. Thus, studies that fall above the regression line have effect-size estimates larger than the overall effect size, and those falling below the line have estimates that are smaller than the overall effect size.

In the absence of substantial heterogeneity, we expect around 95% of the studies to lie within the 95% CI region (shaded area). In our example, there are 6 trials out of 13 that are outside of the CI region. We should suspect the presence of heterogeneity in these data. In fact, we did establish in example 1 of [META] **meta regress** that there is at least one moderator, the distance from the equator, that explains some of the variation in the trial effect sizes.

Example 2: Custom legend

Continuing with example 1, let's demonstrate how we can customize the look of the legend produced by default. We use `meta update` to suppress the meta setting information displayed by `meta galbraithplot`.

```
. quietly meta update, nometashow
. meta galbraithplot, legend(symxsize(*0.4) position(12) ring(0)
> region(lcolor(black)))
```

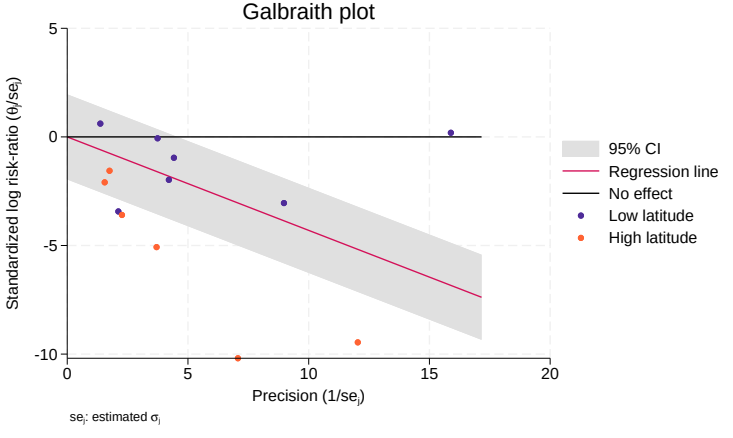


We customized the legend with a few suboptions specified in `legend()` (see [G-3] ***legend_options***). We used `symxsize(*0.4)` to set the width of the key symbols to 40% of their default width. We used `position(12)` to position the label at 12 o'clock and `ring(0)` to place the legend inside the plot region. We used `region(lcolor(black))` to add a black border around the legend region.

Example 3: Labeling trials

Continuing with example 1, we established that there is heterogeneity among the studies given the wide spread of the plotted circles around the regression line. We also know from example 1 of [META] **meta regress** that there is at least one moderator, the distance from the equator, that explains some of the variation in the trial effect sizes. We would like to highlight this fact on the Galbraith plot by creating two study groups corresponding to low (`latitude_c` <0) and high (`latitude_c` ≥ 0) absolute (mean-centered) latitudes and assigning a different color marker for each group on the plot. We will use the `addplot()` option.

```
. generate double precision = 1/_meta_se
. generate double zscore = _meta_es*precision
. local opts legend(order(1 3 4 5 "Low latitude" 6 "High latitude"))
. meta galbraithplot, msymbol(none)
> addplot(scatter zscore precision if latitude_c < 0, `opts' ||
> scatter zscore precision if latitude_c >= 0, `opts')
```



First, we generated two new variables, `precision` and `zscore`, that contain the precisions, $1/\hat{\sigma}_j$, and z scores, $\hat{\theta}_j/\hat{\sigma}_j$, of the studies. Then, we constructed a Galbraith plot without study markers (without the blue circles) using the `msymbol(none)` option. Finally, we used `addplot()` to overlay two scatterplots corresponding to low and high latitudes. The `order()` suboption within `legend()` displays informative legend keys for the added scatterplots in the legend box at the bottom of the plot (see [G-3] ***legend_options***).

All circles in the "high latitude" group (colder climate) fall below the regression line. Thus, the reported risk ratios in colder climates are below the overall risk-ratio estimate, confirming our findings in example 9 of [META] **meta** that the vaccine is more efficient in colder areas. In the "low latitude" group, only one study (study 7: `Vandiviere et al., 1973`) had a risk ratio below the overall risk-ratio estimate. Note that this study was also identified as an outlier in the bubble plot of example 4 of [META] **estat bubbleplot**.

Stored results

`meta galbraithplot` stores the following in `r()`:

Scalars

`r(theta)`	estimated overall effect size
`r(tau2)`	estimated between-study variance (when `random()` is specified)

Macros

`r(model)`	meta-analysis model
`r(method)`	meta-analysis estimation method

References

Anzures-Cabrera, J., and J. P. T. Higgins. 2010. Graphical displays for meta-analysis: An overview with suggestions for practice. *Research Synthesis Methods* 1: 66–80. https://doi.org/10.1002/jrsm.6.

Galbraith, R. F. 1988. A note on graphical representation of estimated odds ratios from several clinical trials. *Statistics in Medicine* 7: 889–894. https://doi.org/10.1002/sim.4780070807.

Also see

[META] **meta data** — Declare meta-analysis data

[META] **meta forestplot** — Forest plots+

[META] **meta labbeplot** — L'Abbé plots

[META] **meta regress** — Meta-analysis regression

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta labbeplot — L'Abbé plots

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
References | Also see

Description

`meta labbeplot` produces L'Abbé plots for a meta-analysis that compares the binary outcomes of two groups. These plots are useful for assessing heterogeneity and comparing study-specific event rates in the two groups.

Quick start

Construct a L'Abbé plot based on the effect size for two-sample binary data computed by `meta esize`

```
meta labbeplot
```

Same as above, but request that the overall effect size be computed using a random-effects REML method instead of the default common-effect inverse-variance method

```
meta labbeplot, random(reml)
```

Same as above, but specify that study-marker sizes be proportional to weights from a random-effects model instead of the default common-effect model

```
meta labbeplot, random(reml) reweighted
```

Modify the default looks of the reference line and the overall effect-size line

```
meta labbeplot, rlopts(lcolor(red)) esopts(lpattern(solid))
```

Menu

Statistics > Meta-analysis

Syntax

```
meta labbeplot [ if ] [ in ] [ , options ]
```

options	Description
Main	
random[(remethod)]	random-effects meta-analysis
common[(cefemethod)]	common-effect meta-analysis
fixed[(cefemethod)]	fixed-effects meta-analysis
reweighted	make bubble size depend on random-effects weights
[no]metashow	display or suppress meta settings in the output
graph_options	affect rendition of overall L'Abbé plot

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

remethod	Description
reml	restricted maximum likelihood; the default
mle	maximum likelihood
ebayes	empirical Bayes
dlaird	DerSimonian–Laird
sjonkman	Sidik–Jonkman
hedges	Hedges
hschmidt	Hunter–Schmidt

cefemethod	Description
mhaenszel	Mantel–Haenszel
invvariance	inverse variance
ivariance	synonym for invvariance

graph_options	Description
RL options	
rlopts(line_options)	affect rendition of the plotted reference line indicating no effect
ES options	
esopts(line_options)	affect rendition of the plotted estimated effect-size line
Add plots	
addplot(plot)	add other plots to the contour plot
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than by() documented in [G-3] ***twoway_options***

Options

Main

Options `random()`, `common()`, and `fixed()` specify a meta-analysis model to use when estimating the overall effect size. For historical reasons, the default is `common(invvariance)`, regardless of the global model declared by `meta esize`. Specify one of these options with `meta labbeplot` to override this default. Options `random()`, `common()`, and `fixed()` may not be combined. Also see *Meta-analysis models* in [META] **Intro**.

`random` and `random(`*remethod*`)` specify that a random-effects model be assumed for meta-analysis; see *Random-effects model* in [META] **Intro**.

remethod specifies the type of estimator for the between-study variance τ^2. *remethod* is one of `reml`, `mle`, `ebayes`, `dlaird`, `sjonkman`, `hedges`, or `hschmidt`. `random` is a synonym for `random(reml)`. See *Options* in [META] **meta esize** for more information.

`common` and `common(`*cefemethod*`)` specify that a common-effect model be assumed for meta-analysis; see *Common-effect ("fixed-effect") model* in [META] **Intro**. Also see the discussion in [META] **meta data** about common-effect versus fixed-effects models.

`common` implies `common(mhaenszel)`.

cefemethod is one of `mhaenszel` or `invvariance` (synonym `ivariance`). See *Options* in [META] **meta esize** for more information.

`fixed` and `fixed(`*cefemethod*`)` specify that a fixed-effects model be assumed for meta-analysis; see *Fixed-effects model* in [META] **Intro**. Also see the discussion in [META] **meta data** about fixed-effects versus common-effect models.

`fixed` implies `fixed(mhaenszel)`.

cefemethod is one of `mhaenszel` or `invvariance` (synonym `ivariance`); see *Options* in [META] **meta esize** for more information.

`reweighted` is used with random-effects meta-analysis. It specifies that the sizes of the bubbles be proportional to the weights from the random-effects meta-analysis, $w_j^* = 1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$. By default, the sizes are proportional to the precision of each study, $w_j = 1/\hat{\sigma}_j^2$.

`metashow` and `nometashow` display or suppress the meta setting information. By default, this information is displayed at the top of the output. You can also specify `nometashow` with `meta update` to suppress the meta setting output for the entire meta-analysis session.

RL options

`rlopts(`*line_options*`)` affects the rendition of the plotted reference (diagonal) line that indicates no effect of the intervention or treatment; see [G-3] ***line_options***.

ES options

`esopts(`*line_options*`)` affects the rendition of the dashed line that plots the estimated overall effect size; see [G-3] ***line_options***.

Add plots

`addplot(`*plot*`)` allows adding more `graph twoway` plots to the graph; see [G-3] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G-3] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G-3] ***title_options***) and for saving the graph to disk (see [G-3] ***saving_option***).

Remarks and examples

The L'Abbé plot (L'Abbé, Detsky, and O'Rourke 1987) is a scatterplot of the summary outcome measure such as log odds in the control group on the x axis and of that in the treatment group on the y axis. This plot is used with two-sample binary data declared by `meta esize`. The plotted summary outcome measure depends on the chosen effect size. It is log odds when the effect size is log odds-ratio, log risk when the effect size is log risk-ratio, and risk when the effect size is risk difference. The summary outcome measures are plotted as circles with their sizes (areas) proportional to study precisions. The plot also contains a reference (diagonal) line, which indicates identical outcomes in the two groups and thus represents no effect, and the estimated overall effect-size line.

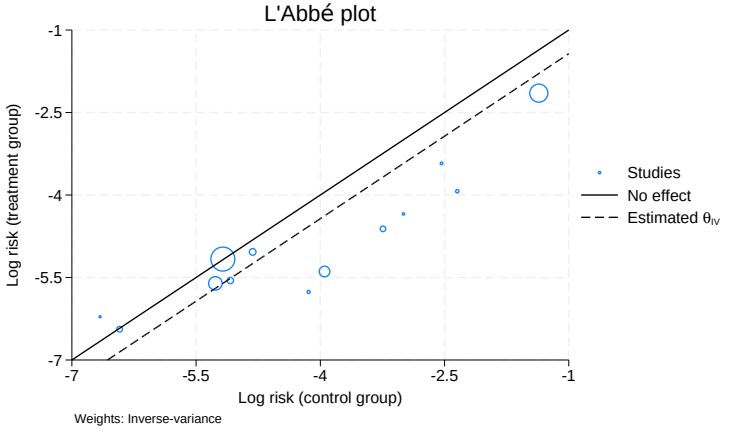
The L'Abbé plot explores between-study heterogeneity by comparing group-level summary outcome measures across studies. It can also be used to determine which type of effect size is more homogeneous across studies. Compared with other meta-analysis graphs, one important advantage of the L'Abbé plot is that it displays the data on individual studies for each of the two groups. Thus, in addition to identifying outlying studies, it can also identify the outlying groups within studies. Also see Anzures-Cabrera and Higgins (2010) for more detail.

`meta labbeplot` produces L'Abbé plots. The plotted summary outcome measure is determined automatically based on the declared effect size. Unlike other `meta` commands, for historical reasons, `meta labbeplot` assumes a common-effect model with the inverse-variance method when computing the overall effect size to be plotted. You can use `random()`, `common()`, or `fixed()` to specify a different meta-analysis model or method. By default, `meta labbeplot` uses the precision weights, $1/\hat{\sigma}_j^2$, but, with a random-effects model, you can instead choose to use the random-effects weights, $1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$. You can also control the look of the plotted reference and effect-size lines by specifying the `rlopts()` and `esopts()` options.

▷ Example 1: Basic L'Abbé plot

Consider the declared version of the BCG dataset, `bcgset.dta`, which we used in, for instance, example 1 of [META] **meta regress**. Let's produce the L'Abbé plot for these data.

```
. use https://www.stata-press.com/data/r18/bcgset
(Efficacy of BCG vaccine against tuberculosis; set with -meta esize-)
. meta labbeplot
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
      Summary data: npost nnegt nposc nnegc
              Model: Common effect
             Method: Inverse-variance
```



From the displayed meta settings, the declared effect size is a log risk-ratio. Thus, `meta labbeplot` plots the log risks on the scatterplot. The treatment-group log risk is on the y axis, and the control-group log risk is on the x axis. The sizes of the plotted markers (circles) are proportional to the precision of the trials. Large circles represent more precise, larger trials, whereas small circles represent less precise, smaller trials.

The solid reference line ($y = x$) represents the "no-effect" line. That is, the log risks (or risks) in the two groups for the trials on the line are either the same or very similar. There are two trials that are on the line in our example: one is a large trial, the other one is a small trial. The log risks for these trials are very similar in the two groups, and the corresponding log risk-ratios are close to zero.

If a circle is above the reference line, the risk in the treatment group is higher than the risk in the control group for that study. Conversely, if a circle is below the line, the risk in the treatment group is lower than the risk in the control group. In our example, one trial is above the reference line, suggesting that the risk in the treatment group is higher, but this is a very small trial. The remaining trials are below the line, suggesting that the risk is lower in the treatment group. However, the trials demonstrating large differences between the groups are also smaller (less precise) trials.

The dashed line is the overall effect-size line. The intercept of this line equals the estimate of the overall effect size, which is the overall log risk-ratio in our example. The actual estimate of the overall effect size is not important in the L'Abbé plot. What is important is whether the circles follow the effect-size line or deviate from it. When the circles deviate from the effect-size line greatly, this may be a sign of study heterogeneity. In our example, there are at least five trials that are far away from the effect-size

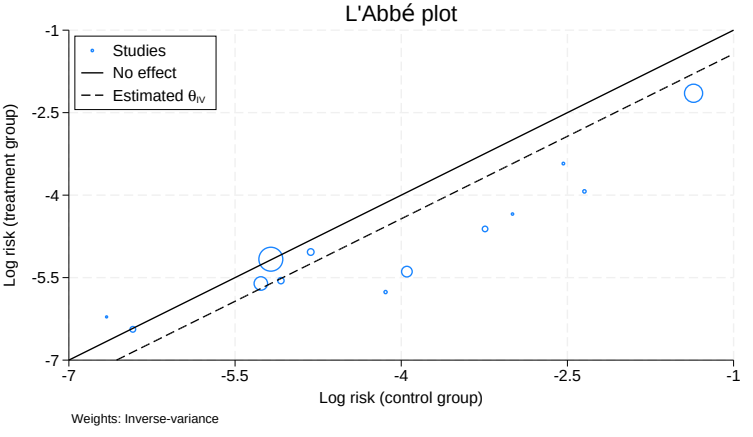
line. We should suspect the presence of heterogeneity in these data. In fact, we did establish in example 1 of [META] **meta regress** that there is at least one moderator, the distance from the equator, that explains some of the variation in the trial effect sizes.

◁

▷ Example 2: Custom legend

Continuing with example 1, let's demonstrate how we can customize the look of the legend produced by default.

```
. meta labbeplot, legend(symxsize(*0.6) position(10) ring(0)
> region(lcolor(black)))
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
      Summary data: npost nnegt nposc nnegc
              Model: Common effect
             Method: Inverse-variance
```



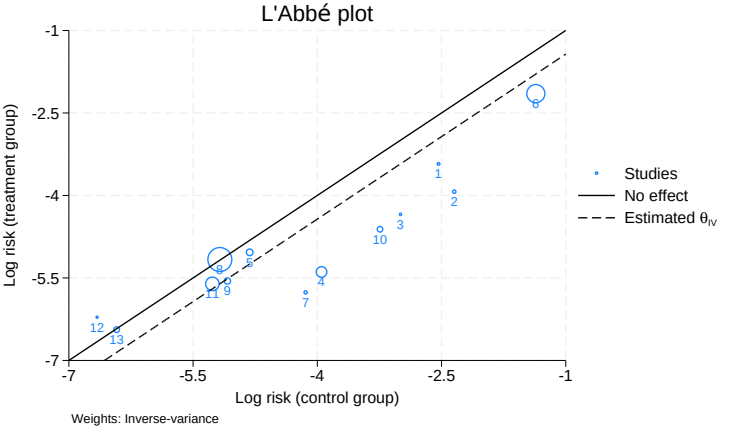
We customized the legend with a few suboptions specified in `legend()`. We used `symxsize(*0.6)` to set the width of the key symbols to 60% of their default width. We used `position(10)` to position the label at 10 o'clock and `ring(0)` to place the legend inside the plot region. We used `region(lcolor(black))` to add a black border around the legend region.

◁

▷ Example 3: Labeling trials

Continuing with example 1, let's say it would be nice to mark the circles with the trial labels. We use the `addplot()` option and follow similar steps to those described in example 3 of [META] **estat bubbleplot**, except here we generate new variables for the added scatterplot.

```
. generate double lnriskt = ln(npost/(npost + nnegt))
. generate double lnriskc = ln(nposc/(nposc + nnegc))
. local opts msymbol(none) mlabel(trial) mlabpos(6) mlabcolor(stblue)
. meta labbeplot, addplot(scatter lnriskt lnriskc, 'opts' legend(order(1 2 3)))
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
      Summary data: npost nnegt nposc nnegc
              Model: Common effect
             Method: Inverse-variance
```



First, we generated two new variables, `lnriskt` and `lnriskc`, that contain the log risks in the treatment and control groups. Then, we used `addplot()` to overlay the same scatterplot as produced by `meta labbeplot` but without the markers and with marker labels. We specified other options to improve the look of the graph; see example 3 of [META] **estat bubbleplot** for details. Also see example 4 of [META] **estat bubbleplot** for how to further improve the positioning of the labels.

◁

Stored results

`meta labbeplot` stores the following in `r()`:

Scalars

`r(theta)`	estimated overall effect size
`r(xmin)`	minimum value in the control group (x axis)
`r(xmax)`	maximum value in the control group
`r(ymin)`	minimum value in the treatment group (y axis)
`r(ymax)`	maximum value in the treatment group

Macros

`r(model)`	meta-analysis model
`r(method)`	meta-analysis estimation method

Methods and formulas

Let a_j, b_j, c_j, and d_j define cell counts of a 2×2 table for study j; see *Effect sizes for two-group comparison of binary outcomes* in [META] **meta esize**. Let y_j and x_j be the summary measures such as log odds for study j in the treatment and control groups. The L'Abbé plot produces a scatterplot of (y_j, x_j) with the sizes of markers (areas of circles) proportional to the weights $w_j = 1/\hat{\sigma}_j^2$ or, if `reweighted` is specified with a random-effects model, $w_j = 1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$.

When the effect size is risk difference, y_j and x_j are the risks given by

$$y_j = \frac{a_j}{a_j + b_j} \quad \text{and} \quad x_j = \frac{c_j}{c_j + d_j}$$

When the effect size is log risk-ratio, y_j and x_j are the log risks given by

$$y_j = \log\left(\frac{a_j}{a_j + b_j}\right) \quad \text{and} \quad x_j = \log\left(\frac{c_j}{c_j + d_j}\right)$$

When the effect size is log odds-ratio, y_j and x_j are the log odds given by

$$y_j = \log\left(\frac{a_j}{b_j}\right) \quad \text{and} \quad x_j = \log\left(\frac{c_j}{d_j}\right)$$

The plotted reference line is the diagonal line. Studies that have the same values of the summary outcome measures in the two groups will have $y_j = x_j$ and thus will fall on the reference line.

The effect-size (dashed) line is a 45-degree line with an intercept equal to the estimated overall effect size. By default, the overall effect-size is estimated assuming a common-effect model with the inverse-variance method, but this can be changed by specifying one of `random()`, `common()`, or `fixed()`.

References

Anzures-Cabrera, J., and J. P. T. Higgins. 2010. Graphical displays for meta-analysis: An overview with suggestions for practice. *Research Synthesis Methods* 1: 66–80. https://doi.org/10.1002/jrsm.6.

L'Abbé, K. A., A. S. Detsky, and K. O'Rourke. 1987. Meta-analysis in clinical research. *Annals of Internal Medicine Journal* 107: 224–233. https://doi.org/10.7326/0003-4819-107-2-224.

Also see

[META] **meta data** — Declare meta-analysis data

[META] **meta esize** — Compute effect sizes and declare meta-analysis data+

[META] **meta forestplot** — Forest plots+

[META] **meta galbraithplot** — Galbraith plots

[META] **meta regress** — Meta-analysis regression

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta regress — Meta-analysis regression

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
References | Also see

Description

`meta regress` performs meta-analysis regression, or meta-regression, which is a linear regression of the study effect sizes on study-level covariates (moderators). Meta-regression investigates whether between-study heterogeneity can be explained by one or more moderators. You can think of meta-regression as a standard meta-analysis that incorporates moderators into the model. `meta regress` performs both random-effects and fixed-effects meta-regression.

Quick start

Perform meta-regression of the effect size, `_meta_es`, on covariate (moderator) `x1`

```
meta regress x1
```

Same as above, but assume a DerSimonian–Laird random-effects method instead of the method declared by either `meta set` or `meta esize`

```
meta regress x1, random(dlaird)
```

Add a factor variable `a`, and request a Knapp–Hartung adjustment to the standard errors of coefficients

```
meta regress x1 i.a, random(dlaird) se(khartung)
```

Perform a sensitivity analysis by assuming a fixed value of 0.2 for the between-study variance τ^2

```
meta regress x1 i.a, tau2(0.2)
```

Menu

Statistics > Meta-analysis

Syntax

Meta-regression using meta-analysis model as declared with meta set or meta esize

```
meta regress moderators [if] [in] [, reopts options]
```

Random-effects meta-regression

```
meta regress moderators [if] [in], random[(remethod)] [reopts options]
```

Fixed-effects meta-regression

```
meta regress moderators [if] [in], fixed [multiplicative options]
```

Constant-only meta-regression

```
meta regress _cons [if] [in] [, modelopts]
```

reopts	Description
`tau2(#)`	sensitivity meta-analysis using a fixed value of between-study variance τ^2
`i2(#)`	sensitivity meta-analysis using a fixed value of heterogeneity statistic I^2_{res}
`se(`*seadj*`)`	adjust standard errors of the coefficients

options	Description
Model	
`noconstant`	suppress constant term
`tdistribution`	report t tests instead of z tests for the coefficients
Reporting	
`level(#)`	set confidence level; default is as declared for meta-analysis
`noheader`	suppress output header
`[no]metashow`	display or suppress meta settings in the output
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Maximization	
maximize_options	control the maximization process; seldom used
`coeflegend`	display legend instead of statistics

moderators may contain factor variables; see **[U] 11.4.3 Factor variables**.

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

`coeflegend` does not appear in the dialog box.

See **[U] 20 Estimation and postestimation commands** for more capabilities of estimation commands.

remethod	Description
reml	restricted maximum likelihood; the default
mle	maximum likelihood
ebayes	empirical Bayes
dlaird	DerSimonian–Laird
sjonkman	Sidik–Jonkman
hedges	Hedges
hschmidt	Hunter–Schmidt

modelopts is any option except noconstant.

Options

Model

noconstant; see [R] **Estimation options**. This option is not allowed with constant-only meta-regression.

Options random() and fixed, when specified with meta regress, temporarily override the global model declared by meta set or meta esize during the computation. Options random(), common, and fixed may not be combined. If these options are omitted, the declared meta-analysis model is assumed; see *Declaring a meta-analysis model* in [META] **meta data**. Also see *Meta-analysis models* in [META] **Intro**.

random and random(*remethod*) specify that a random-effects model be assumed for meta-regression; see *Random-effects model* in [META] **Intro**.

remethod specifies the type of estimator for the between-study variance τ^2. *remethod* is one of reml, mle, ebayes, dlaird, sjonkman, hedges, or hschmidt. random is a synonym for random(reml). See *Options* in [META] **meta esize** for more information.

fixed specifies that a fixed-effects model be assumed for meta-regression; see *Fixed-effects model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**.

reopts are tau2(#), i2(#), and se(khartung[, truncated]). These options are used with random-effects meta-regression.

tau2(#) specifies the value of the between-study variance parameter, τ^2, to use for the random-effects meta-regression. This option is useful for exploring the sensitivity of the results to different levels of between-study heterogeneity. Only one of tau2() or i2() may be specified.

i2(#) specifies the value of the residual heterogeneity statistic I^2_{res} (as a percentage) to use for the random-effects meta-regression. This option is useful for exploring the sensitivity of the results to different levels of between-study heterogeneity. Only one of i2() or tau2() may be specified.

`se(`*seadj*`)` specifies that the adjustment *seadj* be applied to the standard errors of the coefficients. Additionally, the tests of significance of the coefficients are based on a Student's t distribution instead of the normal distribution.

seadj is `khartung`[`, truncated`]. Adjustment `khartung` specifies that the Knapp–Hartung adjustment (Hartung and Knapp 2001a, 2001b; Knapp and Hartung 2003), also known as the Sidik–Jonkman adjustment (Sidik and Jonkman 2002), be applied to the standard errors of the coefficients. `hknapp` and `sjonkman` are synonyms for `khartung`. `truncated` specifies that the truncated Knapp–Hartung adjustment (Knapp and Hartung 2003), also known as the modified Knapp–Hartung adjustment, be used.

`multiplicative` performs a fixed-effects meta-regression that accounts for residual heterogeneity by including a multiplicative variance parameter ϕ. ϕ is referred to as an "(over)dispersion parameter". See *Introduction* for details.

`tdistribution` reports t tests instead of z tests for the coefficients. This option is useful, for instance, when `meta regress` is used to conduct a regression-based test for funnel-plot asymmetry. Traditionally, the test statistic from this test is compared with critical values from a Student's t distribution instead of the default normal distribution. This option may not be combined with option `se()`.

Reporting

`level(#)` specifies the confidence level, as a percentage, for confidence intervals. The default is as declared for the meta-analysis session; see *Declaring a confidence level for meta-analysis* in [META] **meta data**. Also see option `level()` in [META] **meta set**.

`noheader` suppresses the output header, either at estimation or upon replay.

`metashow` and `nometashow` display or suppress the meta setting information. By default, this information is displayed at the top of the output. You can also specify `nometashow` with `meta update` to suppress the meta setting output for the entire meta-analysis session.

display_options: `noci`, `nopvalues`, `noomitted`, `vsquish`, `noemptycells`, `baselevels`, `allbaselevels`, `nofvlabel`, `fvwrap(#)`, `fvwrapon(`*style*`)`, `cformat(%`*fmt*`)`, `pformat(%`*fmt*`)`, `sformat(%`*fmt*`)`, and `nolstretch`; see [R] **Estimation options**.

Maximization

maximize_options: `iterate(#)`, `tolerance(#)`, `nrtolerance(#)`, `nonrtolerance` (see [R] **Maximize**), `from(#)`, and `showtrace`. These options control the iterative estimation of the between-study variance parameter, τ^2, with random-effects methods `reml`, `mle`, and `ebayes`. These options are seldom used.

`from(#)` specifies the initial value for τ^2 during estimation. By default, the initial value for τ^2 is the noniterative Hedges estimator.

`showtrace` displays the iteration log that contains the estimated parameter τ^2, its relative difference with the value from the previous iteration, and the scaled gradient.

The following option is available with `meta regress` but is not shown in the dialog box:

`coeflegend`; see [R] **Estimation options**.

Remarks and examples

Remarks are presented under the following headings:

Introduction
Examples of using meta regress

Introduction

Meta-regression is a regression performed in the context of meta-analysis. It is used to study the relationship between study effect sizes and covariates. Meta-regression is analogous to standard regression used when individual data are available, but in meta-regression, the observations are the studies, the outcome of interest is the effect size, and the covariates are recorded at the study level. The study-level covariates in meta-regression are known as moderators. Several examples of moderators include study location, study test environment, drug administration method. For a general overview and discussions about meta-regression, see Berlin and Antman (1992), Berkey et al. (1995), and Thompson and Higgins (2002).

The goal of meta-regression is to explore and explain the between-study heterogeneity as a function of moderators. Two types of regression models, fixed-effects (FE) and random-effects (RE), are available. An FE meta-regression assumes that all heterogeneity between study effect sizes can be accounted for by the included moderators. An RE meta-regression accounts for potential additional variability unexplained by the included moderators, also known as residual heterogeneity. Because a common-effect meta-analysis model implies no study heterogeneity, it is not applicable to meta-regression, except in a less interesting case of a constant-only model, which is equivalent to the standard common-effect meta-analysis; see [META] **meta summarize**.

`meta regress` fits meta-regression. Use the `random()` option to fit an RE meta-regression and the `fixed` option to fit an FE meta-regression. Also see *Default meta-analysis model and method* in [META] **meta data** to learn about the default regression model used by `meta regress`.

For the jth study, let $\hat{\theta}_j$ denote the effect size, $\hat{\sigma}^2_j$ its variance, and $\mathbf{x}_j$ be a $1 \times p$ vector of moderators with the corresponding unknown $p \times 1$ coefficient vector $\boldsymbol{\beta}$.

An FE meta-regression (Greenland 1987) is given by

$$\hat{\theta}_j = \mathbf{x}_j\boldsymbol{\beta} + \epsilon_j, \text{ weighted by } w_j = \frac{1}{\hat{\sigma}^2_j}, \text{ where } \epsilon_j \sim N\left(0, \hat{\sigma}^2_j\right)$$

Residual heterogeneity may be incorporated into an FE meta-regression via a multiplicative factor, ϕ, applied to each of the variances $\hat{\sigma}^2_j$. This leads to a multiplicative meta-regression or FE meta-regression with multiplicative dispersion parameter (Thompson and Sharp 1999)

$$\hat{\theta}_j = \mathbf{x}_j\boldsymbol{\beta} + \epsilon^{\phi}_j, \text{ weighted by } w_j = \frac{1}{\hat{\sigma}^2_j}, \text{ where } \epsilon^{\phi}_j \sim N\left(0, \hat{\sigma}^2_j\phi\right)$$

This regression model may be specified by the combination of `fixed` and `multiplicative` options.

Another method of incorporating residual heterogeneity is to include an additive between-study variance component, τ^2, that leads to an RE meta-regression (Berkey et al. 1995), also known as a mixed model in the meta-analysis literature:

$$\hat{\theta}_j = \mathbf{x}_j\boldsymbol{\beta} + \epsilon^*_j = \mathbf{x}_j\boldsymbol{\beta} + u_j + \epsilon_j, \text{ weighted by } w^*_j = \frac{1}{\hat{\sigma}^2_j + \hat{\tau}^2}, \text{ where } \epsilon^*_j \sim N\left(0, \hat{\sigma}^2_j + \tau^2\right)$$

As we mentioned earlier, an RE meta-regression assumes that the moderators explain only part of the heterogeneity, and a random-effects term $u_j \sim N(0, \tau^2)$ is used to account for the remainder.

Harbord and Higgins (2016) point out that some authors (Thompson and Sharp 1999; Higgins and Thompson 2004) argue that an FE meta-regression should not be used because, in practice, the included moderators rarely capture all the between-study heterogeneity and that the failure of the FE regression to capture the extra between-study heterogeneity can lead to excessive type I errors. Also, the results from an FE meta-regression, including its multiplicative version, may not be generalized to populations from which the observed studies are a sample (Konstantopoulos and Hedges 2009). If you do not specify a meta-analysis model with `meta set` or `meta esize` during declaration, an RE meta-regression will be assumed by `meta regress`.

Meta-regression can also be considered an extension of subgroup analysis (see `meta summarize, subgroup()` in [META] **meta summarize**) to include continuous moderators in addition to the categorical ones. In particular, an FE meta-regression with the subgroup variable specified as a factor variable (see **[U] 11.4.3 Factor variables**) is equivalent to the FE subgroup analysis on that variable.

It is recommended that you have at least 10 studies per moderator to perform meta-regression (Borenstein et al. 2009 , chap. 20). Otherwise, you may not be able to estimate the effects of moderators reliably. For more recommendations regarding meta-regression, see Schmidt and Hunter (2015, chap. 9), Deeks, Macaskill, and Irwig (2005), Harbord and Higgins (2016), Sharp (2016), and Thompson and Higgins (2002).

Examples of using meta regress

Consider a dataset from Colditz et al. (1994) of clinical trials that explore the efficacy of a Bacillus Calmette-Guérin (BCG) vaccine in the prevention of tuberculosis (TB). This dataset was introduced in *Efficacy of BCG vaccine against tuberculosis (bcg.dta)* of [META] **meta**. In this section, we use its declared version and focus on the demonstration of various options of `meta regress` and explanation of its output.

```
. use https://www.stata-press.com/data/r18/bcgset
(Efficacy of BCG vaccine against tuberculosis; set with -meta esize-)
. meta query, short
-> meta esize npost - nnegc, esize(lnrratio) studylabel(studylbl)
  Effect-size label: Log risk-ratio
   Effect-size type: lnrratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Random effects
             Method: REML
```

`meta query, short` reminds us about the main settings of the declaration step. Our data were declared by using `meta esize` with variables `npost`, `nnegt`, `nposc`, and `nnegc` representing the summary data from 2×2 tables, which record the numbers of positive and negative TB cases in the treatment and control groups. The computed effect sizes are log risk-ratios; their values and standard errors are stored in the respective system variables `_meta_es` and `_meta_se`. The `studylbl` variable supplies the study labels to be used in the output. The declared meta-analysis model is the default random-effects model with the REML estimation method.

Examples are presented under the following headings:

Example 1: Random-effects meta-regression
Example 2: Sidik–Jonkman random-effects method
Example 3: Truncated Knapp–Hartung standard-error adjustment
Example 4: Sensitivity meta-analysis
Example 5: Fixed-effects meta-regression
Example 6: Multiplicative meta-regression
Example 7: Constant-only model

Example 1: Random-effects meta-regression

In example 9 of [META] **meta**, following Berkey et al. (1995), we fit a meta-regression with a centered absolute latitude, `latitude_c`, as the moderator to address heterogeneity. Let's refit this model here and focus on the specification and output from `meta regress`.

```
. meta regress latitude_c
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Random-effects meta-regression                  Number of obs  =        13
Method: REML                                    Residual heterogeneity:
                                                          tau2 =   .07635
                                                       I2 (%) =    68.39
                                                           H2 =     3.16
                                              R-squared (%) =    75.63
                                                Wald chi2(1)   =     16.36
                                                Prob > chi2    =    0.0001
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
  latitude_c |  -.0291017   .0071953    -4.04   0.000    -.0432043   -.0149991
       _cons |  -.7223204   .1076535    -6.71   0.000    -.9333174   -.5113234
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(11) = 30.73   Prob > Q_res = 0.0012
```

Unlike with many Stata regression commands, we do not specify the dependent variable with `meta regress`. The command includes it automatically from the declared meta settings. `meta regress` provides a short summary of the settings, which you can suppress with the `nometashow` option. System variable `_meta_es` contains the effect sizes and is thus used as the dependent variable. System variable `_meta_se` contains effect-size standard errors; it is used to construct the weights for the regression.

The header includes the information about the meta-analysis model and reports various summaries such as heterogeneity statistics and the model test. For example, the results are based on 13 studies. The reported I^2_{res} statistic is 68%, which still suggests moderate heterogeneity, using the categorization of Higgins et al. (2003), even after including `latitude_c` as the moderator. In other words, 68% of the variability in the residuals is still attributed to the between-study variation, whereas only 32% is attributed to the within-study variation. The adjusted R^2 statistic can be used to assess the proportion of between-study variance explained by the covariates; see (6) in *Methods and formulas* for its definition used in the meta-analysis literature. Here roughly 76% of the between-study variance is explained by the covariate `latitude_c`.

The output header also displays a model test that all coefficients other than the intercept are equal to zero based on the χ^2 distribution with $p - 1$ degrees of freedom. In our example, the χ^2 test statistic is 16.36 with a p-value of 0.0001. We have only one moderator, so the results of the model test in our example are equivalent to the z test (χ^2 value equals squared z value) of the coefficient of `latitude_c` reported in the output table.

The regression coefficient for `latitude_c` is -0.029, which means that every one degree of latitude corresponds to a decrease of 0.0291 units in log risk-ratio. The intercept, $\hat{\beta}_0$, is -0.722, which means that the overall risk ratio at the mean latitude (`latitude_c` $= 0$ corresponds to `latitude` ≈ 33.46) is $\exp(-0.722) = 0.46$. Both of these coefficients are statistically significantly different from zero based on the reported z tests.

Finally, a test of residual homogeneity is reported at the bottom of the output. The test statistic Q_{res} is 30.73 with a p-value of 0.0012, which suggests the presence of heterogeneity among the residuals.

◁

❑ Technical note

Heterogeneity statistics I^2_{res} and H^2_{res}, reported under `Residual heterogeneity:` in the header, are extensions of the corresponding statistics I^2 and H^2 from standard meta-analysis to meta-regression (Higgins and Thompson 2002). They measure the remaining between-study heterogeneity among the residuals after adjusting for the variability due to moderators. Similarly, the test of residual homogeneity based on the Q_{res} statistic is the extension of the standard meta-analysis homogeneity test based on the Cochran's Q statistic to meta-regression. See *Residual heterogeneity measures* and *Residual homogeneity test* in *Methods and formulas*.

❑

▷ Example 2: Sidik–Jonkman random-effects method

Continuing with example 1, let's demonstrate the use of a different RE method, for instance, the Sidik–Jonkman method, instead of the default REML method.

```
. meta regress latitude_c, random(sjonkman)
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se

Random-effects meta-regression                  Number of obs  =        13
Method: Sidik-Jonkman                           Residual heterogeneity:
                                                           tau2 =    .2318
                                                         I2 (%) =   86.79
                                                            H2 =    7.57
                                                R-squared (%) =   32.90
                                                Wald chi2(1)  =    6.50
                                                Prob > chi2   =  0.0108
------------------------------------------------------------------------------
     _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
--------------+---------------------------------------------------------------
  latitude_c |  -.0280714   .0110142    -2.55   0.011    -.0496589   -.0064838
       _cons |  -.7410395   .1602117    -4.63   0.000    -1.055049   -.4270304
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(11) = 30.73   Prob > Q_res = 0.0012
```

The estimate of the regression coefficient for `latitude_c` is -0.028 and is similar to the REML estimate of -0.029, but the standard errors are quite different: 0.011 versus 0.007. Recall that REML assumes that the error distribution is normal, whereas the Sidik–Jonkman estimator does not. Thus, its standard error estimates are likely to be larger than those from REML. The estimates of the between-study variance, τ^2, are also very different: 0.23 compared with the REML estimate of 0.08.

◁

▷ Example 3: Truncated Knapp–Hartung standard-error adjustment

Continuing with example 1, let's use an alternative standard-error computation sometimes used in practice—the truncated Knapp–Hartung method.

```
. meta regress latitude_c, se(khartung, truncated)
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Random-effects meta-regression                  Number of obs  =        13
Method: REML                                    Residual heterogeneity:
SE adjustment: Truncated Knapp-Hartung                    tau2 =   .07635
                                                        I2 (%) =   68.39
                                                            H2 =    3.16
                                                R-squared (%) =   75.63
                                                Model F(1,11)  =   12.59
                                                Prob > F       =  0.0046
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      t    P>|t|     [95% conf. interval]
-------------+----------------------------------------------------------------
  latitude_c |  -.0291017   .0082014    -3.55   0.005    -.0471529   -.0110505
       _cons |  -.7223204   .1227061    -5.89   0.000    -.9923946   -.4522462
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(11) = 30.73   Prob > Q_res = 0.0012
```

The reported standard errors are larger than those from example 1. This is expected because the Knapp–Hartung adjustment incorporates the uncertainty in estimating τ^2 in the standard error computation. Also, the inferences for the tests of coefficients and the model test are now based on the Student's t and F distributions, respectively, instead of the default normal and χ^2 distributions.

◁

▷ Example 4: Sensitivity meta-analysis

We can perform sensitivity analysis to explore the impact of the various levels of heterogeneity on the regression results. Continuing with example 1, let's fit a meta-regression assuming that the residual heterogeneity statistic I^2_{res} equals 90%.

```
. meta regress latitude_c, i2(90)
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Random-effects meta-regression                  Number of obs  =        13
Method: User-specified I2                       Residual heterogeneity:
                                                          tau2 =    .3176
                                                        I2 (%) =   90.00
                                                            H2 =   10.00
                                                Wald chi2(1)   =    4.89
                                                Prob > chi2    =  0.0269
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
  latitude_c |  -.0277589   .0125474    -2.21   0.027    -.0523514   -.0031664
       _cons |  -.7443082   .1812664    -4.11   0.000    -1.099584   -.3890326
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(11) = 30.73   Prob > Q_res = 0.0012
```

The estimate of the coefficient for `latitude_c` is now -0.028 with a standard error estimate of 0.01.

Let's now fit a meta-regression assuming the between-study variance of 0.01.

```
. meta regress latitude_c, tau2(0.01)
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Random-effects meta-regression                  Number of obs  =        13
Method: User-specified tau2                     Residual heterogeneity:
                                                          tau2 =       .01
                                                       I2 (%) =     22.08
                                                           H2 =      1.28
                                                Wald chi2(1)   =     57.62
                                                Prob > chi2    =    0.0000
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
  latitude_c |  -.0295601   .0038942    -7.59   0.000    -.0371926   -.0219277
       _cons |  -.6767043   .0617892   -10.95   0.000    -.7978089   -.5555998
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(11) = 30.73   Prob > Q_res = 0.0012
```

The specified value of τ^2 corresponds to the I^2_{res} value of 22.08%. The coefficient estimate is now -0.03 with a standard error of 0.004.

In both sensitivity analyses, `latitude_c` remained a statistically significant moderator for the log risk-ratios.

◁

▷ Example 5: Fixed-effects meta-regression

Instead of an RE meta-regression as in example 1, we can use the `fixed` option to fit an FE meta-regression. The use of an FE meta-regression is usually discouraged in the meta-analysis literature because it assumes that all between-study heterogeneity is accounted for by the specified moderators (Harbord and Higgins 2016; Thompson and Sharp 1999; Higgins and Thompson 2004). This is often an unrealistic assumption in meta-analysis. We fit this model in our example for the purpose of demonstration.

```
. meta regress latitude_c, fixed
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Fixed-effects meta-regression                   Number of obs  =        13
Method: Inverse-variance                        Wald chi2(1)   =    121.50
                                                Prob > chi2    =    0.0000
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
  latitude_c |  -.0292369   .0026524   -11.02   0.000    -.0344356   -.0240383
       _cons |  -.6347482   .0445446   -14.25   0.000    -.7220541   -.5474423
------------------------------------------------------------------------------
```

Because the FE regression assumes no additional residual heterogeneity, the residual heterogeneity statistics and the residual homogeneity test are not reported with `meta regress, fixed`.

The coefficient estimates are similar to those from example 1, but standard errors from the FE regression are smaller. This is because the FE regression does not account for the residual heterogeneity that is not explained by the included moderators.

Considering the presence of residual heterogeneity in these data, we should go back to our RE analysis or explore the multiplicative meta-regression, which we demonstrate in example 6.

◁

▷ Example 6: Multiplicative meta-regression

An FE meta-regression in example 5 does not account for residual heterogeneity. An extension of this regression model that does, known as a multiplicative meta-regression (see *Introduction* and *Methods and formulas*), has been considered in the meta-analysis literature. An RE meta-regression is the preferred analysis these days, but we provide the multiplicative meta-regression for completeness.

Continuing with example 5, we add the `multiplicative` option to fit an FE meta-regression with a multiplicative dispersion parameter ϕ.

```
. meta regress latitude_c, fixed multiplicative
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Fixed-effects meta-regression                   Number of obs  =        13
Error: Multiplicative                           Dispersion phi =      2.79
Method: Inverse-variance                        Wald chi2(1)   =     43.49
                                                Prob > chi2    =    0.0000
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
  latitude_c |  -.0292369   .0044335    -6.59   0.000    -.0379265   -.0205474
       _cons |  -.6347482   .0744564    -8.53   0.000    -.7806801   -.4888163
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(11) = 30.73   Prob > Q_res = 0.0012
```

The estimate of the dispersion parameter, reported in the header as `Dispersion phi`, is 2.79. It is greater than 1, which suggests the presence of residual heterogeneity in these data. The coefficient estimates are the same as those in example 5, but the standard errors are about two times larger.

◁

▷ Example 7: Constant-only model

The primary use of `meta regress` is to fit meta-regression models containing moderators. You can also fit a constant-only model (without moderators), although this is less common in the context of meta-regression.

To fit a constant-only model with many regression estimation commands, you simply omit the covariates in the command specification. This would not work with `meta regress` because, without the dependent-variable specification, we would have to type

```
. meta regress
```

which means replaying previous estimation results consistently across Stata. The above will either issue an error that previous estimation results are not found or redisplay the results from the previous `meta regress` specification.

Instead, to fit a constant-only model with `meta regress`, you specify the designator `_cons` following the command name.

```
. meta regress _cons
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Random-effects meta-regression                  Number of obs  =        13
Method: REML                                    Residual heterogeneity:
                                                           tau2 =    .3132
                                                      I2 (%) =    92.22
                                                          H2 =    12.86
                                                Wald chi2(0)   =         .
                                                Prob > chi2    =         .
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       _cons |  -.7145323   .1797815    -3.97   0.000    -1.066898    -.362167
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(12) = 152.23  Prob > Q_res = 0.0000
```

Note that the estimated value of $\hat{\tau}^2$ is now 0.313, whereas in example 1 it was 0.076. That is, the inclusion of covariate `latitude_c` in example 1 reduced $\hat{\tau}^2$ from 0.313 to 0.076 for a relative reduction of $(0.313 - 0.076)/0.313 \approx 76\%$.

The reason a constant-only meta-regression is not as common is because it produces the same results as a standard meta-analysis.

```
. meta summarize, nostudies
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
        Study label: studylbl
Meta-analysis summary                           Number of studies =     13
Random-effects model                            Heterogeneity:
Method: REML                                               tau2 =  0.3132
                                                      I2 (%) =    92.22
                                                          H2 =    12.86

        theta: Overall Log risk-ratio
------------------------------------------------------------------------------
             |    Estimate    Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       theta |   -.7145323    .1797815    -3.97   0.000    -1.066898    -.362167
------------------------------------------------------------------------------
Test of homogeneity: Q = chi2(12) = 152.23                Prob > Q = 0.0000
```

See [META] **meta summarize** for details.

◁

Stored results

meta regress stores the following in e():

Scalars

e(N)	number of observations (studies)
e(df_m)	model degrees of freedom
e(df_r)	residual degrees of freedom
e(chi2)	model χ^2 Wald test statistic
e(F)	model F statistic
e(p)	p-value for model test
e(phi)	dispersion parameter
e(tau2)	between-study variance
e(I2_res)	I^2_{res} heterogeneity statistic
e(H2_res)	H^2_{res} heterogeneity statistic
e(R2)	R^2 heterogeneity measure
e(Q_res)	Cochran's Q residual homogeneity test statistic
e(df_Q_res)	degrees of freedom for residual homogeneity test
e(p_Q_res)	p-value for residual homogeneity test
e(seadj)	standard error adjustment
e(converged)	1 if converged, 0 otherwise (with iterative random-effects methods)

Macros

e(cmd)	meta regress
e(cmdline)	command as typed
e(depvar)	name of dependent variable, _meta_es
e(indepvars)	names of independent variables (moderators)
e(title)	title in estimation output
e(model)	meta-analysis model
e(method)	meta-analysis estimation method
e(seadjtype)	type of standard error adjustment
e(properties)	b V
e(estat_cmd)	program used to implement estat
e(predict)	program used to implement predict
e(marginsok)	predictions allowed by margins
e(marginsnotok)	predictions disallowed by margins
e(marginsdefault)	default predict() specification for margins
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices

e(b)	coefficient vector
e(V)	variance–covariance matrix of the estimators

Functions

e(sample)	marks estimation sample

In addition to the above, the following is stored in r():

Matrices

r(table)	matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

meta regress also creates a system variable, _meta_regweight, that contains meta-regression weights.

Methods and formulas

Methods and formulas are presented under the following headings:

Fixed-effects meta-regression
Random-effects meta-regression
Iterative methods for computing $\hat{\tau}^2$
Noniterative methods for computing $\hat{\tau}^2$
Knapp–Hartung standard-error adjustment
Residual homogeneity test
Residual heterogeneity measures

Fixed-effects meta-regression

For an overview of estimation methods used by meta-regression, see Berkey et al. (1995), Sidik and Jonkman (2005), and Viechtbauer et al. (2015).

Consider an FE meta-analysis, where $\hat{\theta}_j \sim N(\theta_j, \hat{\sigma}_j^2)$, θ_j is the true effect size for study j, $\hat{\theta}_j$ is the estimated effect size, and $\hat{\sigma}_j^2$ is the variance of $\hat{\theta}_j$. In an FE meta-regression (Greenland 1987), the study-specific mean, θ_j, is expressed as

$$\theta_j = \beta_0 + \beta_1 x_{1j} + \cdots + \beta_{p-1} x_{p-1,j} = \mathbf{x}_j \boldsymbol{\beta}$$

where $\mathbf{x}_j = (1, x_{1j}, \ldots, x_{p-1,j})$ is a $1 \times p$ vector of categorical and continuous moderators (covariates) and $\boldsymbol{\beta}$ is a $p \times 1$ vector of regression coefficients to be estimated.

Defining $K \times p$ matrix $\mathbf{X} = (\mathbf{x}_1', \mathbf{x}_2', \ldots, \mathbf{x}_K')'$ and $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_K)'$. Let $w_j = 1/\hat{\sigma}_j^2$ be the weight associated with study j in an FE meta-analysis. The vector of estimated regression coefficients is

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\hat{\boldsymbol{\theta}}$$

where $\mathbf{W} = \text{diag}(w_1, w_2, \ldots, w_K)$.

The above FE regression does not account for residual heterogeneity. This can lead to coefficient standard errors that are too small. Thompson and Sharp (1999) incorporated residual heterogeneity into the model by including a multiplicative variance parameter:

$$\hat{\theta}_j \sim N\left(\mathbf{x}_j\boldsymbol{\beta}, \phi\sigma_j^2\right)$$

For a multiplicative FE meta-regression, $\mathbf{W}$ in the above is replaced with $\mathbf{W}^{\phi} = \text{diag}(w_1^{\phi}, w_2^{\phi}, \ldots, w_K^{\phi})$, where the weights are defined as $w_j^{\phi} = 1/(\hat{\phi}\hat{\sigma}_j^2)$. $\hat{\phi}$ is estimated as the mean squared error from the weighted linear regression with weights proportional to $1/\hat{\sigma}_j^2$.

Next, we present another method of incorporating residual heterogeneity by including an additive between-study variance parameter.

Random-effects meta-regression

An RE meta-regression (Berkey et al. 1995) model may be expressed as

$$\hat{\theta}_j = \mathbf{x}_j\boldsymbol{\beta} + u_j + \epsilon_j \qquad u_j \sim N(0,\tau^2) \qquad \epsilon_j \sim N(0,\hat{\sigma}_j^2)$$

All algorithms for RE meta-regression first estimate the between-study variance, τ^2. The regression coefficients are then estimated via weighted least squares,

$$\widehat{\boldsymbol{\beta}}^* = (\mathbf{X}'\mathbf{W}^*\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}^*\hat{\boldsymbol{\theta}}$$

where $\mathbf{W}^* = \text{diag}(w_1^*, w_2^*, \ldots, w_K^*)$ and $w_j^* = 1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$.

All the estimators of τ^2 can be expressed in terms of the matrix

$$\mathbf{P} = \mathbf{A} - \mathbf{A}\mathbf{X}(\mathbf{X}'\mathbf{A}\mathbf{X})^{-1}\mathbf{X}'\mathbf{A} \tag{1}$$

where $\mathbf{A}$ is a $p \times p$ diagonal weight matrix whose elements depend on the type of estimator (Viechtbauer et al. 2015).

The formulas in the following sections are based on Viechtbauer et al. (2015).

Iterative methods for computing $\hat{\tau}^2$

The three estimators described below do not have a closed-form solution, and an iterative algorithm is needed to obtain an estimate of τ^2. The Fisher scoring algorithm, described below, is used to estimate τ^2.

All three estimators start with an initial estimate of τ^2 based on the Hedges estimator, $\hat{\tau}_0^2 = \hat{\tau}_{\text{HE}}^2$, but you can specify your own initial estimate in the `from()` option. The estimate is then updated at each iteration via the formula,

$$\hat{\tau}^2_{\text{new}} = \hat{\tau}^2_{\text{current}} + \delta$$

where δ is a function of $\hat{\tau}^2_{\text{current}}$ and its functional form depends on the estimation method.

The iteration terminates when `reldif`$(\hat{\tau}^2_{\text{new}}, \hat{\tau}^2_{\text{current}})$ is less than `tolerance()` and the scaled gradient, computed based on the log-likelihood functions provided below, is less than `nrtolerance()`; see [R] **Maximize**.

The MLE of τ^2 is the value that maximizes the log-likelihood function (Hardy and Thompson 1996)

$$\ln L_{\text{ML}}(\tau^2) = -\frac{1}{2}\left\{K\ln(2\pi) + \ln\left|\tau^2\mathbf{I} + \mathbf{W}^{-1}\right| + \hat{\boldsymbol{\theta}}'\mathbf{P}\hat{\boldsymbol{\theta}}\right\}$$

The MLE formula for δ is

$$\delta_{\text{MLE}} = \frac{\hat{\boldsymbol{\theta}}'\mathbf{P}\mathbf{P}\hat{\boldsymbol{\theta}} - \text{tr}(\mathbf{W}^*)}{\text{tr}(\mathbf{W}^*\mathbf{W}^*)}$$

The MLE estimator of τ^2 does not incorporate the uncertainty about the unknown regression coefficients $\boldsymbol{\beta}$ and thus can be negatively biased.

The REML estimate of τ^2 is the value that maximizes the restricted log-likelihood function,

$$\ln L_{\text{REML}}\left(\tau^2\right) = \ln L_{\text{ML}}\left(\tau^2\right) - \frac{1}{2}\ln\left|\sum_{j=1}^{K}\frac{1}{\tau^2+\hat{\sigma}_j^2}\mathbf{x}_j'\mathbf{x}_j\right| + \frac{p}{2}\ln(2\pi)$$

and the REML formula for δ is

$$\delta_{\text{REML}} = \frac{\hat{\boldsymbol{\theta}}'\mathbf{P}\mathbf{P}\hat{\boldsymbol{\theta}} - \text{tr}(\mathbf{P})}{\text{tr}(\mathbf{P}\mathbf{P})}$$

The empirical Bayes estimator for τ^2 was introduced by Morris (1983) and was first used in the meta-analytic context by Berkey et al. (1995). This estimator is also known as the Paule–Mandel estimator (Paule and Mandel 1982). The empirical Bayes formula for δ is

$$\delta_{\text{EB}} = \frac{K/(K-p)\hat{\boldsymbol{\theta}}'\mathbf{P}\hat{\boldsymbol{\theta}} - K}{\text{tr}(\mathbf{W}^*)}$$

For the three above estimators, $\mathbf{A} = \mathbf{W}^*$ in the definition of the $\mathbf{P}$ matrix from (1).

Noniterative methods for computing $\hat{\tau}^2$

This section describes noniterative methods, which have closed-form expressions.

The method of moments estimator of τ^2 (DuMouchel and Harris [1983, eq. 3.12]; also see Raudenbush [2009, eq. 16.43]), which can be viewed as an extension of the DerSimonian–Laird estimator from the RE meta-analysis to meta-regression, is

$$\begin{aligned}\hat{\tau}^2_{\text{DL}} &= \frac{\hat{\boldsymbol{\theta}}'\mathbf{P}\hat{\boldsymbol{\theta}} - (K-p)}{\text{tr}(\mathbf{P})} \\ &= \frac{Q_{\text{res}} - (K-p)}{\text{tr}(\mathbf{W}) - \text{tr}\left\{\mathbf{W}\mathbf{X}\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{W}\right\}} = \frac{Q_{\text{res}} - (K-p)}{\sum_{j=1}^{K} w_j\left(1-h_j\right)}\end{aligned} \tag{2}$$

where $\mathbf{P}$ is defined in (1) with $\mathbf{A} = \mathbf{W}$, h_j is the jth diagonal element of the "hat" matrix $\mathbf{X}(\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}$, and Q_{res} is defined in (3).

For a constant-only model, when $p = 1$, (2) reduces to the DerSimonian–Laird estimator from *Noniterative methods* in [META] **meta summarize**.

Hedges (1983) used OLS to provide a method of moments estimator of $\hat{\tau}^2$ for the RE meta-analysis. In the context of meta-regression, the extension of the Hedges's (HE) estimator introduced by Raudenbush (2009, eq. 16.41) is

$$\begin{aligned}\hat{\tau}^2_{\text{HE}} &= \frac{\hat{\boldsymbol{\theta}}'\mathbf{P}\hat{\boldsymbol{\theta}} - \text{tr}(\mathbf{P}\mathbf{W}^{-1})}{K-p} \\ &= \frac{\sum_{j=1}^{K}\left(\hat{\theta}_j - \mathbf{x}_j\widehat{\boldsymbol{\beta}}_{\text{ols}}\right)^2 - \sum_{j=1}^{K}\hat{\sigma}_j^2\left(1-h_j^{\text{ols}}\right)}{K-p}\end{aligned}$$

where $\mathbf{P}$ is defined in (1) with $\mathbf{A} = \mathbf{I}$, $\widehat{\boldsymbol{\beta}}_{\text{ols}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}\hat{\boldsymbol{\theta}}$, and h_j^{ols} is the jth diagonal element of the OLS hat matrix $\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$.

Sidik and Jonkman (2005) proposed the following estimator. Consider an initial estimate of τ^2,

$$\hat{\tau}^2_0 = \frac{\sum_{j=1}^{K}\left(\hat{\theta}_j - \bar{\theta}\right)^2}{K} \qquad \bar{\theta} = \frac{\sum_{j=1}^{K}\hat{\theta}_j}{K}$$

Then, the estimator is defined as

$$\hat{\tau}^2_{\text{SJ}} = \frac{\hat{\boldsymbol{\theta}}'\mathbf{P}\hat{\boldsymbol{\theta}}}{K-p} = \frac{\sum_{j=1}^{K} w_j^{\text{SJ}}\left(\hat{\theta}_j - \mathbf{x}_j\widehat{\boldsymbol{\beta}}_{\text{SJ}}\right)^2}{K-p}$$

where $w_j^{\text{SJ}} = \hat{\tau}^2_0/(\hat{\sigma}^2_j + \hat{\tau}^2_0)$ is a diagonal element of $\mathbf{A}$ from (1), $\widehat{\boldsymbol{\beta}}_{\text{SJ}} = (\mathbf{X}'\mathbf{W}^{\text{SJ}}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}^{\text{SJ}}\hat{\boldsymbol{\theta}}$, and $\mathbf{W}^{\text{SJ}}$ is a $K \times K$ diagonal matrix with elements w_j^{SJ}.

The Sidik–Jonkman estimator is not truncated because, theoretically, it should always produce a nonnegative estimate. However, Viechtbauer et al. (2015) point out that, technically, a negative value can be produced in practice in an unlikely case of all $\hat{\theta}_j$'s being identical.

Viechtbauer et al. (2015) provide the following extension for the estimator of τ^2, which was originally introduced by Schmidt and Hunter (2015) in the context of RE meta-analysis, to meta-regression

$$\hat{\tau}^2_{\text{HS}} = \frac{\hat{\boldsymbol{\theta}}'\mathbf{P}\hat{\boldsymbol{\theta}} - K}{\text{tr}(\mathbf{W})} = \frac{Q_{\text{res}} - K}{\text{tr}(\mathbf{W})}$$

where $\mathbf{P}$ is defined in (1) with $\mathbf{A} = \mathbf{W}$.

Knapp–Hartung standard-error adjustment

By default, the inference about the regression coefficients and their confidence intervals from meta-regression is based on a normal distribution. The test of the significance of all regression coefficients is based on a χ^2 distribution with $p-1$ degrees of freedom.

Knapp and Hartung (2003) proposed an adjustment to the standard errors of the estimated regression coefficients to account for the uncertainty in the estimation of τ^2. They showed that the corresponding tests of individual regression coefficients and their confidence intervals are based on the Student's t distribution with $K-p$ degrees of freedom and that the overall test of significance is based on an F distribution with $p-1$ numerator and $K-p$ denominator.

The Knapp–Hartung adjustment first calculates the quadratic form,

$$q_{\text{KH}} = \frac{\hat{\boldsymbol{\theta}}'\mathbf{P}\hat{\boldsymbol{\theta}}}{K-p}$$

where $\mathbf{P}$ is defined in (1) with $\mathbf{A} = \mathbf{W}^*$. It then multiplies the regular expressions of the variances of regression coefficients by q_{KH} or, in the case of the truncated Knapp–Hartung adjustment, by $\max(1, q_{\text{KH}})$.

Residual homogeneity test

Consider a test of residual homogeneity, which mathematically translates to $H_0: \tau^2 = 0$ for the random-effects meta-regression and to $H_0: \phi = 1$ for the fixed-effects meta-regression with multiplicative dispersion parameter ϕ. This test is based on the residual weighted sum of squares, Q_{res},

$$Q_{\text{res}} = \sum_{j=1}^{K} w_j \left(\hat{\theta}_j - \mathbf{x}_j\widehat{\boldsymbol{\beta}}\right)^2 = \sum_{j=1}^{K} \left(\frac{\hat{\theta}_j - \mathbf{x}_j\widehat{\boldsymbol{\beta}}}{\hat{\sigma}_j}\right)^2 \tag{3}$$

which is a generalization of the heterogeneity test statistic, Q (see *Homogeneity test* in [META] **meta summarize**), to the context of meta-regression.

Under the null hypothesis of residual homogeneity, Q_{res} follows a χ^2 distribution with $K-p$ degrees of freedom (Seber and Lee 2003, sec. 2.4).

Residual heterogeneity measures

The I^2_{res} statistic represents the percentage of residual between-study variation relative to the total variability. For an RE meta-regression, it is defined by Higgins and Thompson (2002) as

$$I^2_{\text{res}} = \frac{\hat{\tau}^2}{\hat{\tau}^2 + s^2} \times 100\% \tag{4}$$

where $s^2 = (K-p)/\text{tr}(\mathbf{P})$ and $\mathbf{A} = \mathbf{W}$ is used to define $\mathbf{P}$. In the meta-regression context, the H^2 statistic is defined as

$$H^2_{\text{res}} = \frac{\hat{\tau}^2 + s^2}{s^2} \tag{5}$$

Adjusted R^2 (Harbord and Higgins 2016; Borenstein et al. 2009) measures the proportion of the between-study variance that is explained by the moderators. It is defined as

$$R^2 = \frac{\hat{\tau}^2_c - \hat{\tau}^2}{\hat{\tau}^2_c} \times 100\% \tag{6}$$

where $\hat{\tau}^2_c$ is the between-study variance estimated from a constant-only model.

References

Berkey, C. S., D. C. Hoaglin, F. Mosteller, and G. A. Colditz. 1995. A random-effects regression model for meta-analysis. *Statistics in Medicine* 14: 395–411. https://doi.org/10.1002/sim.4780140406.

Berlin, J. A., and E. M. Antman. 1992. Advantages and limitations of meta-analytic regressions of clinical trials data. *Controlled Clinical Trials* 13: 422. https://doi.org/10.1016/0197-2456(92)90151-O.

Borenstein, M., L. V. Hedges, J. P. T. Higgins, and H. R. Rothstein. 2009. *Introduction to Meta-Analysis*. Chichester, UK: Wiley.

Colditz, G. A., T. F. Brewer, C. S. Berkey, M. E. Wilson, E. Burdick, H. V. Fineberg, and F. Mosteller. 1994. Efficacy of BCG vaccine in the prevention of tuberculosis: Meta-analysis of the published literature. *Journal of the American Medical Association* 271: 698–702. https://doi.org/10.1001/jama.1994.03510330076038.

Deeks, J. J., P. Macaskill, and L. Irwig. 2005. The performance of tests of publication bias and other sample size effects in systematic reviews of diagnostic test accuracy was assessed. *Journal of Clinical Epidemiology* 58: 882–893. https://doi.org/10.1016/j.jclinepi.2005.01.016.

DuMouchel, W. H., and J. E. Harris. 1983. Bayes methods for combining the results of cancer studies in humans and other species. *Journal of the American Statistical Association* 78: 293–308. https://doi.org/10.2307/2288631.

Greenland, S. 1987. Quantitative methods in the review of epidemiologic literature. *Epidemiologic Reviews* 9: 1–30. https://doi.org/10.1093/oxfordjournals.epirev.a036298.

Harbord, R. M., and J. P. T. Higgins. 2016. "Meta-regression in Stata". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 85–111. 2nd ed. College Station, TX: Stata Press.

Hardy, R. J., and S. G. Thompson. 1996. A likelihood approach to meta-analysis with random effects. *Statistics in Medicine* 15: 619–629. https://doi.org/10.1002/(SICI)1097-0258(19960330)15:6<619::AID-SIM188>3.0.CO;2-A.

Hartung, J., and G. Knapp. 2001a. On tests of the overall treatment effect in meta-analysis with normally distributed responses. *Statistics in Medicine* 20: 1771–1782. https://doi.org/10.1002/sim.791.

———. 2001b. A refined method for the meta-analysis of controlled clinical trials with binary outcome. *Statistics in Medicine* 20: 3875–3889. https://doi.org/10.1002/sim.1009.

Hedges, L. V. 1983. A random effects model for effect sizes. *Psychological Bulletin* 93: 388–395. https://doi.org/10.1037/0033-2909.93.2.388.

Higgins, J. P. T., and S. G. Thompson. 2002. Quantifying heterogeneity in a meta-analysis. *Statistics in Medicine* 21: 1539–1558. https://doi.org/10.1002/sim.1186.

———. 2004. Controlling the risk of spurious findings from meta-regression. *Statistics in Medicine* 23: 1663–1682. https://doi.org/10.1002/sim.1752.

Higgins, J. P. T., S. G. Thompson, J. J. Deeks, and D. G. Altman. 2003. Measuring inconsistency in meta-analyses. *BMJ* 327: 557–560. https://doi.org/10.1136/bmj.327.7414.557.

Knapp, G., and J. Hartung. 2003. Improved tests for a random effects meta-regression with a single covariate. *Statistics in Medicine* 22: 2693–2710. https://doi.org/10.1002/sim.1482.

Konstantopoulos, S., and L. V. Hedges. 2009. "Analyzing effect sizes: Fixed-effects models". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 279–293. 2nd ed. New York: Russell Sage Foundation.

Morris, C. N. 1983. Parametric empirical Bayes inference: Theory and applications. *Journal of the American Statistical Association* 78: 47–55. https://doi.org/10.2307/2287098.

Paule, R. C., and J. Mandel. 1982. Consensus values and weighting factors. *Journal of Research of the National Bureau of Standards* 87: 377–385. https://doi.org/10.6028/jres.087.022.

Raudenbush, S. W. 2009. "Analyzing effect sizes: Random-effects models". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 295–316. 2nd ed. New York: Russell Sage Foundation.

Schmidt, F. L., and J. E. Hunter. 2015. *Methods of Meta-Analysis: Correcting Error and Bias in Research Findings*. 3rd ed. Thousand Oaks, CA: Sage. https://doi.org/10.4135/9781483398105.

Seber, G. A. F., and A. J. Lee. 2003. *Linear Regression Analysis*. 2nd ed. Hoboken, NJ: Wiley. https://doi.org/10.1002/9780471722199.

Sharp, S. J. 2016. "Meta-analysis regression". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 112–120. 2nd ed. College Station, TX: Stata Press.

Sidik, K., and J. N. Jonkman. 2002. A simple confidence interval for meta-analysis. *Statistics in Medicine* 21: 3153–3159. https://doi.org/10.1002/sim.1262.

———. 2005. A note on variance estimation in random effects meta-regression. *Journal of Biopharmaceutical Statistics* 15: 823–838. https://doi.org/10.1081/BIP-200067915.

Thompson, S. G., and J. P. T. Higgins. 2002. How should meta-regression analyses be undertaken and interpreted? *Statistics in Medicine* 21: 1559–1573. https://doi.org/10.1002/sim.1187.

Thompson, S. G., and S. J. Sharp. 1999. Explaining heterogeneity in meta-analysis: A comparison of methods. *Statistics in Medicine* 18: 2693–2708. https://doi.org/10.1002/(sici)1097-0258(19991030)18:20<2693::aid-sim235>3.0.co;2-v.

Viechtbauer, W., J. A. López-López, J. Sánchez-Meca, and F. Marín-Martínez. 2015. A comparison of procedures to test for moderators in mixed-effects meta-regression models. *Psychological Methods* 20: 360–374. https://doi.org/10.1037/met0000023.

Also see

[META] **meta regress postestimation** — Postestimation tools for meta regress

[META] **meta data** — Declare meta-analysis data

[META] **meta forestplot** — Forest plots+

[META] **meta galbraithplot** — Galbraith plots

[META] **meta labbeplot** — L'Abbé plots

[META] **meta summarize** — Summarize meta-analysis data+

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

[U] 20 Estimation and postestimation commands

meta regress postestimation — Postestimation tools for meta regress

Postestimation commands | predict | margins
Remarks and examples | Methods and formulas | References
Also see

Postestimation commands

The following postestimation command is of special interest after `meta regress`:

Command	Description
`estat bubbleplot`	bubble plots

The following standard postestimation commands are also available:

Command	Description
`contrast`	contrasts and ANOVA-style joint tests of parameters
`estat summarize`	summary statistics for the estimation sample
`estat vce`	variance–covariance matrix of the estimators (VCE)
`estimates`	cataloging estimation results
`etable`	table of estimation results
`lincom`	point estimates, standard errors, testing, and inference for linear combinations of parameters
`margins`	marginal means, predictive margins, marginal effects, and average marginal effects
`marginsplot`	graph the results from margins (profile plots, interaction plots, etc.)
`nlcom`	point estimates, standard errors, testing, and inference for nonlinear combinations of parameters
`predict`	predictions and their SEs, leverage statistics, etc.
`predictnl`	point estimates, standard errors, testing, and inference for generalized predictions
`pwcompare`	pairwise comparisons of parameters
`test`	Wald tests of simple and composite linear hypotheses
`testnl`	Wald tests of nonlinear hypotheses

predict

Description for predict

`predict` creates a new variable containing predictions such as linear predictions, residuals, leverage, and standard errors. After random-effects meta-regression, you can also obtain estimates of random effects and their standard errors.

Menu for predict

Statistics > Postestimation

Syntax for predict

Syntax for obtaining predictions other than best linear unbiased predictions (BLUPs) of random effects

`predict` [*type*] *newvar* [*if*] [*in*] [, *statistic* `fixedonly` `se(`*sespec*`)`]

Syntax for obtaining BLUPs of random effects and their standard errors after random-effects meta-regression

`predict` [*type*] *newvar* [*if*] [*in*], `reffects` [`se(`*newvar*`)` `reses(`*resesspec*`)`]

statistic	Description
Main	
`xb`	linear prediction; the default
`stdp`	standard error of the linear prediction
`fitted`	fitted values, fixed-portion linear prediction plus predicted random effects
`residuals`	residuals, response minus fitted values
`leverage` | `hat`	leverage (diagonal elements of hat matrix)

Unstarred statistics are available both in and out of sample; type `predict ... if e(sample) ...` if wanted only for the estimation sample.

Options for predict

Main

`xb`, the default, calculates the linear prediction $\mathbf{x}_j\widehat{\boldsymbol{\beta}}$. For the random-effects meta-regression, this corresponds to the fixed portion of the linear predictor based on the estimated regression coefficients. That is, this is equivalent to fixing all random effects in the model to their theoretical mean value of 0.

`stdp` calculates the standard error of the linear prediction.

`fitted` calculates the fitted values. With fixed-effects meta-regression or with random-effects meta-regression when option `fixedonly` is also specified, this option is equivalent to `xb`. For random-effects meta-regression without `fixedonly`, it calculates $\mathbf{x}_j\widehat{\boldsymbol{\beta}}+u_j$, which is equal to the fixed portion of the linear prediction plus predicted random effects.

`residuals` calculates the residuals, which are equal to the responses minus the fitted values. With fixed-effects meta-regression or with random-effects meta-regression when option `fixedonly` is also specified, it calculates $\widehat{\theta}_j - \mathbf{x}_j\widehat{\boldsymbol{\beta}}$. The former are known as marginal residuals in the context of the random-effects model. For random-effects meta-regression without `fixedonly`, this option calculates $\widehat{\theta}_j - (\mathbf{x}_j\widehat{\boldsymbol{\beta}} + u_j)$, which are known as conditional residuals.

`leverage` or `hat` calculates the diagonal elements of the projection ("hat") matrix.

`fixedonly` specifies that all random effects be set to zero, which is equivalent to using only the fixed portion of the model, when computing results for random-effects models. This option may be specified only with statistics `fitted`, `residuals`, or `leverage`.

`reffects` calculates best linear unbiased predictions (BLUPs) of the random effects.

`se(`*newvar*`[ , marginal ])` calculates the standard errors of the corresponding predicted values. This option may be specified only with statistics `reffects`, `fitted`, and `residuals`. When specified with `reffects`, `se(`*newvar*`)` is a synonym to `reses(`*newvar*`, diagnostic)`.

Suboption `marginal` is allowed only with random-effects meta-regression and requires option `fixedonly`. It computes marginal standard errors, when you type

`. predict ...,` *statistic* `se(`*newvar*`, marginal) fixedonly`

instead of the standard errors conditional on zero random effects, which are computed when you type

`. predict ...,` *statistic* `se(`*newvar*`) fixedonly`

`marginal` is not allowed in combination with `reffects`.

`reses(`*resesspec*`)` calculates the standard errors of the random effects; see option `reffects`. This option may not be combined with option `se()`. The syntax for *resesspec* is

newvar`[ , comparative | diagnostic ]`

`comparative`, the default, computes comparative random-effects standard errors. For linear models, these correspond to posterior standard deviations of random effects and to standard errors of marginal prediction errors $\widehat{u}_j - u_j$. These standard errors are used for inference about the random effects.

`diagnostic` computes diagnostic random-effects standard errors. These correspond to marginal standard errors of BLUPs, $\mathrm{SE}(\widehat{u}_j)$. These standard errors are used for model diagnostics.

margins

Description for margins

`margins` estimates margins of response for linear predictions.

Menu for margins

Statistics > Postestimation

Syntax for margins

`margins` [*marginlist*] [`,` *options*]

`margins` [*marginlist*] `,` `predict(`*statistic* `...)` [*options*]

statistic	Description
`xb`	linear prediction; the default
`fitted`	fitted values; implies `fixedonly`
`stdp`	not allowed with `margins`
`residuals`	not allowed with `margins`
`leverage` | `hat`	not allowed with `margins`
`reffects`	not allowed with `margins`

Statistics not allowed with `margins` are functions of stochastic quantities other than `e(b)`.

For the full syntax, see [R] **margins**.

Remarks and examples

We demonstrate some of the postestimation features, including `estat bubbleplot`, `margins`, and `predict` after `meta regress`.

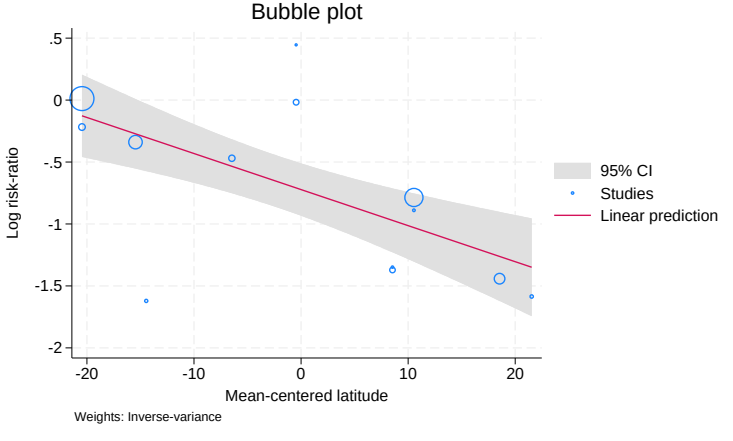
Example 1: Bubble plot

Consider the declared BCG dataset of clinical trials that studied the efficacy of a Bacillus Calmette-Guérin (BCG) vaccine in the prevention of tuberculosis (TB) (Colditz et al. 1994). In example 1 of [META] **meta regress**, we used `meta regress` to fit a simple meta-regression to these data with the continuous moderator `latitude_c` to explore heterogeneity.

```
. use https://www.stata-press.com/data/r18/bcgset
(Efficacy of BCG vaccine against tuberculosis; set with -meta esize-)
. meta regress latitude_c
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Random-effects meta-regression                  Number of obs  =        13
Method: REML                                    Residual heterogeneity:
                                                          tau2 =   .07635
                                                       I2 (%)  =   68.39
                                                           H2  =    3.16
                                                R-squared (%)  =   75.63
                                                Wald chi2(1)   =   16.36
                                                Prob > chi2    =  0.0001
------------------------------------------------------------------------------
     _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
--------------+---------------------------------------------------------------
   latitude_c |  -.0291017   .0071953    -4.04   0.000    -.0432043   -.0149991
        _cons |  -.7223204   .1076535    -6.71   0.000    -.9333174   -.5113234
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(11) = 30.73   Prob > Q_res = 0.0012
```

Whenever there is one continuous moderator in a meta-regression, a so-called bubble plot is commonly used to explore the relationship between the effect size and that moderator. Let's use `estat bubbleplot` to produce the bubble plot after the fitted meta-regression.

```
. estat bubbleplot
```



A bubble plot is a scatterplot of the observed effect sizes against the moderator overlaid with the predicted regression and confidence-intervals lines. Each study is represented by a circle (bubble) with the size (area) proportional to the study precision, $1/\hat{\sigma}_j^2$. The larger the size of the bubble, the more precise the study. The coordinates of the center of each circle show the observed value of the effect size on the y axis and that of the moderator (`latitude_c` in our example) on the x axis. The solid line shows the predicted values (predicted log risk-ratios in our example). The predicted 95% confidence intervals are also plotted.

From the plot, the log risk-ratio for the BCG vaccine declines as the distance from the equator increases. There appear to be a couple of outlying studies (see points in the bottom left and middle top sections of the plot), but their bubbles are very small, which suggests that their log risk-ratios estimates had small weights, relative to other studies, in the meta-regression. Outlying studies with large bubbles may be a source of concern because of the large differences in their effect sizes compared with those from the other studies and because of the large impact they have on the regression results.

◁

▷ Example 2: Marginal effects

Continuing with example 1, we found that the log risk-ratio for the BCG decreases as the distance from the equator increases. For example, from the bubble plot, a trial conducted relatively close to the equator, say, in Thailand (with a latitude of 15 or a centered latitude of −18.5), would have a predicted log risk-ratio of about −0.2. A trial conducted in, say, Nepal (with a latitude of 28 or a centered latitude of −5.5), would have a predicted log risk-ratio of about −0.7. And a trial conducted in, say, Ukraine (with a latitude of 50 or a centered latitude of 16.5), would have a predicted log risk-ratio of about −1.

Instead of relying on the graph, we can obtain more precise estimates of the predicted log risk-ratios at different latitude values by using the `margins` command as follows:

```
. margins, at(latitude_c = (-18.5 -5.5 16.5))
Adjusted predictions                                      Number of obs = 13
Expression: Fitted values; fixed portion (xb), predict(fitted fixedonly)
1._at: latitude_c = -18.5
2._at: latitude_c =  -5.5
3._at: latitude_c =  16.5
```

	Delta-method Margin	std. err.	z	P>|z|	[95% conf.	interval]
_at						
1	-.1839386	.1586092	-1.16	0.246	-.4948069	.1269297
2	-.562261	.1091839	-5.15	0.000	-.7762574	-.3482645
3	-1.202499	.1714274	-7.01	0.000	-1.53849	-.8665072

The list of numbers specified in the `at()` option are the values of the latitudes centered around the latitude mean (≈ 33.5).

Note that results produced by `margins` are on the log scale and need to be exponentiated to make interpretations on the natural (risk) scale. For instance, from the output, the risk ratio for regions with `latitude_c` = 16.5 is $\exp(-1.202499) = 0.3$, which means that the vaccine is expected to reduce the risk of TB by 70% for regions with that latitude.

◁

▷ Example 3: Predicted random effects

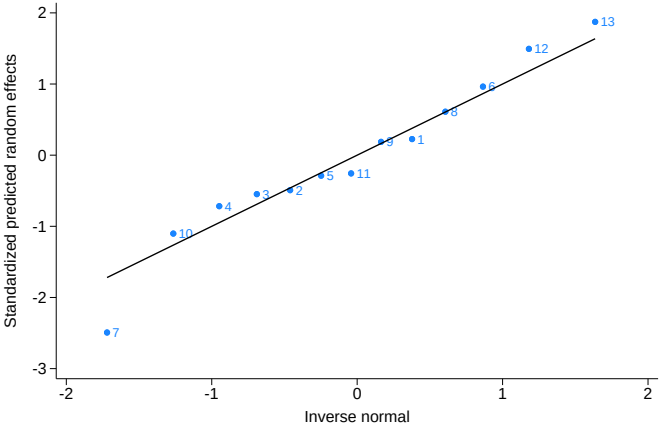
In example 1, we noticed a couple of outlying studies. Let's explore this further by looking at predicted random effects from our random-effects meta-regression.

We first use `predict` with options `reffects` and `se()` to predict the random-effects and estimate their diagnostic standard errors.

```
. predict double u, reffects se(se_u)
```

Then, we generate a new variable, `ustandard`, as the ratio of the predicted random effects to their diagnostic standard errors and use the `qnorm` command (see [R] **Diagnostic plots**) to construct the normal quantile plot.

```
. generate double ustandard = u/se_u
. label variable ustandard "Standardized predicted random effects"
. qnorm ustandard, mlabel(trial)
```



The plot suggests that trial 7, labeled "Vandiviere et al., 1973" in our data, is an outlier. From the data, the log risk-ratio estimate for this trial is -1.62 with the corresponding risk-ratio estimate of about 0.2. This means that, in that trial, the vaccine reduced the risk of TB by roughly 80% even though this trial was conducted relatively close to the equator (in Haiti, with `latitude=19`). In fact, this trial reported the largest risk reduction (smallest log-risk-ratio value) in the meta-analysis. Compare this with trial 11 ("Comstock et al., 1974"), which was conducted in Puerto Rico and has a similar latitude (`latitude=18`) but whose estimated risk reduction was much more moderate, about 29% (with the risk-ratio estimate of $\exp(-0.34) = 0.71$). More investigation is needed to explain the extreme value reported by trial 7. Thus, in this example, you may consider reporting the results of meta-analyses with and without this trial.

◁

Methods and formulas

Methods and formulas are presented under the following headings:

Random-effects meta-regression
Fixed-effects meta-regression

The following formulas are used by `predict`. The notation is based on *Methods and formulas* of [META] **meta regress**.

Random-effects meta-regression

The fixed-portion of the linear prediction (option `xb`) is $\mathbf{x}_j\widehat{\boldsymbol{\beta}}$. The estimated standard error of the fixed-portion of the linear prediction (option `stdp`) for study j is

$$\widehat{\mathrm{SE}}\left(\mathbf{x}_j\widehat{\boldsymbol{\beta}}\right) = \sqrt{\mathbf{x}_j\left(\mathbf{X}'\mathbf{W}^*\mathbf{X}\right)^{-1}\mathbf{x}_j'}$$

The BLUP of the jth random effect (option `reffects`) is

$$\hat{u}_j = \lambda_j\left(\hat{\theta}_j - \mathbf{x}_j\widehat{\boldsymbol{\beta}}\right)$$

where

$$\lambda_j = \frac{\hat{\tau}^2}{\hat{\tau}^2 + \hat{\sigma}_j^2}$$

is the empirical Bayes shrinkage factor for the jth study. When the `reses()` option is also specified, the estimated comparative standard error of $\hat{u}_j$ is

$$\widehat{\mathrm{SE}}\left(\hat{u}_j - u_j\right) = \sqrt{\hat{\tau}^2 - \lambda_j^2\left\{\hat{\sigma}_j^2 + \hat{\tau}^2 - \mathbf{x}_j\left(\mathbf{X}'\mathbf{W}^*\mathbf{X}\right)^{-1}\mathbf{x}_j'\right\}}$$

When suboption `diagnostic` of `reses()` is specified or when the `se()` option is specified, the estimated diagnostic standard error of $\hat{u}_j$ is

$$\widehat{\mathrm{SE}}\left(\hat{u}_j\right) = \lambda_j\sqrt{\hat{\sigma}_j^2 + \hat{\tau}^2 - \mathbf{x}_j\left(\mathbf{X}'\mathbf{W}^*\mathbf{X}\right)^{-1}\mathbf{x}_j'}$$

See Goldstein (2011), Skrondal and Rabe-Hesketh (2009), and Rabe-Hesketh and Skrondal (2022) for more details.

The fitted value (option `fitted`) is

$$\tilde{\theta}_j = \mathbf{x}_j\widehat{\boldsymbol{\beta}} + \hat{u}_j$$

When the `se()` option is also specified, the estimated standard error of $\tilde{\theta}_j$ is

$$\widehat{\mathrm{SE}}\left(\tilde{\theta}_j\right) = \sqrt{\lambda_j^2\left(\hat{\sigma}_j^2 + \hat{\tau}^2\right) + \left(1 - \lambda_j^2\right)\mathbf{x}_j\left(\mathbf{X}'\mathbf{W}^*\mathbf{X}\right)^{-1}\mathbf{x}_j'}$$

The residual (option `residuals`) is

$$e_j = \hat{\theta}_j - \tilde{\theta}_j$$

When option `se()` is also specified, the estimated standard error of e_j is

$$\widehat{\mathrm{SE}}\left(e_j\right) = \sqrt{\left(1+\lambda_j^2\right)\left(\hat{\sigma}_j^2 + \hat{\tau}^2 - \mathbf{x}_j\left(\mathbf{X}'\mathbf{W}^*\mathbf{X}\right)^{-1}\mathbf{x}_j'\right)}$$

The leverage (option `hat`) are the diagonal elements of the hat matrix $\mathbf{X}\left(\mathbf{X}'\mathbf{W}^*\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{W}^*$:

$$h_j^* = \frac{1}{\hat{\tau}^2+\hat{\sigma}_j^2}\mathbf{x}_j\left(\mathbf{X}'\mathbf{W}^*\mathbf{X}\right)^{-1}\mathbf{x}_j'$$

When the `fixedonly` option is specified, the formulas for the fitted values and residuals (including their standard errors) and leverage are adjusted by replacing the value of $\hat{u}_j$ with 0, in which case, $\hat{\tau}^2 = 0$, $\lambda_j = 0$, and $\mathbf{W}^*$ is replaced with $\mathbf{W} = \mathrm{diag}\left(1/\hat{\sigma}_1^2, \ldots, 1/\hat{\sigma}_K^2\right)$. In this case, the standard errors are computed conditionally on zero random effects.

If `se()`'s option `marginal` is specified, then marginal standard errors are computed. This is equivalent to computing $\widehat{\mathrm{SE}}(\tilde{\theta}_j)$ and $\widehat{\mathrm{SE}}\left(e_j\right)$ with $\lambda_j = 0$ but keeping $\hat{\tau}^2$ and $\mathbf{W}^*$ unchanged.

Fixed-effects meta-regression

The linear prediction (option `xb`) is $\mathbf{x}_j\widehat{\boldsymbol{\beta}}$. The estimated standard error of the linear prediction (option `stdp`) for study j is

$$\widehat{\mathrm{SE}}\left(\mathbf{x}_j\widehat{\boldsymbol{\beta}}\right) = \sqrt{\mathbf{x}_j\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{x}_j'}$$

The fitted value (option `fitted`) is the same as the linear prediction:

$$\tilde{\theta}_j = \mathbf{x}_j\widehat{\boldsymbol{\beta}}$$

The residual (option `residuals`) is

$$e_j = \hat{\theta}_j - \tilde{\theta}_j$$

When option `se()` is also specified, the estimated standard error of e_j is

$$\widehat{\mathrm{SE}}\left(e_j\right) = \sqrt{\left(\hat{\sigma}_j^2 - \mathbf{x}_j\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{x}_j'\right)}$$

The leverage (option `hat`) are the diagonal elements of the hat matrix $\mathbf{X}\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{W}$:

$$h_j = \frac{1}{\tilde{\sigma}_j^2}\mathbf{x}_j\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{x}_j'$$

For the multiplicative fixed-effects meta-regression, in the above formulas, replace $\mathbf{W}$ with $\mathbf{W}^{\phi}$ and $\hat{\sigma}_j^2$ with $\hat{\phi}\hat{\sigma}_j^2$, where $\hat{\phi}$ is defined in *Fixed-effects meta-regression* in [META] **meta regress**.

References

Colditz, G. A., T. F. Brewer, C. S. Berkey, M. E. Wilson, E. Burdick, H. V. Fineberg, and F. Mosteller. 1994. Efficacy of BCG vaccine in the prevention of tuberculosis: Meta-analysis of the published literature. *Journal of the American Medical Association* 271: 698–702. https://doi.org/10.1001/jama.1994.03510330076038.

Goldstein, H. 2011. *Multilevel Statistical Models*. 4th ed. Chichester, UK: Wiley. https://doi.org/10.1002/9780470973394.

Rabe-Hesketh, S., and A. Skrondal. 2022. *Multilevel and Longitudinal Modeling Using Stata*. 4th ed. College Station, TX: Stata Press.

Skrondal, A., and S. Rabe-Hesketh. 2009. Prediction in multilevel generalized linear models. *Journal of the Royal Statistical Society*, A ser., 172: 659–687. https://doi.org/10.1111/j.1467-985X.2009.00587.x.

Also see

[META] **meta regress** — Meta-analysis regression

[META] **meta data** — Declare meta-analysis data

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

[U] **20 Estimation and postestimation commands**

estat bubbleplot — Bubble plots after meta regress

Description Quick start Menu Syntax Options
Remarks and examples Methods and formulas References Also see

Description

`estat bubbleplot` produces bubble plots after simple meta-regression with one continuous moderator performed by using `meta regress`. The bubble plot is a scatterplot of effect sizes against a moderator of interest overlaid with the predicted regression line and confidence-interval bands. In a bubble plot, the marker sizes, "bubbles", are proportional to study weights.

Quick start

Fit a random-effects meta-regression with a continuous moderator, `x`

```
meta regress x, random
```

Construct a bubble plot for `x`

```
estat bubbleplot
```

Same as above, but specify that the size of the marker representing studies be proportional to the random-effects weights instead of the default fixed-effects weights

```
estat bubbleplot, reweighted
```

Construct a bubble plot with a 90% confidence interval

```
estat bubbleplot, level(90)
```

Menu

Statistics > Meta-analysis

Syntax

```
estat bubbleplot [ if ] [ in ] [ , options ]
```

options	Description
Main	
reweighted	make bubble size depend on random-effects weights
[no]regline	display or suppress the regression line
[no]ci	display or suppress the confidence intervals
level(#)	set confidence level; default is as declared for meta-analysis
n(#)	evaluate CI lines at # points; default is n(100)
Fitted line	
lineopts(*line_options*)	affect rendition of the plotted regression line
CI plot	
ciopts(*ciopts*)	affect rendition of the plotted CI band
Add plots	
addplot(*plot*)	add other plots to the bubble plot
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than by() documented in [G-3] ***twoway_options***

Options

Main

reweighted is used with random-effects meta-regression. It specifies that the sizes of the bubbles be proportional to the weights from the random-effects meta-regression, $w_j^* = 1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$. By default, the sizes are proportional to the precision of each study, $w_j = 1/\hat{\sigma}_j^2$.

regline and noregline display or suppress the rendition of the regression line. The default, regline, is to display the regression line. Option noregline implies option noci.

ci and noci display or suppress confidence intervals. The default, ci, is to display them.

level(#) specifies the confidence level, as a percentage, for confidence intervals. The default is as declared for the meta-analysis session; see *Declaring a confidence level for meta-analysis* in [META] **meta data**. Also see option level() in [META] **meta set**.

n(#) specifies the number of points at which to evaluate the CIs. The default is n(100).

Fitted line

lineopts(*line_options*) affects the rendition of the plotted regression line; see [G-3] ***line_options***.

CI plot

ciopts(*ciopts*) affects the rendition of the CI band in the bubble plot. *ciopts* are any options as defined in [G-2] **graph twoway rline** and option recast(rarea) as described in [G-3] ***advanced_options***.

Add plots

`addplot(`*plot*`)` allows adding more `graph twoway` plots to the graph; see [G-3] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G-3] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G-3] ***title_options***) and for saving the graph to disk (see [G-3] ***saving_option***).

Remarks and examples

Remarks are presented under the following headings:

Introduction
Examples of using estat bubbleplot

Introduction

A bubble plot (Berkey et al. 1995 ; Thompson and Sharp 1999; Thompson and Higgins 2002) is used after simple meta-regression with a continuous moderator to describe the relation between the effect size and the corresponding moderator. It is used as a tool to assess how well the regression model fits the data and to potentially identify influential and outlying studies. The bubble plot is a scatterplot with the study-specific effect sizes plotted on the y axis and the moderator of interest from the meta-regression plotted on the x axis. The sizes of the markers or "bubbles" are proportional to the precision of each study. The more precise (larger) the study, the larger the size of the bubble. The predicted regression line and confidence bands are overlaid with the scatterplot.

`estat bubbleplot` produces bubble plots after simple meta-regression with a continuous moderator performed by using `meta regress`. Traditionally, the weights used to determine the sizes of the bubbles are the inverses of the effect-size variances, $1/\hat{\sigma}_j^2$. After a random-effects meta-regression, you can specify the `reweighted` option to instead use the random-effects weights, $1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$.

The predicted regression line and the 95% confidence intervals are plotted by default. You can specify the `level()` option to obtain other confidence intervals. You can control the look of the lines by specifying the options `lineopts()` and `ciopts()`. You can also suppress the lines by specifying the options `noregline` and `noci`.

Examples of using estat bubbleplot

In the examples that follow, we demonstrate how to create and customize bubble plots after a meta-regression. Consider the BCG dataset from *Examples of using meta regress* in [META] **meta regress**.

```
. use https://www.stata-press.com/data/r18/bcgset
(Efficacy of BCG vaccine against tuberculosis; set with -meta esize-)
. meta query, short
-> meta esize npost - nnegc, esize(lnrratio) studylabel(studylbl)
  Effect-size label: Log risk-ratio
   Effect-size type: lnrratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Random effects
             Method: REML
```

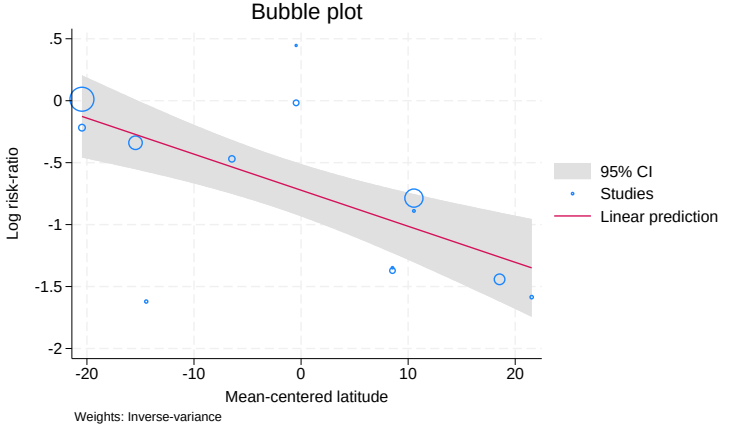
To create these plots, we first fit the random-effects meta-regression shown in example 1 of [META] **meta regress**, but our focus here is not on the interpretation of these plots but on the variety of bubble plots that can be created.

```
. meta regress latitude_c
  Effect-size label: Log risk-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Random-effects meta-regression                  Number of obs   =        13
Method: REML                                    Residual heterogeneity:
                                                          tau2  =   .07635
                                                       I2 (%)   =   68.39
                                                          H2    =    3.16
                                                R-squared (%)   =   75.63
                                                Wald chi2(1)    =    16.36
                                                Prob > chi2     =   0.0001
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
  latitude_c |  -.0291017   .0071953    -4.04   0.000    -.0432043   -.0149991
       _cons |  -.7223204   .1076535    -6.71   0.000    -.9333174   -.5113234
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(11) = 30.73   Prob > Q_res = 0.0012
```

Example 1: A basic bubble plot

To construct a bubble plot after performing a simple meta-regression, we simply type

```
. estat bubbleplot
```

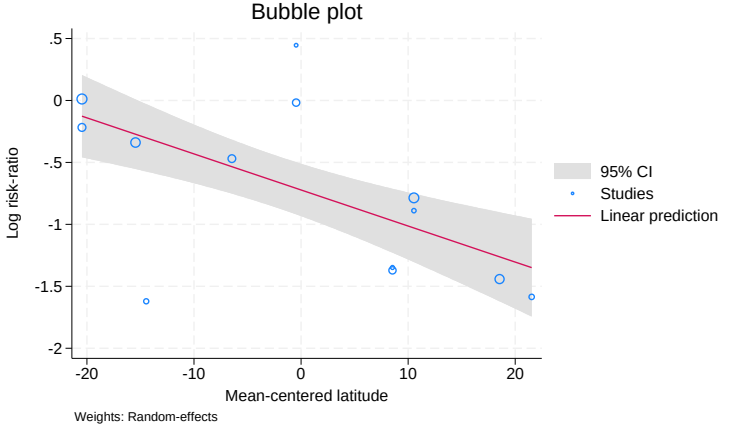


The graph shows the log risk-ratios plotted against the mean-centered latitudes of the studies' locations. By default, the regression line and corresponding confidence intervals are plotted. We could suppress these and plot just the bubbles with options `noregline` and `noci`, respectively. The regression line provides a good fit of the data because most studies are relatively close to it. The log risk-ratios for the BCG vaccine decline with increasing latitude. For more interpretation of the above bubble plot, refer to example 1 of [META] **meta regress postestimation**.

Example 2: Reweighting the bubbles

By default, the bubble sizes are proportional to trial precisions, $1/\hat{\sigma}_j^2$. With the `reweighted` option, we can make the bubble sizes proportional to the weights from the random-effects meta-regression, $1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$. For example, continuing with example 1, we can reweight the bubbles as follows:

```
. estat bubbleplot, reweighted
```

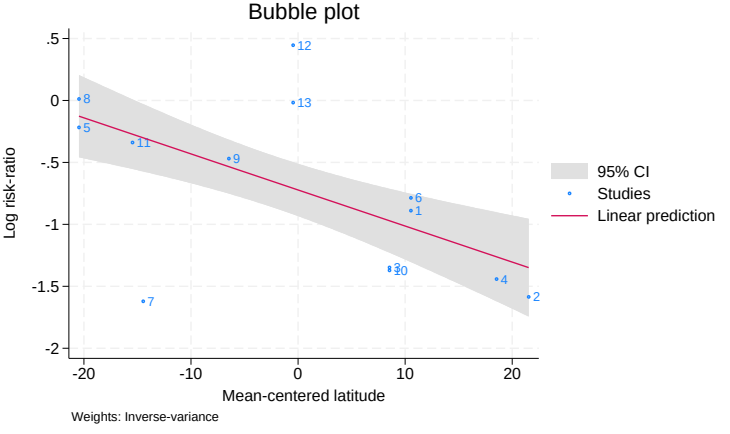


With random-effects weights, the sizes of the bubbles are more uniform across the studies as compared with precision (fixed-effects) weights used in example 1. This will always be true except when $\hat{\tau}^2 = 0$, in which case the bubble sizes will be identical with both types of weights.

▷ Example 3: Using addplot() to add labels for the trials

Below, we discuss how you can add labels to the trials, which are represented by the hollow circles on a bubble plot. Typically, we use `mlabel(`*varname*`)` to add marker labels. For example, if we wish to label the trials according to their trial ID, `trial`, we type

```
. estat bubbleplot, mlabel(trial)
```



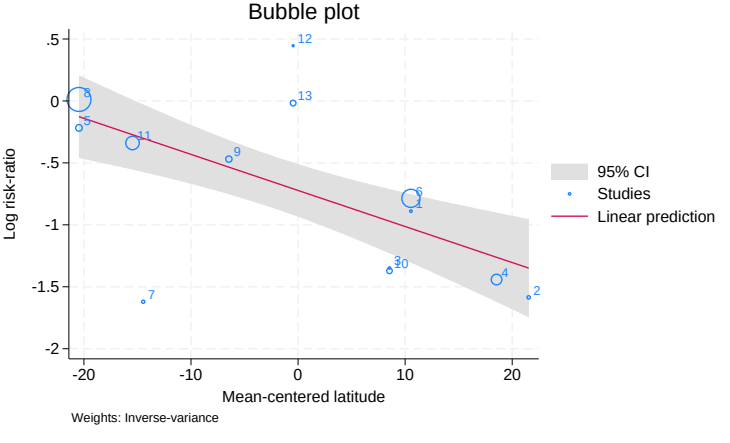
Specifying the `mlabel()` option causes all the markers to have the same size. One way to get around this is by using the `addplot()` option.

We can use `addplot()` to overlay an exact copy of the properly weighted bubble plot but without plotting the markers and symbols, that is, using the `msymbol(none)` option. We can then add labels to these nonplotted symbols to obtain the desired plot. Here is our minimal `addplot()` specification:

```
addplot(scatter _meta_es latitude_c, msymbol(none) mlabel(trial))
```

The full specification is

```
. local opts msymbol(none) mlabel(trial) mlabcolor(stblue) legend(order(1 2 3))
. estat bubbleplot, addplot(scatter _meta_es latitude_c, 'opts' mlabpos(2))
```



We used additional options to fine-tune the bubble plot. The `mlabcolor(stblue)` option controls the color of the study labels. The `legend(order(1 2 3))` option prevents the display of a legend key for the added scatterplot in the legend box at the bottom of the plot. Finally, the `mlabpos(2)` option specifies that marker labels be drawn at the 2 o'clock position.

◁

▷ Example 4: Adjusting label positions

Continuing with example 3, let's customize the labels further. For example, marker labels 3 and 10 (and 6 and 1) are not easily distinguishable. You may provide individual marker label positions for each study by using the `mlabvpos(`*varname*`)` option. *varname* must be created to hold the individual positions (an integer number between 0 to 12) of each marker.

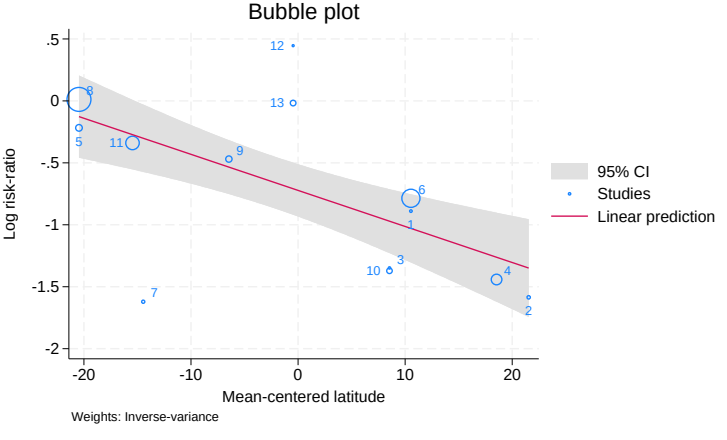
Let's generate our position variable.

```
. generate byte pos = 2
. quietly replace pos = 9 in 10/13
. quietly replace pos = 6 if inlist(trial,1,2,5)
```

We generated a new variable `pos` to hold the individual positions of each marker label. We chose to draw labels at 9 o'clock for trials 10 to 13, at 6 o'clock for trials 1, 2, and 5, and at 2 o'clock for the other trials.

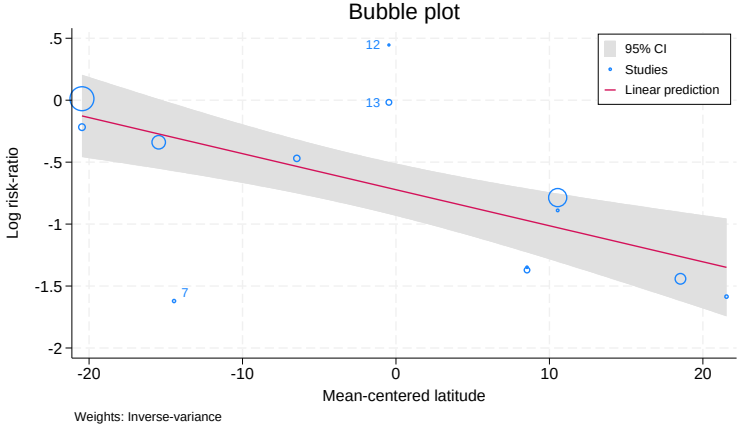
We now use a similar specification of `addplot()` from example 3, but here we add `mlabvpos(pos)` and `mlabgap(*2)` to double the gap size between the marker labels and the markers so that the trial labels do not touch the hollow circles; see trials 6 and 8.

```
. estat bubbleplot, addplot(scatter _meta_es latitude_c, mlabvpos(pos)
> mlabgap(*2) 'opts')
```



We can modify other aspects of the graph such as the legend. Let's place the legend inside the plot region. We also specify `if inlist(trial,7,12,13)` with `estat bubbleplot` to display trial labels only for specific trials (for example, outliers, trials with large weights, and so on).

```
. local legopts legend(ring(0) position(2) size(small) symxsize(*0.3)
> region(lcolor(black)))
. estat bubbleplot, addplot(scatter _meta_es latitude_c
> if inlist(trial,7,12,13), mlabvpos(pos) mlabgap(*2) 'opts') 'legopts'
```



Within the `legend()` option (see [G-3] ***legend_options***), `ring(0)` and `position(2)` specify that the legend be placed inside the plot region at the 2 o'clock position. `size(small)` specifies that a small font be used for the legend key text, and `symxsize(*0.3)` sets the width of the key symbols to 30% of their default width. `region(lcolor(black))` adds a black border around the legend region.

◁

Methods and formulas

`estat bubbleplot` produces a scatterplot with the effect sizes, $\hat{\theta}_j$, stored in the system variable `_meta_es` on the y axis and a moderator from the meta-regression on the x axis. By default, the bubble size is proportional to $w_j = 1/\hat{\sigma}_j^2$. For a random-effects meta-regression, if you specify the `reweighted` option, the weights $w_j^* = 1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$ will be used.

For a simple meta-regression with moderator x_1, the plotted predicted line is $\hat{\beta}_0 + \hat{\beta}_1 x_1 = \mathbf{x}_j\widehat{\boldsymbol{\beta}}$. The CIs are computed as

$$\mathbf{x}_j\widehat{\boldsymbol{\beta}} \pm z_{1-\alpha/2}\widehat{\mathrm{SE}}\left(\mathbf{x}_j\widehat{\boldsymbol{\beta}}\right)$$

where the computation of $\widehat{\mathrm{SE}}(\mathbf{x}_j\widehat{\boldsymbol{\beta}})$ is described in [META] **meta regress postestimation**. The `n()` option specifies how many evaluation points are used to construct the CI plots. By default, 100 points are used. When the `se()` or `tdistribution` option is specified with `meta regress`, the confidence intervals use the $t_{K-2,1-\alpha/2}$ critical value.

References

Berkey, C. S., D. C. Hoaglin, F. Mosteller, and G. A. Colditz. 1995. A random-effects regression model for meta-analysis. *Statistics in Medicine* 14: 395–411. https://doi.org/10.1002/sim.4780140406.

Thompson, S. G., and J. P. T. Higgins. 2002. How should meta-regression analyses be undertaken and interpreted? *Statistics in Medicine* 21: 1559–1573. https://doi.org/10.1002/sim.1187.

Thompson, S. G., and S. J. Sharp. 1999. Explaining heterogeneity in meta-analysis: A comparison of methods. *Statistics in Medicine* 18: 2693–2708. https://doi.org/10.1002/(sici)1097-0258(19991030)18:20<2693::aid-sim235>3.0.co;2-v.

Also see

[META] **meta regress** — Meta-analysis regression

[META] **meta regress postestimation** — Postestimation tools for meta regress

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta funnelplot — Funnel plots

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
References | Also see

Description

`meta funnelplot` produces funnel plots, which are used to explore the presence of small-study effects often associated with publication bias. A funnel plot is a scatterplot of study-specific effect sizes on the x axis against the measures of study precision such as standard errors and inverse standard errors on the y axis. In the absence of small-study effects, the plot should look symmetrical. `meta funnelplot` can also draw contour-enhanced funnel plots, which are useful for investigating whether the plot asymmetry can be attributed to publication bias.

Quick start

Construct a funnel plot for `meta` data, which was declared by either `meta set` or `meta esize`

```
meta funnelplot
```

Specify 1%, 5%, and 10% significance contours to produce a contour-enhanced funnel plot

```
meta funnelplot, contours(1 5 10)
```

Same as above, but base the significance contours on a one-sided lower-tailed z test, and request separate plots for each group of variable `groupvar`

```
meta funnelplot, contours(1 5 10, lower) by(groupvar)
```

Specify the inverse standard error as the precision metric on the y axis

```
meta funnelplot, metric(invse)
```

Menu

Statistics > Meta-analysis

Syntax

Construct a funnel plot

```
meta funnelplot [if] [in] [, level(#) options]
```

Construct a contour-enhanced funnel plot

```
meta funnelplot [if] [in], contours(contourspec) [options]
```

options	Description
Model	
`random[(`*remethod*`)]`	random-effects meta-analysis
`common[(`*cefemethod*`)]`	common-effect meta-analysis
`fixed[(`*cefemethod*`)]`	fixed-effects meta-analysis
Options	
`by(`*varlist*`, ...)`	construct a separate plot for each group formed by *varlist*
`metric(`*metric*`)`	specify y-axis metric; default is `metric(se)`
`n(#)`	evaluate CI lines or significance contours at # points; default is `n(300)`
`[no]metashow`	display or suppress meta settings in the output
graph_options	affect rendition of overall funnel plot

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

remethod	Description
`reml`	restricted maximum likelihood; the default
`mle`	maximum likelihood
`ebayes`	empirical Bayes
`dlaird`	DerSimonian–Laird
`sjonkman`	Sidik–Jonkman
`hedges`	Hedges
`hschmidt`	Hunter–Schmidt

cefemethod	Description
`mhaenszel`	Mantel–Haenszel
`invvariance`	inverse variance
`ivariance`	synonym for `invvariance`

graph_options	Description
ES line	
`esopts(`*line_options*`)`	affect rendition of estimated effect-size line
CI plot	
* `ciopts(`*ciopts*`)`	affect rendition of the confidence intervals
Add plots	
`addplot(`*plot*`)`	add other plots to the funnel plot
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options documented in [G-3] ***twoway_options***

*`ciopts(`*ciopts*`)` is not available for a contour-enhanced funnel plot.

Options

Main

`contours(`*contourspec*`)` specifies that a contour-enhanced funnel plot be plotted instead of the default standard funnel plot; see *Contour-enhanced funnel plots*. This option may not be combined with options `ciopts()` and `level()`.

contourspec is *numlist*[, `lower` `upper` `lines` *graph_options*]. *numlist* specifies the levels of significance (as percentages) and may contain no more than 8 integer values between 1 and 50.

`lower` and `upper` specify that the significance contours be based on one-sided lower- or upper-tailed z tests of individual effect sizes. In other words, the studies in the shaded area of a specific contour c are considered not statistically significant based on one-sided lower- or upper-tailed z tests with $\alpha = c/100$. By default, the contours correspond to the two-sided z tests.

`lines` specifies that only the contours lines be plotted. That is, no shaded regions will be displayed.

graph_options are any of the options documented in [G-3] ***area_options*** except `recast()` or, if option `lines` is specified, any of the options documented in [G-3] ***line_options***.

Model

Options `random()`, `common()`, and `fixed()` specify a meta-analysis model to use when estimating the overall effect size. For historical reasons, the default is `common(invvariance)`, regardless of the global model declared by `meta set` or `meta esize`. Specify one of these options with `meta funnelplot` to override this default. Options `random()`, `common()`, and `fixed()` may not be combined. Also see *Meta-analysis models* in [META] **Intro**.

`random` and `random(`*remethod*`)` specify that a random-effects model be assumed for meta-analysis; see *Random-effects model* in [META] **Intro**.

remethod specifies the type of estimator for the between-study variance τ^2. *remethod* is one of `reml`, `mle`, `ebayes`, `dlaird`, `sjonkman`, `hedges`, or `hschmidt`. `random` is a synonym for `random(reml)`. See *Options* in [META] **meta esize** for more information.

`common` and `common(`*cefemethod*`)` specify that a common-effect model be assumed for meta-analysis; see *Common-effect ("fixed-effect") model* in [META] **Intro**. Also see the discussion in [META] **meta data** about common-effect versus fixed-effects models.

common implies common(mhaenszel) for effect sizes lnoratio, lnrratio, and rdiff and common(invvariance) for all other effect sizes. common(mhaenszel) is supported only with effect sizes lnoratio, lnrratio, and rdiff.

cefemethod is one of mhaenszel or invvariance (synonym ivariance). See *Options* in [META] **meta esize** for more information.

fixed and fixed(*cefemethod*) specify that a fixed-effects model be assumed for meta-analysis; see *Fixed-effects model* in [META] **Intro**. Also see the discussion in [META] **meta data** about fixed-effects versus common-effect models.

fixed implies fixed(mhaenszel) for effect sizes lnoratio, lnrratio, and rdiff and fixed(invvariance) for all other effect sizes. fixed(mhaenszel) is supported only with effect sizes lnoratio, lnrratio, and rdiff.

cefemethod is one of mhaenszel or invvariance (synonym ivariance); see *Options* in [META] **meta esize** for more information.

Options

by(*varlist*[, *byopts*]) specifies that a separate plot for each group defined by *varlist* be produced. *byopts* are any of the options documented in [G-3] ***by_option***. by() is useful to explore publication bias in the presence of between-study heterogeneity induced by a set of categorical variables. These variables must then be specified in the by() option.

metric(*metric*) specifies the precision metric on the y axis. *metric* is one of se, invse, var, invvar, n, or invn. When *metric* is one of n or invn, no CIs or significance contours are plotted. The default is metric(se).

se specifies that the standard error, $\hat{\sigma}_j$, be used as the precision metric.

invse specifies that the inverse of the standard error, $1/\hat{\sigma}_j$, be used as the precision metric.

var specifies that the variance, $\hat{\sigma}^2_j$, be used as the precision metric.

invvar specifies that the inverse of the variance, $1/\hat{\sigma}^2_j$, be used as the precision metric.

n specifies that the sample size, n_j, be used as the precision metric.

invn specifies that the inverse of the sample size, $1/n_j$, be used as the precision metric.

level(#) specifies the confidence level, as a percentage, for confidence intervals. The default is as declared for the meta-analysis session; see *Declaring a confidence level for meta-analysis* in [META] **meta data**. Also see option level() in [META] **meta set**. This option may not be combined with option contours().

n(#) specifies the number of points at which to evaluate the CIs or, if option contours() is specified, significance contours. The default is n(300).

metashow and nometashow display or suppress the meta setting information. By default, this information is displayed at the top of the output. You can also specify nometashow with meta update to suppress the meta setting output for the entire meta-analysis session.

ES line

esopts(*line_options*) affects the rendition of the line that plots the estimated overall effect size; see [G-3] ***line_options***.

CI plot

`ciopts(`*ciopts*`)` affects the rendition of the (pseudo) CI lines in a funnel plot. *ciopts* are any of the options documented in [G-3] ***line_options*** and option `recast(rarea)` as described in [G-3] ***advanced_options***. This option may not be combined with option `contours()`.

Add plots

`addplot(`*plot*`)` allows adding more `graph twoway` plots to the graph; see [G-3] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G-3] ***twoway_options***. These include options for titling the graph (see [G-3] ***title_options***) and for saving the graph to disk (see [G-3] ***saving_option***).

Remarks and examples

Remarks are presented under the following headings:

Introduction
  Funnel plots
  Contour-enhanced funnel plots
Using meta funnelplot
Examples of using meta funnelplot

Introduction

A funnel plot is used to visually explore "small-study effects". The term small-study effects (Sterne, Gavaghan, and Egger 2000) is used in meta-analysis to describe the cases when the results of smaller studies differ systematically from the results of larger studies. For instance, smaller studies often report larger effect sizes than the larger studies. One of the reasons for the presence of small-study effects is publication bias, also referred to as reporting bias.

For more formal testing of small-study effects, see [META] **meta bias**. To assess the impact of publication bias on the results, see [META] **meta trimfill**.

Also see *Publication bias* of [META] **Intro** for information about publication bias.

Funnel plots

The funnel plot (Light and Pillemer 1984) is a scatterplot of the study-specific effect sizes against measures of study precision. This plot is commonly used to explore publication bias. In the absence of publication bias, the shape of the scatterplot should resemble a symmetric (inverted) funnel.

In a funnel plot, the effect sizes, $\hat{\theta}_j$'s, from individual studies are plotted on the x axis, and measures of study precision such as standard errors, $\hat{\sigma}_j$'s, or sample sizes, n_j's, are plotted on the y axis (Sterne and Harbord 2016). The line corresponding to the estimated overall effect size is also plotted on a funnel plot. In addition to standard errors and sample sizes, other choices for metrics on the y axis include inverse standard errors, $1/\hat{\sigma}_j$'s; variances, $\hat{\sigma}^2_j$'s; inverse variances, $1/\hat{\sigma}^2_j$'s; and inverse sample sizes, $1/n_j$'s. Sterne and Egger (2001) studied various metrics and found that the standard error metric performed well in many cases.

In the absence of publication bias (and between-study heterogeneity), the studies should be distributed symmetrically about the overall effect size because the sampling error is random. Also, the effect-size estimates from the smaller studies will be more variable than those from the larger studies. Thus, the scatter will be wider at the base of the plot creating, in the absence of bias, a symmetrical funnel shape or, more precisely, a symmetrical inverted funnel shape. When the statistically nonsignificant results of smaller studies are not published (and thus not included in the meta-analysis), an asymmetrical shape of the funnel plot may be observed. In this case, the estimate of the overall effect size will overestimate the true effect size. See Sterne, Becker, and Egger (2005) for details. Also see *Examples of using meta funnelplot* for examples of funnel plots.

Sutton (2009) states that when the y-axis metric is one of standard error, variance, or their inverses, a $(1 - \alpha) \times 100\%$ CI can be formed around the overall estimate. This CI can provide more formal interpretation of the plot. But the author suggests that caution be used when interpreting these CIs because they are formed around the estimate of the overall effect size that may be affected by publication bias. This is one of the reasons why the funnel-plot CIs are often referred to as pseudo CIs.

In general, there may be many reasons for an asymmetric funnel plot such as the choice of the plotted effect size (Sterne et al. 2011), the presence of a moderator correlated with the study effect and study size (Peters et al. 2008), or simply chance. One of the more common reasons, however, is the presence of substantial between-study heterogeneity (Sterne, Gavaghan, and Egger 2000).

The between-study heterogeneity, if present, must be addressed before the exploration of publication bias. For instance, if there are study-level covariates that explain the differences between the studies, their influence can distort a funnel plot if they are not accounted for in the main meta-analysis (Sutton 2009). Suppose that during our subgroup meta-analysis (see option `subgroup()` in [META] **meta summarize**), we identified a categorical variable that explains most of the heterogeneity between the studies. The exploration of the publication bias should then be performed separately for each group. That is, a separate funnel plot should be constructed for each subgroup. In the case of a continuous variable, some authors suggest constructing a funnel plot based on the residuals, $\hat{\theta}_j - \mathbf{x}_j\widehat{\boldsymbol{\beta}}$, on the x axis against their standard errors on the y axis, where the residuals are obtained from a meta-regression that uses this continuous variable as the moderator; see [META] **meta regress** and [META] **meta regress postestimation**.

Contour-enhanced funnel plots

Peters et al. (2008) (also see Palmer et al. [2016]) suggest that contour lines of statistical significance (or significance contours) be added to the funnel plot. These "contour-enhanced" funnel plots are useful for determining whether the funnel-plot asymmetry is potentially caused by publication bias or is perhaps due to other reasons. The contour lines that correspond to certain significance levels ($\alpha = 0.01$, 0.05, 0.1, etc.) of tests of zero effect sizes are overlaid on the funnel plot. Publication bias is suspect if there are studies, especially smaller studies, that are missing in the nonsignificant regions. Otherwise, other reasons may explain the presence of the funnel-plot asymmetry.

Using meta funnelplot

`meta funnelplot` produces funnel plots. By default, it plots effect sizes against the standard errors, but you can specify a different metric in the `metric()` option. For historical reasons, the default meta-analysis model is the common-effect model with the inverse-variance estimation method. You can specify one of `random()`, `common()`, or `fixed()` to use a different model or method.

By default, the CIs are displayed, which correspond to the confidence level as declared by `meta set` or `meta esize`. You can specify a different level in the `level()` option. You can also specify the `ciopts()` option to modify the default look of the CI lines.

Instead of the CIs, you can request a contour-enhanced funnel plot by specifying the desired levels of significance (as a percentage) in the `contours()` option. The default significance contours are based on two-sided significance tests of individual effect sizes. You can use `lower` or `upper` within `contours()` to specify that the significance contours be based on the corresponding one-sided tests. You can also specify `lines` within `contours()` to recast the contours to be displayed as lines instead of shaded area plots.

You can use the `by(`*varlist*`)` option to produce separate funnel plots for each group defined by *varlist*. This option is useful after a subgroup analysis (see option `subgroup()` in [META] **meta summarize**). If a subgroup analysis identified a categorical variable that explains some of the between-study variability, that variable must be specified in the `by()` option when using `meta funnelplot` to explore publication bias.

You can also change the default look of the effect-size line by specifying the `esopts()` option.

In the next section, we describe some of the uses of `meta funnelplot`.

Examples of using meta funnelplot

Recall the NSAIDS data of 37 trials on the effectiveness and safety of topical nonsteroidal anti-inflammatory drugs for acute pain described in *Effectiveness of nonsteroidal anti-inflammatory drugs (nsaids.dta)* of [META] **meta**. In this section, we use its declared version and focus on the demonstration of various options of `meta funnelplot` and explanation of its output.

```
. use https://www.stata-press.com/data/r18/nsaidsset
(Effectiveness of nonsteroidal anti-inflammatory drugs; set with -meta esize-)

. meta query, short
-> meta esize nstreat nftreat nscontrol nfcontrol

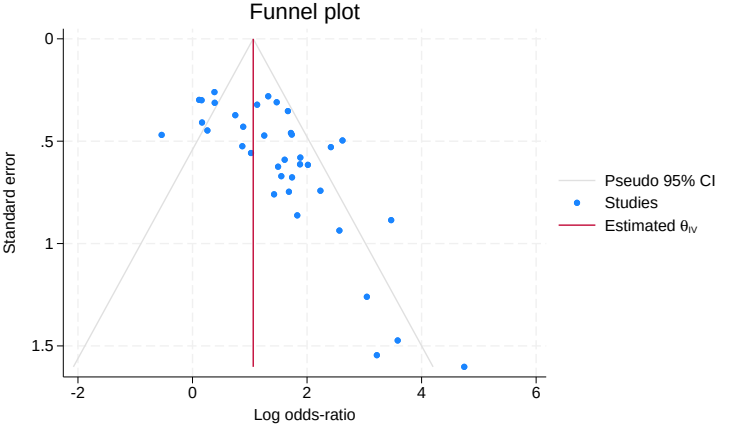
  Effect-size label: Log odds-ratio
   Effect-size type: lnoratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Random effects
             Method: REML
```

`meta query, short` reminds us about the main settings of the declaration step. Our data were declared by using `meta esize` with variables `nstreat`, `nftreat`, `nscontrol`, and `nfcontrol` representing the summary data from 2×2 tables, which record the numbers of successes and failures in the treatment and control arms. The computed effect sizes are log odds-ratios; their values and standard errors are stored in the respective system variables `_meta_es` and `_meta_se`. The declared meta-analysis model is the default random-effects model with the REML estimation method.

Example 1: Funnel plot

To produce a funnel plot for the declared NSAIDS data, we simply type

```
. meta funnelplot
  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Common effect
             Method: Inverse-variance
```



The scatterplot of log odds-ratios against their standard errors is produced. The estimated effect-size line and the corresponding pseudo 95% CIs are also plotted. The funnel plot is clearly asymmetric with smaller, less precise studies—studies with larger standard errors—reporting larger effect sizes than the more precise studies. This may suggest the presence of publication bias. The plotted pseudo CI lines are not genuine CI limits, but they provide some insight into the spread of the observed effect sizes about the estimate of the overall effect size. In the absence of publication bias and heterogeneity, we would expect the majority of studies to be randomly scattered within the CI region resembling an inverted funnel shape.

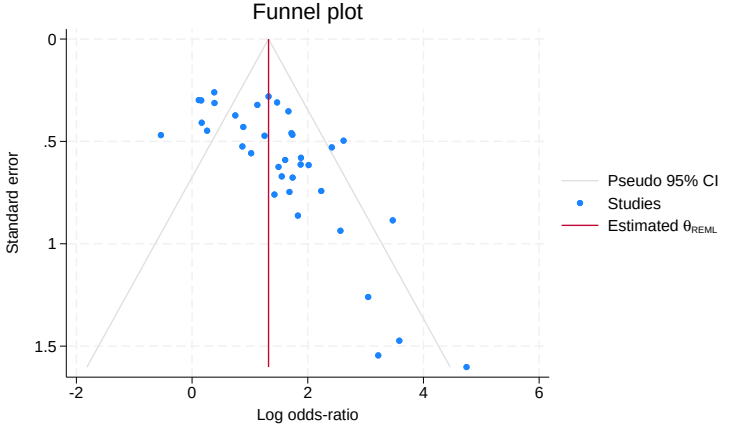
Notice that although the declared meta-analysis model was the random-effects model with the REML estimation method, the default model used by `meta funnelplot` was the common-effect model with the inverse-variance method, as is indicated in the brief output of meta settings reported by the command. This is the model traditionally used with funnel plots in the literature. The reported model and method are used to compute the estimate of the overall effect size, the overall log odds-ratio in our example, which is depicted by the reference (red) effect-size line.

Example 2: Random-effects REML model

In example 1, we pointed out that, for historical reasons, `meta funnelplot` uses the common-effect model with the inverse-variance method by default instead of the declared ones.

If desired, we can obtain the results assuming the declared random-effects model with the REML estimation method by specifying the `random` option.

```
. meta funnelplot, random
  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Random effects
             Method: REML
```



From the output of meta settings, a random-effects model with the REML method is now used to estimate the overall log odds-ratio. Our conclusion about the existence of potential publication bias remains the same.

For brevity, let's suppress the meta setting information from the output of `meta funnelplot` for the rest of the analysis. We can do this by specifying the `nometashow` option with `meta update` (see [META] **meta update**).

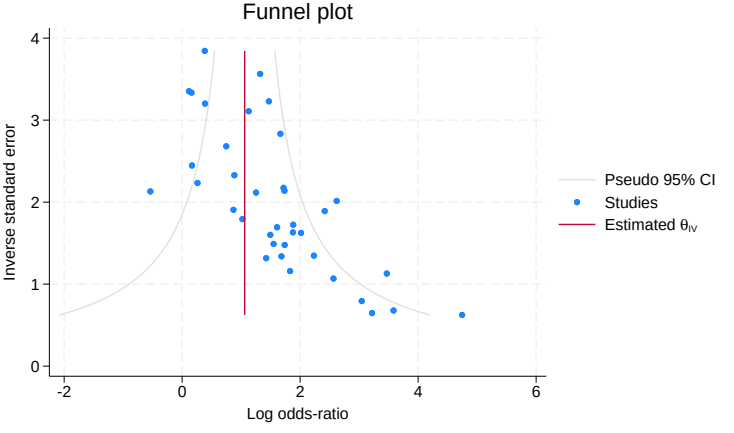
```
. quietly meta update, nometashow
```

▷ Example 3: Inverse-standard-error metric

Continuing with example 1, we can use a different precision metric on the y axis. Using different metrics may reveal different characteristics of the observed studies. For example, for the standard error metric, more weight is given to the smaller studies, which are more susceptible to publication bias. For the inverse-standard-error metric, more weight is given to the larger studies.

Below, we specify the inverse-standard-error metric by using the `metric(invse)` option.

```
. meta funnelplot, metric(invse)
```



Because our y-axis metric is now $1/\hat{\sigma}_j$, the shape of the plotted CI lines is a hyperbola. The interpretation, however, is similar. We still want to see the majority of studies be concentrated within the regions defined by this hyperbola.

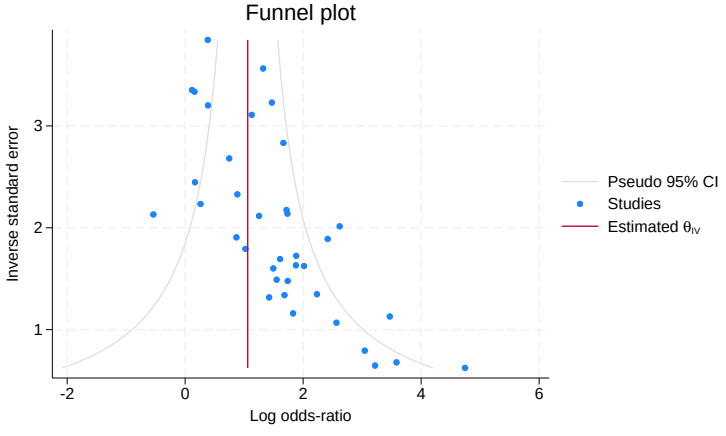
In this metric, the focus is on larger studies with the smaller studies compressed at the bottom. We can see that the asymmetry does not appear to be present for larger studies with, say, $1/\hat{\sigma}_j > 2$. But the asymmetry is present for the smaller studies.

◁

Example 4: The ylabel() option

Continuing with example 3, we can improve the look of the plot by restricting the y-axis values to the observed range. We can do this by specifying the `ylabel(1 2 3)` option:

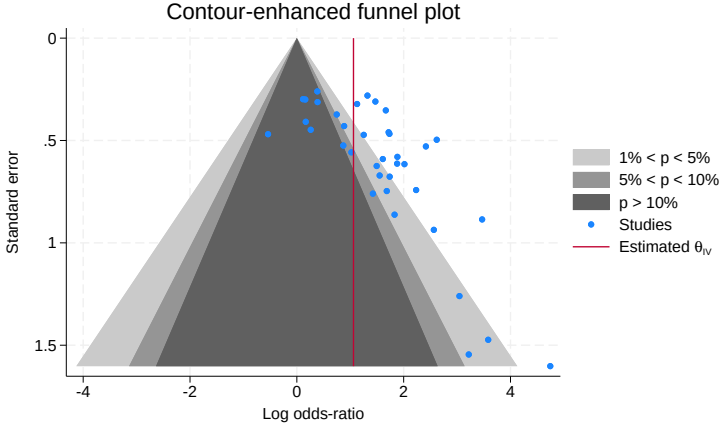
```
. meta funnelplot, metric(invse) ylabel(1 2 3)
```



Example 5: Contour-enhanced funnel plot

In example 12 of [META] **meta**, we considered contour-enhanced funnel plots for the NSAIDS data to help us identify whether the asymmetry observed in example 1 is because of publication bias or perhaps some other reasons. Let's obtain the 1%, 5%, and 10% contour-enhanced funnel plots by specifying the `contours(1 5 10)` option:

```
. meta funnelplot, contours(1 5 10)
```



The plotted contour regions define the regions of statistical significance (or nonsignificance) of the individual effect sizes θ_j. That is, if you consider the significance test of $H_0\colon \theta_j = 0$ for a study j, the contour regions represent the critical regions of such tests for all studies at the specified significance

level such as 1%, 5%, and 10% levels in our example. Thus, if a study falls outside a certain region, we have statistical evidence to reject the null hypothesis of no effect at the significance level corresponding to that region.

In our example, for studies in the white region, the null hypothesis of no effect can be rejected at the 1% significance level. That is, the significance tests for these studies would have p-values less than 0.01 or 1%. For studies in the light-gray region, the p-values would be between 1% and 5%. For studies in the darker-gray region, the p-values would be between 5% and 10%. And for studies in the darkest-gray region, the p-values would be larger than 10%.

The plot clearly shows that almost all smaller studies report a statistically significant result, favoring the treatment, either at the 1% or 5% level. On the other hand, some of the larger (more precise) studies (in the darkest-gray region) report nonsignificant results. The hypothetical missing studies—the studies that would make the scatterplot look symmetric with respect to the solid red vertical line—appear to fall in the darkest-gray region corresponding to a p-value of more than 10%. Because we are "missing" small studies in a region of statistical nonsignificance, this suggests that the observed asymmetry in the funnel plot is likely because of publication bias.

Also see example 14 of [META] **meta**.

◁

▷ Example 6: The addplot() option

Continuing with example 5, let's use the `addplot()` option to add the pseudo 95% CIs around the effect-size line to our contour-enhanced plot. These CIs are computed as follows: $\hat{\theta}_{\text{IV}} \pm 1.96 \times y$, where y represents a standard error of $\hat{\theta}_{\text{IV}}$, varying within the range of observed standard errors. Here 1.96 corresponds to the $z_{0.975}$ critical value.

We first obtain the estimated value of the overall effect size, which is available in the `r(theta)` scalar from the previous run of `meta funnelplot`. We store it in the `theta` scalar.

```
. display r(theta)
1.0604829
. scalar theta = r(theta)
```

The estimate of the overall effect size may also be obtained from `meta summarize` (see [META] **meta summarize**):

```
. meta summarize, common(invvariance) nostudies
Meta-analysis summary                           Number of studies =     37
Common-effect model
Method: Inverse-variance
        theta: Overall Log odds-ratio
```

	Estimate	Std. err.	z	P>|z|	[95% conf.	interval]
theta	1.060483	.0758709	13.98	0.000	.9117788	1.209187

```
. display r(theta)
1.0604829
```

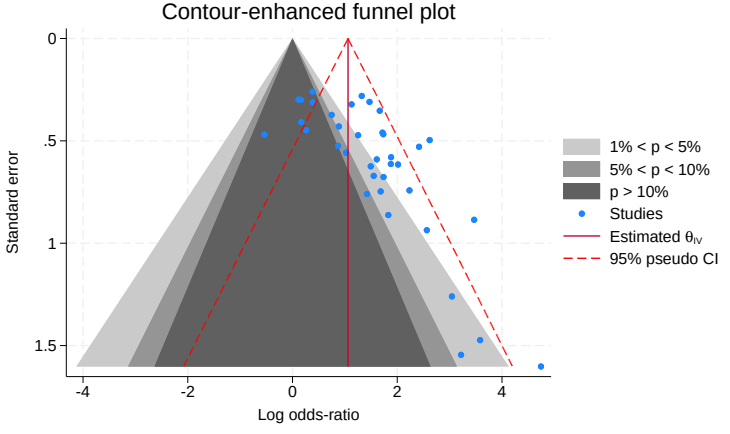
The CI lines can be constructed by using twoway's function command (see [G-2] **graph twoway function**):

```
. twoway function theta-1.96*x, horizontal range(0 1.6) ||
>        function theta+1.96*x, horizontal range(0 1.6)
```

In the above, x plays the role of y in the earlier expression for CIs. We used the horizontal option to interchange the roles of y and x in the function because $\hat{\theta}_j$ appears on the x axis and standard errors on the y axis in the funnel plot. We also specified the range() option so that the range of the plotted function matches the observed range for the standard errors.

We use the above specification in the addplot() option with meta funnelplot. Because addplot() implies a twoway plot, we can omit twoway within addplot(). We also specify several other options to improve the look of the graph, which we describe later.

```
. local opts horizontal range(0 1.6) lpattern(dash) lcolor("red")
> legend(order(1 2 3 4 5 6) label(6 "95% pseudo CI"))
. meta funnel, contours(1 5 10)
> addplot(function theta-1.96*x, 'opts' || function theta+1.96*x, 'opts')
```



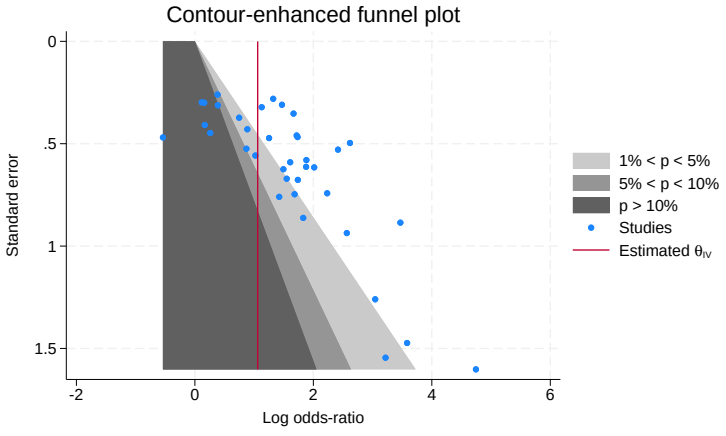
We changed the color and pattern of the CI lines by using options lpattern() and lcolor(). We used legend()'s suboption order() to display only the first six keys for the legend to avoid duplicate keys for the two CI plots. We also provided a more descriptive label for the CI legend key. Also, to be more precise, we could have replaced 1.96 in the above with invnormal(.975), which computes the corresponding quantile of the standard normal distribution.

◁

▷ Example 7: Upper one-sided significance contours

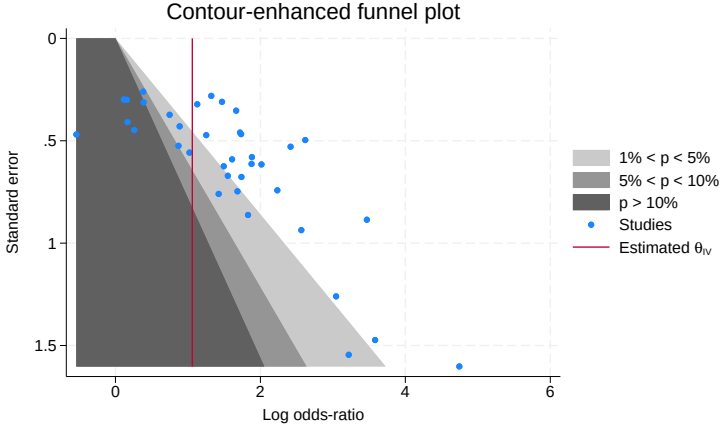
Continuing with example 5, let's produce a contour-enhanced funnel plot based on one-sided significance tests instead of the default two-sided tests. We can specify lower or upper within contours() to produce a funnel plot with contours based on lower or upper one-sided tests. Below, we specify the upper suboption:

```
. meta funnelplot, contours(1 5 10, upper)
```



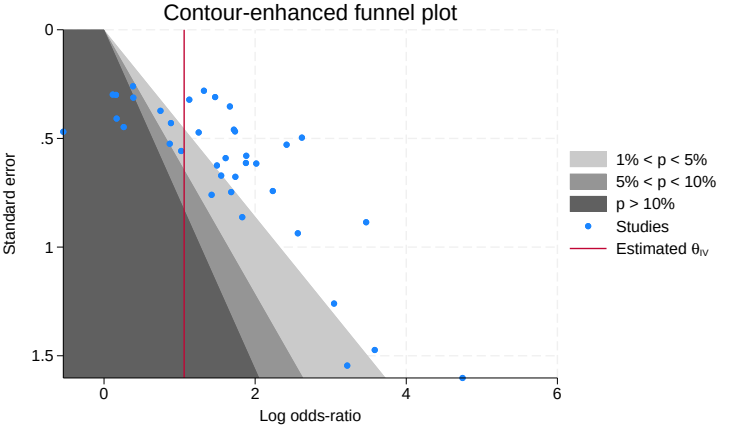
The interpretation of a one-sided contour-enhanced funnel plot is similar to that of a two-sided one. Studies that fall in the upper-tailed region (the white region to the right) are statistically significant at the 1% level based on a one-sided upper-tailed test. The white space on the left is uninformative, and we can suppress it by disallowing the x axis to extend to -2. This may be done by specifying `xlabel(0(2)6)` (see [G-3] ***axis_label_options***).

```
. meta funnelplot, contours(1 5 10, upper) xlabel(0(2)6)
```



If we want to suppress all the extra white space around the plot, we can specify `plotregion(margin(zero))` (see [G-3] ***region_options***).

```
. meta funnelplot, contours(1 5 10, upper) xlabel(0(2)6)
> plotregion(margin(zero))
```



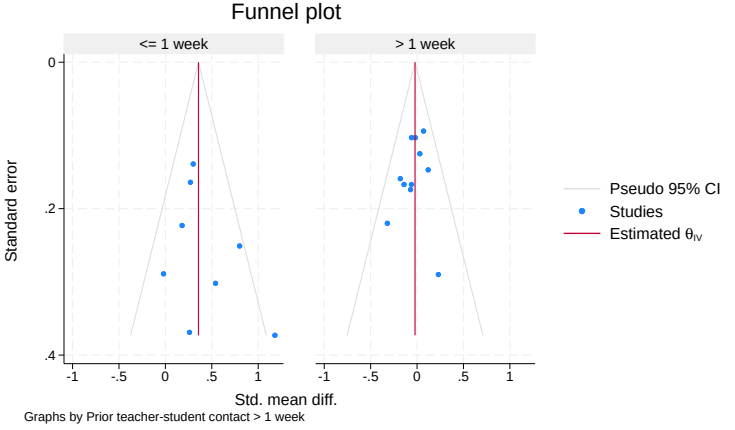
Example 8: Group-specific funnel plots: option by()

In example 9 of [META] **meta summarize**, we performed subgroup analysis for the pupil IQ data to account for the between-study heterogeneity induced by the binary covariate `week1`, which records whether teachers had prior contact with students for more than 1 week or for 1 week or less. See *Effects of teacher expectancy on pupil IQ (pupiliq.dta)* of [META] **meta** for details about the data.

Let's check for publication bias in these data, accounting for the between-study heterogeneity. We can do this by looking at a funnel plot separately for each category of `week1`. We will use `meta funnel`'s option `by()` to produce such plots.

We use a previously declared version of the dataset.

```
. use https://www.stata-press.com/data/r18/pupiliqset, clear
(Effects of teacher expectancy on pupil IQ; set with -meta set-)
. meta funnelplot, by(week1)
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
              Model: Common effect
             Method: Inverse-variance
```



The above graph shows funnel plots of the subgroups for prior contact of one week or less and more than one week, respectively. These funnels are centered on different effect-size values, but there is little evidence of asymmetry in either plot. We should be careful with our interpretation, however, because we have only a few studies in each plot.

◁

Stored results

`meta funnelplot` stores the following in `r()`:

Scalars

`r(theta)`	estimated overall effect size
`r(xmin)`	minimum abscissa of scatter points
`r(xmax)`	maximum abscissa of scatter points
`r(ymin)`	minimum ordinate of scatter points
`r(ymax)`	maximum ordinate of scatter points

Macros

`r(model)`	meta-analysis model
`r(method)`	meta-analysis estimation method
`r(metric)`	metric for the y axis
`r(contours)`	significance levels of contours

Methods and formulas

`meta funnel` produces a scatterplot of individual effect sizes, $\hat{\theta}_j$'s, on the x axis and measures of study precision on the y axis. The supported measures of precision are the standard errors, $\hat{\sigma}_j$'s (default); inverse standard errors, $1/\hat{\sigma}_j$'s; variances, $\hat{\sigma}_j^2$'s; inverse variances, $1/\hat{\sigma}_j^2$'s; sample sizes, n_j's; and inverse sample sizes, $1/n_j$'s.

The effect-size reference (solid red) line is plotted at the estimate of the overall effect size. By default, the overall effect size is estimated assuming a common-effect model with the inverse-variance method, but this can be changed by specifying one of `random()`, `common()`, or `fixed()`.

The pseudo $(1-\alpha)\times 100\%$ CI lines, plotted by default, are constructed as $\hat{\theta} \pm z_{1-\alpha/2} f(y)$, where $\hat{\theta}$ is the estimate of the overall effect size, $z_{1-\alpha/2}$ is the $(1-\alpha/2)$th quantile of the standard normal distribution, and $f(y)$ plays the role of the varying standard error of $\hat{\theta}$ as a function of the y-axis values and depends on the chosen metric. When standard errors, $\hat{\sigma}_j$'s, are plotted on the y axis,

$$f(y) = y$$

and the CI curves form straight lines. When variances, $\hat{\sigma}_j^2$'s, are plotted on the y axis,

$$f(y) = \sqrt{y}$$

and the CI curves form a parabola. When inverse standard deviations, $1/\hat{\sigma}_j$'s, or inverse variances, $1/\hat{\sigma}_j^2$, are plotted on the y axis

$$f(y) = \frac{1}{y} \qquad \text{and} \qquad f(y) = \frac{1}{\sqrt{y}}$$

and the CI curves form a hyperbola.

When the `contours()` option is specified, the contour region corresponding to a significance level α (specified as a percentage in `contours()`) for a two-sided test is defined as the set of points (x, y),

$$\left\{(x,y) : \left|\frac{x}{f(y)}\right| \leq z_{1-\alpha/2}\right\}$$

where $f(y)$ depends on the chosen metric and is defined as before. For upper one-sided tests, the contour region is defined as

$$\left\{(x,y) : \frac{x}{f(y)} \geq z_{1-\alpha}\right\}$$

and for lower one-sided tests, it is defined as

$$\left\{(x,y) : \frac{x}{f(y)} \leq z_{\alpha}\right\}$$

The `n(#)` option specifies how many evaluation points are used to construct the CI lines or, when the `contours()` option is specified, the significance contours. By default, 300 points are used for y for CI lines and for each of x and y for contours.

References

Light, R. J., and D. B. Pillemer. 1984. *Summing Up: The Science of Reviewing Research*. Cambridge, MA: Harvard University Press. https://doi.org/10.2307/j.ctvk12px9.

Palmer, T. M., J. L. Peters, A. J. Sutton, and S. G. Moreno. 2016. "Contour-enhanced funnel plots for meta-analysis". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 139–152. 2nd ed. College Station, TX: Stata Press.

Peters, J. L., A. J. Sutton, D. R. Jones, K. R. Abrams, and L. Rushton. 2008. Contour-enhanced meta-analysis funnel plots help distinguish publication bias from other causes of asymmetry. *Journal of Clinical Epidemiology* 61: 991–996. https://doi.org/10.1016/j.jclinepi.2007.11.010.

Sterne, J. A. C., B. J. Becker, and M. Egger. 2005. "The funnel plot". In *Publication Bias in Meta-Analysis: Prevention, Assessment and Adjustments*, edited by H. R. Rothstein, A. J. Sutton, and M. Borenstein, 73–98. Chichester, UK: Wiley. https://doi.org/10.1002/0470870168.ch5.

Sterne, J. A. C., and M. Egger. 2001. Funnel plots for detecting bias in meta-analysis: Guidelines on choice of axis. *Journal of Clinical Epidemiology* 54: 1046–1055. https://doi.org/10.1016/s0895-4356(01)00377-8.

Sterne, J. A. C., D. Gavaghan, and M. Egger. 2000. Publication and related bias in meta-analysis: Power of statistical tests and prevalence in the literature. *Journal of Clinical Epidemiology* 53: 1119–1129. https://doi.org/10.1016/S0895-4356(00)00242-0.

Sterne, J. A. C., and R. M. Harbord. 2016. "Funnel plots in meta-analysis". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 124–138. 2nd ed. College Station, TX: Stata Press.

Sterne, J. A. C., A. J. Sutton, J. P. A. Ioannidis, N. Terrin, D. R. Jones, J. Lau, J. R. Carpenter, G. Rücker, R. M. Harbord, C. H. Schmid, J. Tetzlaff, J. J. Deeks, J. L. Peters, P. Macaskill, G. Schwarzer, S. Duval, D. G. Altman, D. Moher, and J. P. T. Higgins. 2011. Recommendations for examining and interpreting funnel plot asymmetry in meta-analyses of randomised controlled trials. *BMJ* 343: d4002. https://doi.org/10.1136/bmj.d4002.

Sutton, A. J. 2009. "Publication bias". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 435–452. 2nd ed. New York: Russell Sage Foundation.

Also see

[META] **meta bias** — Tests for small-study effects in meta-analysis

[META] **meta data** — Declare meta-analysis data

[META] **meta trimfill** — Nonparametric trim-and-fill analysis of publication bias

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta bias — Tests for small-study effects in meta-analysis

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
References | Also see

Description

`meta bias` performs tests for the presence of small-study effects in a meta-analysis, also known as tests for funnel-plot asymmetry and publication-bias tests. Three regression-based tests and a nonparametric rank correlation test are available. For regression-based tests, you can include moderators to account for potential between-study heterogeneity.

Quick start

Test for small-study effects by using the Egger regression-based test

```
meta bias, egger
```

Same as above, but include a moderator `x1` to account for between-study heterogeneity induced by `x1`

```
meta bias x1, egger
```

Same as above, but assume a random-effects model with the empirical Bayes method for estimating τ^2 in the regression-based test

```
meta bias x1, egger random(ebayes)
```

With log risk-ratios, test for small-study effects by using the Harbord regression-based test with moderators `x1` and `x2` to account for between-study heterogeneity

```
meta bias x1 i.x2, harbord
```

With log odds-ratios, test for small-study effects by using the Peters regression-based test and assuming a common-effect model

```
meta bias, peters common
```

Menu

Statistics > Meta-analysis

Syntax

Regression-based tests for small-study effects

Test using meta-analysis model as declared with meta set or meta esize

```
meta bias [moderators] [if] [in], regtest [modelopts]
```

Random-effects meta-analysis model

```
meta bias [moderators] [if] [in], regtest random[(remethod)]
    [se(seadj) options]
```

Common-effect meta-analysis model

```
meta bias [if] [in], regtest common [options]
```

Fixed-effects meta-analysis model

```
meta bias [moderators] [if] [in], regtest fixed [multiplicative options]
```

Traditional test

```
meta bias [if] [in], regtest traditional [options]
```

Nonparametric rank correlation test for small-study effects

```
meta bias [if] [in], begg [ [no]metashow detail]
```

regtest	Description
egger	Egger's test
harbord	Harbord's test
peters	Peters's test

modelopts is any option relevant for the declared model.

remethod	Description
reml	restricted maximum likelihood; the default
mle	maximum likelihood
ebayes	empirical Bayes
dlaird	DerSimonian–Laird
sjonkman	Sidik–Jonkman
hedges	Hedges
hschmidt	Hunter–Schmidt

options	Description
Main	
`tdistribution`	report t test instead of z test
[`no`]`metashow`	display or suppress meta settings in the output
`detail`	display intermediate estimation results
Maximization	
maximize_options	control the maximization process of the between-study variance

moderators may contain factor variables; see **[U] 11.4.3 Factor variables**.

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

Options

Main

One of `egger`, `harbord`, `peters`, or `begg` (or their synonyms) must be specified. In addition to the traditional versions of the regression-based tests, their random-effects versions and extensions to allow for moderators are also available.

`egger` (synonym `esphillips`) specifies that the regression-based test of Egger, Davey Smith, and Phillips (1997) be performed. This test is known as the Egger test in the literature. This is the test of the slope in a weighted regression of the effect size, `_meta_es`, on its standard error, `_meta_se`, optionally adjusted for *moderators*. This test tends to have an inflated type I error rate for two-sample binary data.

`harbord` (synonym `hesterne`) specifies that the regression-based test of Harbord, Egger, and Sterne (2006) be performed. This test is known as the Harbord test. This is the test of the slope in a weighted regression of Z_j/V_j on $1/\sqrt{V_j}$, optionally adjusting for *moderators*, where Z_j is the score of the likelihood function and V_j is the score variance. This test is used for two-sample binary data with effect sizes log odds-ratio and log risk-ratio. It was designed to reduce the correlation between the effect-size estimates and their corresponding standard errors, which is inherent to the Egger test with two-sample binary data.

`peters` (synonym `petersetal`) specifies that the regression-based test of Peters et al. (2006) be performed. This test is known as the Peters test in the literature. This is the test of the slope in a weighted regression of the effect size, `_meta_es`, on the inverse sample size, $1/n_j$, optionally adjusted for *moderators*. The Peters test is used with two-sample binary data for log odds-ratios. Because it regresses effect sizes on inverse sample sizes, they are independent by construction.

`begg` (synonym `bmazumdar`) specifies that the nonparametric rank correlation test of Begg and Mazumdar (1994) be performed. This is not a regression-based test, so only options `metashow`, `nometashow`, and `detail` are allowed with it. This test is known as the Begg test in the literature. This test is no longer recommended in the literature and provided for completeness.

Options `random()`, `common`, and `fixed`, when specified with `meta bias` for regression-based tests, temporarily override the global model declared by `meta set` or `meta esize` during the computation. Options `random()`, `common`, and `fixed` may not be combined. If these options are omitted, the declared meta-analysis model is assumed; see *Declaring a meta-analysis model* in [META] **meta data**. Also see *Meta-analysis models* in [META] **Intro**.

`random` and `random(`*remethod*`)` specify that a random-effects model be assumed for regression-based test; see *Random-effects model* in [META] **Intro**.

remethod specifies the type of estimator for the between-study variance τ^2. *remethod* is one of `reml`, `mle`, `ebayes`, `dlaird`, `sjonkman`, `hedges`, or `hschmidt`. `random` is a synonym for `random(reml)`. See *Options* in [META] **meta esize** for more information.

`common` specifies that a common-effect model be assumed for regression-based test; see *Common-effect ("fixed-effect") model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about common-effect versus fixed-effects models. `common` is not allowed in the presence of moderators.

`fixed` specifies that a fixed-effects model be assumed for regression-based test; see *Fixed-effects model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about fixed-effects versus common-effect models.

`se(`*seadj*`)` specifies that the adjustment *seadj* be applied to the standard errors of the coefficients. Additionally, the tests of significance of the coefficients are based on a Student's t distribution instead of the normal distribution. `se()` is allowed only with random-effects models.

seadj is `khartung[ , truncated]`. Adjustment `khartung` specifies that the Knapp–Hartung adjustment (Hartung and Knapp 2001a, 2001b; Knapp and Hartung 2003), also known as the Sidik–Jonkman adjustment (Sidik and Jonkman 2002), be applied to the standard errors of the coefficients. `hknapp` and `sjonkman` are synonyms for `khartung`. `truncated` specifies that the truncated Knapp–Hartung adjustment (Knapp and Hartung 2003), also known as the modified Knapp–Hartung adjustment, be used.

`traditional` specifies that the traditional version of the selected regression-based test be performed. This option is equivalent to specifying options `fixed`, `multiplicative`, and `tdistribution`. It may not be specified with *moderators*.

`multiplicative` performs a fixed-effects regression-based test that accounts for residual heterogeneity by including a multiplicative variance parameter ϕ. ϕ is referred to as an "(over)dispersion parameter". See *Introduction* in [META] **meta regress** for details.

`tdistribution` reports a t test instead of a z test. This option may not be combined with option `se()`.

`metashow` and `nometashow` display or suppress the meta setting information. By default, this information is displayed at the top of the output. You can also specify `nometashow` with `meta update` to suppress the meta setting output for the entire meta-analysis session.

`detail` specifies that intermediate estimation results be displayed. For regression-based tests, the results from the regression estimation will be displayed. For the nonparametric test, the results from `ktau` ([R] **spearman**) will be displayed.

Maximization

maximize_options: `iterate(#)`, `tolerance(#)`, `nrtolerance(#)`, `nonrtolerance` (see [R] **Maximize**), `from(#)`, and `showtrace`. These options control the iterative estimation of the between-study variance parameter, τ^2, with random-effects methods `reml`, `mle`, and `ebayes`. These options are seldom used.

`from(#)` specifies the initial value for τ^2 during estimation. By default, the initial value for τ^2 is the noniterative Hedges estimator.

`showtrace` displays the iteration log that contains the estimated parameter τ^2, its relative difference with the value from the previous iteration, and the scaled gradient.

Remarks and examples

Remarks are presented under the following headings:

Introduction
Using meta bias
Examples of using meta bias

Introduction

As we discussed in *Introduction* of [META] **meta funnelplot**, there is a tendency for smaller studies to report different, often larger, effect sizes than the larger studies. There are various reasons that explain this tendency, but the two more common ones are between-study heterogeneity and publication bias. We covered the between-study heterogeneity in [META] **meta summarize** and [META] **meta regress**. Here we focus on publication bias.

Publication bias often arises when the decision of whether to publish a study depends on the statistical significance of the results of the study. Typically, nonsignificant results from small studies have a tendency of not getting published. See *Publication bias* of [META] **Intro** for details.

The funnel plot ([META] **meta funnelplot**) is commonly used to investigate publication bias or, more generally, small-study effects in meta-analysis. The presence of asymmetry in the funnel plot may indicate the presence of publication bias. Graphical evaluation of funnel plots is useful for data exploration but may be subjective when detecting the asymmetry. Thus, a more formal evaluation of funnel-plot asymmetry is desired. Statistical tests were developed for detecting the asymmetry in a funnel plot; they are often called tests for funnel-plot asymmetry. They are also sometimes referred to as tests of publication bias, but this terminology may be misleading because the presence of a funnel-plot asymmetry is not always due to publication bias (for example, Sterne et al. [2011]). Thus, we prefer a more generic term—tests for small-study effects—suggested by Sterne, Gavaghan, and Egger (2000).

There are two types of tests for small-study effects: regression-based tests and a nonparametric rank-based test. The main idea behind these tests is to determine whether there is a statistically significant association between the effect sizes and their measures of precision such as effect-size standard errors.

The Egger regression-based test (Egger et al. 1997) performs a weighted linear regression of the effect sizes, $\hat{\theta}_j$'s, on their standard errors, $\hat{\sigma}_j$'s, weighted by the precision, $1/\hat{\sigma}_j$'s. The test for the zero slope in that regression provides a formal test for small-study effects. In some cases, such as in the presence of a large true effect or with two-sample binary data, the Egger test tends to have an inflated type I error (for example, Harbord, Harris, and Sterne [2016]). Two alternative tests, the Harbord test and the Peters test, were proposed to alleviate the type I error problem in those cases.

The Harbord regression-based test (Harbord, Egger, and Sterne 2006) corresponds to the zero-slope test in a weighted regression of Z_j/V_j's on $1/\sqrt{V_j}$'s, where Z_j is the score of the likelihood function and V_j is the score variance. The Peters regression-based test (Peters et al. 2006) corresponds to the zero-slope test in a weighted regression of the effect sizes, $\hat{\theta}_j$'s, on the respective inverse sample sizes, $1/n_j$'s. With two-sample binary data, these tests tend to perform better than the Egger test in terms of the type I error while maintaining similar power.

The rank correlation Begg test (Begg and Mazumdar 1994) tests whether Kendall's rank correlation between the effect sizes and their variances equals zero. The regression-based tests tend to perform better in terms of type I error than the rank correlation test. This test is provided mainly for completeness.

See Harbord, Harris, and Sterne (2016) and Steichen (2016) for more details about these tests.

As we discussed in [META] **meta funnelplot**, the presence of between-study heterogeneity may affect the symmetry of a funnel plot. Thus, any statistical method based on the funnel plot will also be affected (Sutton 2009). To account for the between-study heterogeneity, the regression-based tests can be extended to incorporate moderators that may help explain the heterogeneity (Sterne and Egger 2005).

The traditional version of the regression-based tests used a multiplicative fixed-effects meta-regression to account for residual heterogeneity (see *Introduction* of [META] **meta regress**). In addition to adjusting for moderators, a random-effects meta-regression is considered a better alternative to account for residual heterogeneity.

Ioannidis and Trikalinos (2007) provide the following recommendations for when it is appropriate to use small-study tests: a) the number of studies should be greater than 10; b) there should be at least one study with a statistically significant result; c) there should be no significant heterogeneity ($I^2 < 50\%$); and d) the ratio of the maximum to minimum variances across studies should be larger than 4; that is, $\max\left(\{\hat{\sigma}_j^2\}_{j=1}^K\right) / \min\left(\{\hat{\sigma}_j^2\}_{i=1}^K\right) > 4$. If a) is violated, the tests may have low power. If c) is violated, the asymmetry of the funnel plot may be induced by between-study heterogeneity rather than publication bias. If d) is violated, the funnel plot will look more like a horizontal line than an inverted funnel, and the funnel-asymmetry tests will have an inflated type I error. Also see Sterne et al. (2011) for details.

The results of the tests of small-study effects should be interpreted with caution. In the presence of small-study effects, apart from publication bias, other reasons should also be explored to explain the presence of small-study effects. If small-study effects are not detected by a test, their existence should not be ruled out because the tests tend to have low power.

Also see [META] **meta trimfill** for assessing the impact of publication bias on the results.

Using meta bias

`meta bias` performs tests for small-study effects. These tests are also known as the tests for funnel-plot asymmetry and tests for publication bias. You can choose from three regression-based tests: the Egger test (option `egger`), the Harbord test for two-sample binary data with effect sizes log odds-ratio and log risk-ratio (option `harbord`), and the Peters test for log odds-ratios (option `peters`). You can also perform the Begg nonparametric rank correlation test (option `begg`), but this test is no longer recommended in the meta-analysis literature.

Next, we describe the features that are relevant only to the regression-based tests. These tests are based on meta-regression of effect sizes and their measures of precision.

The default meta-analysis model (and method) are as declared by `meta set` or `meta esize`; see *Declaring a meta-analysis model* in [META] **meta data**. You can change the defaults by specifying one of options `random()`, `common()`, or `fixed()`.

Because the regression-based tests use meta-regression, many of the options of `meta regress` (see [META] **meta regress**) apply to `meta bias` as well. For example, you can specify that a multiplicative meta-regression be used by the test with option `multiplicative`. And you can specify to use the t test instead of a z test for inference with option `tdistribution`.

The regression-based tests support the `traditional` option, which specifies that the tests be performed as originally published. This option is a shortcut for `fixed`, `multiplicative`, and `tdistribution`.

To account for between-study heterogeneity when checking for publication bias, you can specify moderators with the regression-based tests.

Examples of using meta bias

Recall the pupil IQ data (Raudenbush and Bryk 1985; Raudenbush 1984) described in *Effects of teacher expectancy on pupil IQ (pupiliq.dta)* of [META] **meta**. Here we will use its declared version and will focus on the demonstration of various options of `meta bias` and explanation of its output.

```
. use https://www.stata-press.com/data/r18/pupiliqset
(Effects of teacher expectancy on pupil IQ; set with -meta set-)
. meta query, short
-> meta set stdmdiff se , studylabel(studylbl) eslabel(Std. mean diff.)
  Effect-size label: Std. mean diff.
   Effect-size type: Generic
        Effect size: stdmdiff
          Std. err.: se
              Model: Random effects
             Method: REML
```

From the meta summary, our data were declared by using `meta set` with variables `stdmdiff` and `se` specifying the effect sizes and their standard errors, respectively. The declared meta-analysis model is the default random-effects model with the REML estimation method.

Examples are presented under the following headings:

Example 1: Small-study effects due to a confounding moderator
Example 2: Traditional tests and detailed output
Example 3: Harbord's test for small-study effects

Example 1: Small-study effects due to a confounding moderator

Our main focus is on investigating the potential presence of small-study effects by using a regression-based test. Because we are working with continuous data, we will use the Egger test.

```
. meta bias, egger
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
Regression-based Egger test for small-study effects
Random-effects model
Method: REML
H0: beta1 = 0; no small-study effects
            beta1 =      1.83
      SE of beta1 =     0.724
                z =      2.53
       Prob > |z| =    0.0115
```

From the output header, the regression-based test uses the declared random-effects model with REML estimation to account for residual heterogeneity. The estimated slope, $\widehat{\beta}_1$, is 1.83 with a standard error of 0.724, giving a test statistic of $z = 2.53$ and a p-value of 0.0115. This means that there is some evidence of small-study effects.

In example 9 of [META] **meta summarize**, we used subgroup-analysis on binary variable `week1`, which records whether teachers had prior contact with students for more than 1 week or for 1 week or less, to account for between-study heterogeneity. It explained most of the heterogeneity present among the effect sizes, with generally higher effect sizes in the low contact group.

Moderators that can explain a substantial amount of the heterogeneity should be included in the regression-based test as a covariate. By properly accounting for heterogeneity through the inclusion of `week1`, we can test for small-study effects due to reasons other than heterogeneity. We include factor variable `week1` as a moderator as follows:

```
. meta bias i.week1, egger
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
Regression-based Egger test for small-study effects
Random-effects model
Method: REML
Moderators: week1
H0: beta1 = 0; no small-study effects
            beta1 =      0.30
      SE of beta1 =     0.729
                z =      0.41
       Prob > |z| =    0.6839
```

Now that we have accounted for heterogeneity through moderator `week1`, the Egger test statistic is 0.41 with a p-value of 0.6839. Therefore, we have strong evidence to say that the presence of small-study effects was the result of heterogeneity induced by teacher-student prior contact time.

◁

▷ Example 2: Traditional tests and detailed output

For illustration, we perform the traditional version of the Egger regression-based test by specifying the `traditional` option. We also use the `detail` option to report the meta-regression results used to construct the Egger test.

```
. meta bias, egger traditional detail
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
Fixed-effects meta-regression                   Number of obs  =        19
Error: Multiplicative                           Dispersion phi =      1.69
Method: Inverse-variance                        Model F(1,17)  =      4.17
                                                Prob > F       =    0.0571
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      t    P>|t|     [95% conf. interval]
-------------+----------------------------------------------------------------
    _meta_se |   1.627717   .7975212     2.04   0.057    -.0549052     3.31034
       _cons |  -.1797108    .126835    -1.42   0.175    -.4473093     .0878876
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(17) = 28.77   Prob > Q_res = 0.0367
Regression-based Egger test for small-study effects
Fixed-effects model
Method: Inverse-variance
H0: beta1 = 0; no small-study effects
            beta1 =      1.63
      SE of beta1 =     0.798
                t =      2.04
       Prob > |t| =    0.0571
```

The traditional version also suggests the presence of small-study effects, but its p-value, 0.0571, is larger than that from example 1.

The results of the above command is identical to the following:

```
. meta regress _meta_se, fixed multiplicative tdistribution
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
Fixed-effects meta-regression                   Number of obs  =        19
Error: Multiplicative                           Dispersion phi =      1.69
Method: Inverse-variance                        Model F(1,17)  =      4.17
                                                Prob > F       =    0.0571
------------------------------------------------------------------------------
    _meta_es | Coefficient  Std. err.      t    P>|t|     [95% conf. interval]
-------------+----------------------------------------------------------------
    _meta_se |   1.627717   .7975212     2.04   0.057    -.0549052     3.31034
       _cons |  -.1797108    .126835    -1.42   0.175    -.4473093     .0878876
------------------------------------------------------------------------------
Test of residual homogeneity: Q_res = chi2(17) = 28.77   Prob > Q_res = 0.0367
```

The header and coefficient table from `meta bias`'s detailed output is identical to that produced by `meta regress` (see [META] **meta regress**).

◁

▷ Example 3: Harbord's test for small-study effects

In example 1 of [META] **meta funnelplot**, we explored the presence of publication bias in the NSAIDS data, which was described in *Effectiveness of nonsteroidal anti-inflammatory drugs (nsaids.dta)* of [META] **meta**. The contour-enhanced funnel plot from example 5 of [META] **meta funnelplot** revealed that the funnel-plot asymmetry was caused by the absence of small studies in the region where the tests of the log odds-ratios equal to zero were not statistically significant. This may suggest the presence of publication bias. We can explore this more formally by performing a test for small-study effects.

We use the declared version of the NSAIDS dataset.

```
. use https://www.stata-press.com/data/r18/nsaidsset, clear
(Effectiveness of nonsteroidal anti-inflammatory drugs; set with -meta esize-)
. meta query, short
-> meta esize nstreat nftreat nscontrol nfcontrol
  Effect-size label: Log odds-ratio
   Effect-size type: lnoratio
        Effect size: _meta_es
          Std. err.: _meta_se
              Model: Random effects
             Method: REML
```

The declared effect size is log odds-ratio, so we will use the Harbord regression-based test to investigate whether the small-study effects (or funnel-plot asymmetry) is present in these data.

```
. meta bias, harbord
  Effect-size label: Log odds-ratio
        Effect size: _meta_es
          Std. err.: _meta_se
Regression-based Harbord test for small-study effects
Random-effects model
Method: REML
H0: beta1 = 0; no small-study effects
            beta1 =      3.03
      SE of beta1 =     0.741
                z =      4.09
      Prob > |z| =    0.0000
```

The p-value is less than 0.0001, so we reject the null hypothesis of no small-study effects. It is difficult to be certain whether the small-study affects are driven by publication bias because of the presence of substantial heterogeneity in these data (see [META] **meta summarize**). Note that the regression-based test assumed an (REML) random-effects model, which accounts for heterogeneity present among the studies. If we had access to study-level covariates for these data that could explain some of the between-study variability, we could have specified them with `meta bias`.

◁

Stored results

For regression-based tests, `meta bias` stores the following in `r()`:

Scalars

`r(beta1)`	estimate of the main slope coefficient
`r(se)`	standard error for the slope estimate
`r(z)`	z statistic
`r(t)`	t statistic
`r(p)`	two-sided p-value

Macros

`r(testtype)`	type of test: `egger`, `harbord`, or `peters`
`r(model)`	meta-analysis model
`r(method)`	meta-analysis estimation method
`r(moderators)`	moderators used in regression-based tests

Matrices

`r(table)`	regression results

For Begg's test, `meta bias` stores the following in `r()`:

Scalars

`r(score)`	Kendall's score estimate
`r(se_score)`	standard error of Kendall's score
`r(z)`	z test statistic
`r(p)`	two-sided p-value

Macros

`r(testtype)`	`begg`

Methods and formulas

Methods and formulas are presented under the following headings:

Regression-based tests
Egger's linear regression test
Harbord's test for log odds-ratios or log risk-ratios
Peters's test for log odds-ratios
Begg's rank correlation test

Let K be the number of studies for a given meta-analysis. For the jth study, $\hat{\theta}_j$ denotes the estimated effect size, and $\hat{\sigma}^2_j$ denotes the effect-size (within-study) variance. The tests are applicable to any type of effect size as long as it is asymptotically normally distributed.

For two-sample binary data, also consider the following 2×2 table for the jth study.

group	event	no event	size
treatment	a_j	b_j	$n_{1j} = a_j + b_j$
control	c_j	d_j	$n_{2j} = c_j + d_j$

The total sample size for the jth study is denoted by $n_j = n_{1j} + n_{2j}$.

Regression-based tests

Regression-based tests use meta-regression to examine a linear relationship between the individual effect sizes and measures of study precision such as the effect-size standard errors, possibly adjusting for moderators that explain some of the between-study variability.

In the subsections below, we provide the traditional versions of the regression-based tests. The extensions of traditional versions include the support of other models such as a random-effects model and the support of moderators.

In the presence of moderators, the test for small-study effects is the test of H_0: $\beta_1 = 0$ in the corresponding meta-regression with the following linear predictor,

$$\mathbf{x}_j\boldsymbol{\beta} = \beta_0 + \beta_1 m_j + \beta_2 x_{2,j} + \cdots + \beta_{p-1} x_{p-1,j}$$

where $x_{2,j}, \ldots, x_{p-1,j}$ represent the *moderators* specified with `meta bias` and $m_j = \hat{\sigma}_j$ for the Egger test, $m_j = 1/\sqrt{V_j}$ for the Harbord test, and $m_j = 1/n_j$ for the Peters test. See the subsections below for details about these tests. Also see Sterne and Egger (2005).

The computations of regression-based tests are based on the corresponding meta-regression models; see *Methods and formulas* of [META] **meta regress**.

The formulas below are based on Harbord, Harris, and Sterne (2016), Sterne and Egger (2005), and Peters et al. (2010).

Egger's linear regression test

The formulas and discussion in this subsection are based on Sterne and Egger (2005).

The test proposed by Egger, Davey Smith, Schneider, and Minder (1997) is based on a simple linear regression of the standard normal variate, which is defined as the individual effect-size estimate divided by its standard error, against the study precision, which is defined as the reciprocal of the standard error:

$$E\left(\frac{\hat{\theta}_j}{\hat{\sigma}_j}\right) = b_0 + b_1\frac{1}{\hat{\sigma}_j} \tag{1}$$

The Egger test of no small-study effects is the test of H_0: $b_0 = 0$.

Linear regression model (1) is equivalent to the weighted linear regression of the effect sizes $\hat{\theta}_j$'s on their standard errors $\hat{\sigma}_j$'s,

$$E\left(\hat{\theta}_j\right) = b_1 + b_0\hat{\sigma}_j \tag{2}$$

with weights inversely proportional to the variances of the effect sizes, $w_j = 1/\hat{\sigma}_j^2$. Note that the intercept b_0 in regression (1) corresponds to the slope in the weighted regression (2). Therefore, Egger test for small-study effects corresponds to a test of a linear trend in a funnel plot (see [META] **meta funnelplot**) of effect sizes against their standard errors.

Let's denote $\beta_0 = b_1$ and $\beta_1 = b_0$. The statistical model for the traditional Egger's test, as it originally appeared in the literature (Egger et al. 1997), is given by

$$\hat{\theta}_j = \beta_0 + \beta_1\hat{\sigma}_j + \epsilon_j \quad \text{weighted by } w_j = 1/\hat{\sigma}_j^2, \text{ where } \epsilon_j \sim N\left(0,\, \hat{\sigma}_j^2\phi\right)$$

and ϕ is the overdispersion parameter as defined in multiplicative meta-regression; see *Introduction* of [META] **meta regress**.

Egger's test for small-study effects is the test of H_0: $\beta_1 = 0$, and the null hypothesis is rejected if

$$t_{\text{egger}} = \left|\frac{\hat{\beta}_1}{\widehat{\text{SE}}\left(\hat{\beta}_1\right)}\right| > t_{K-2,1-\alpha/2}$$

where $t_{K-2,1-\alpha/2}$ is the $(1-\alpha/2)$th quantile of the Student's t distribution with $K-2$ degrees of freedom. The above test is performed when you specify options `egger` and `traditional`.

❑ Technical note

Sterne and Egger (2005) point out that, originally, Egger et al. (1997) used a weighted version of (1) with weights equal to the inverse of the variances of effect sizes ($1/\hat{\sigma}_j^2$'s). The authors strongly recommend that this version of the test not be used because it does not have a theoretical justification.

❑

Harbord's test for log odds-ratios or log risk-ratios

Consider the fixed-effects model $\hat{\theta}_j \sim N(\theta, \hat{\sigma}_j^2)$. For a study j, let Z_j be the first derivative (score) and V_j be the negative second derivative (Fisher's information) of the model log likelihood with respect to θ evaluated at $\theta = 0$ (Whitehead and Whitehead 1991; Whitehead 1997).

For two-sample binary data, Harbord, Egger, and Sterne (2006) proposed a modification of the Egger test based on the intercept in an unweighted regression of $Z_j/\sqrt{V_j}$ against $\sqrt{V_j}$:

$$E\left(\frac{Z_j}{\sqrt{V_j}}\right) = b_0 + b_1\sqrt{V_j} \tag{3}$$

When the effect of interest is the log odds-ratio,

$$Z_j = \frac{a_j - (a_j + c_j)\, n_{1j}}{n_j} \text{ and } V_j = \frac{n_{1j} n_{2j} (a_j + c_j)(b_j + d_j)}{n_j^2 (n_j - 1)}$$

Note that Z_j and V_j are the numerator and denominator of the log Peto's odds-ratio as defined in *Methods and formulas* of [META] **meta esize**.

When the effect of interest is the log risk-ratio,

$$Z_j = \frac{a_j n_j - (a_j + c_j)\, n_{1j}}{b_j + d_j} \text{ and } V_j = \frac{n_{1j} n_{2j} (a_j + c_j)}{n_j (b_j + d_j)}$$

Whitehead (1997) showed that when θ_j is small and n_j is large, $\hat{\theta}_j \approx Z_j/V_j$ and $\hat{\sigma}_j^2 \approx 1/V_j$. In this case, the Harbord regression model (3) is equivalent to Egger's regression model (1). Thus, Harbord's test becomes equivalent to Egger's test when all studies are large and have small effect sizes (Harbord, Harris, and Sterne 2016).

As with Egger's test, if we use the weighted version of regression model (3) and denote $\beta_0 = b_1$ and $\beta_1 = b_0$ in that model, the statistical model for the Harbord test, as it originally appeared in the literature, is given by

$$\frac{Z_j}{V_j} = \beta_0 + \beta_1 \frac{1}{\sqrt{V_j}} + \epsilon_j \quad \text{weighted by } w_j = V_j, \text{ where } \epsilon_j \sim N\left(0, \frac{\phi}{V_j}\right)$$

where ϕ is the overdispersion parameter as defined in multiplicative meta-regression; see *Introduction* of [META] **meta regress**.

Then, the traditional Harbord test is the test of $H_0\colon \beta_1 = 0$, and its null hypothesis is rejected if $t_{\text{harbord}} = \left|\hat{\beta}_1/\text{SE}(\hat{\beta}_1)\right| > t_{K-2,1-\alpha/2}$. This test can be performed when you specify options `harbord` and `traditional`.

Peters's test for log odds-ratios

Peters et al. (2006) provide a test based on the following model:

$$\hat{\theta}_j = \beta_0 + \beta_1 \frac{1}{n_j} + \epsilon_j \quad \text{weighted by } w_j = (a_j + c_j)(b_j + d_j)/n_j, \text{ where } \epsilon_j \sim N(0, \hat{\sigma}_j^2\phi)$$

$\hat{\theta}_j = \ln(\widehat{\text{OR}}_j)$, and ϕ is the overdispersion parameter as defined in multiplicative meta-regression; see *Introduction* of [META] **meta regress**.

The traditional Peters test is the test of $H_0 : \beta_1 = 0$, and its null hypothesis is rejected if $t_{\text{peters}} = \left|\widehat{\beta}_1/\text{SE}(\widehat{\beta}_1)\right| > t_{K-2,1-\alpha/2}$. This test can be performed when you specify options `peters` and `traditional`.

When the test is based on the random-effects model, the weights are given by $w_j = 1/(\hat{\sigma}_j^2 + \hat{\tau}^2)$.

Begg's rank correlation test

Consider the standardized effect sizes

$$\hat{\theta}_j^{\text{s}} = \frac{\hat{\theta}_j - \hat{\theta}_{\text{IV}}}{\sqrt{v_j^{\text{s}}}}$$

where

$$\hat{\theta}_{\text{IV}} = \frac{\sum_{j=1}^{K} \hat{\theta}_j/\hat{\sigma}_j^2}{\sum_{j=1}^{K} 1/\hat{\sigma}_j^2}$$

and

$$v_j^{\text{s}} = \text{Var}\left(\hat{\theta}_j - \hat{\theta}_{\text{IV}}\right) = \hat{\sigma}_j^2 - \left(\sum_{j=1}^{K} \hat{\sigma}_j^{-2}\right)^{-1}$$

The Begg test (Begg and Mazumdar 1994) is Kendall's rank correlation test of independence between $\hat{\theta}_j^{\text{s}}$'s and $\hat{\sigma}_j^2$'s; see *Methods and formulas* of [R] **spearman**.

References

Begg, C. B., and M. Mazumdar. 1994. Operating characteristics of a rank correlation test for publication bias. *Biometrics* 50: 1088–1101. https://doi.org/10.2307/2533446.

Egger, M., G. Davey Smith, and A. N. Phillips. 1997. Meta-analysis: Principles and procedures. *BMJ* 315: 1533–1537. https://doi.org/10.1136/bmj.315.7121.1533.

Egger, M., G. Davey Smith, M. Schneider, and C. Minder. 1997. Bias in meta-analysis detected by a simple, graphical test. *BMJ* 315: 629–634. https://doi.org/10.1136/bmj.315.7109.629.

Harbord, R. M., M. Egger, and J. A. C. Sterne. 2006. A modified test for small-study effects in meta-analyses of controlled trials with binary endpoints. *Statistics in Medicine* 25: 3443–3457. https://doi.org/10.1002/sim.2380.

Harbord, R. M., R. J. Harris, and J. A. C. Sterne. 2016. "Updated tests for small-study effects in meta-analyses". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 153–165. 2nd ed. College Station, TX: Stata Press.

Hartung, J., and G. Knapp. 2001a. On tests of the overall treatment effect in meta-analysis with normally distributed responses. *Statistics in Medicine* 20: 1771–1782. https://doi.org/10.1002/sim.791.

———. 2001b. A refined method for the meta-analysis of controlled clinical trials with binary outcome. *Statistics in Medicine* 20: 3875–3889. https://doi.org/10.1002/sim.1009.

Ioannidis, J. P. A., and T. A. Trikalinos. 2007. The appropriateness of asymmetry tests for publication bias in meta-analyses: A large survey. *Canadian Medical Association Journal* 176: 1091–1096. https://doi.org/10.1503/cmaj.060410.

Knapp, G., and J. Hartung. 2003. Improved tests for a random effects meta-regression with a single covariate. *Statistics in Medicine* 22: 2693–2710. https://doi.org/10.1002/sim.1482.

Peters, J. L., A. J. Sutton, D. R. Jones, K. R. Abrams, and L. Rushton. 2006. Comparison of two methods to detect publication bias in meta-analysis. *Journal of the American Medical Association* 295: 676–680. https://doi.org/10.1001/jama.295.6.676.

Peters, J. L., A. J. Sutton, D. R. Jones, K. R. Abrams, L. Rushton, and S. G. Moreno. 2010. Assessing publication bias in meta-analyses in the presence of between-study heterogeneity. *Journal of the Royal Statistical Society*, A ser., 173: 575–591. https://doi.org/10.1111/j.1467-985X.2009.00629.x.

Raudenbush, S. W. 1984. Magnitude of teacher expectancy effects on pupil IQ as a function of the credibility of expectancy induction: A synthesis of findings from 18 experiments. *Journal of Educational Psychology* 76: 85–97. https://doi.org/10.1037/0022-0663.76.1.85.

Raudenbush, S. W., and A. S. Bryk. 1985. Empirical Bayes meta-analysis. *Journal of Educational Statistics* 10: 75–98. https://doi.org/10.2307/1164836.

Sidik, K., and J. N. Jonkman. 2002. A simple confidence interval for meta-analysis. *Statistics in Medicine* 21: 3153–3159. https://doi.org/10.1002/sim.1262.

Steichen, T. J. 2016. "Tests for publication bias in meta-analysis". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 166–176. 2nd ed. College Station, TX: Stata Press.

Sterne, J. A. C., and M. Egger. 2005. "Regression methods to detect publication and other bias in meta-analysis". In *Publication Bias in Meta-Analysis: Prevention, Assessment and Adjustments*, edited by H. R. Rothstein, A. J. Sutton, and M. Borenstein, 99–110. Chichester, UK: Wiley. https://doi.org/10.1002/0470870168.ch6.

Sterne, J. A. C., D. Gavaghan, and M. Egger. 2000. Publication and related bias in meta-analysis: Power of statistical tests and prevalence in the literature. *Journal of Clinical Epidemiology* 53: 1119–1129. https://doi.org/10.1016/S0895-4356(00)00242-0.

Sterne, J. A. C., A. J. Sutton, J. P. A. Ioannidis, N. Terrin, D. R. Jones, J. Lau, J. R. Carpenter, G. Rücker, R. M. Harbord, C. H. Schmid, J. Tetzlaff, J. J. Deeks, J. L. Peters, P. Macaskill, G. Schwarzer, S. Duval, D. G. Altman, D. Moher, and J. P. T. Higgins. 2011. Recommendations for examining and interpreting funnel plot asymmetry in meta-analyses of randomised controlled trials. *BMJ* 343: d4002. https://doi.org/10.1136/bmj.d4002.

Sutton, A. J. 2009. "Publication bias". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 435–452. 2nd ed. New York: Russell Sage Foundation.

Whitehead, A., and J. Whitehead. 1991. A general parametric approach to the meta-analysis of randomized clinical trials. *Statistics in Medicine* 10: 1665–1677. https://doi.org/10.1002/sim.4780101105.

Whitehead, J. 1997. *The Design and Analysis of Sequential Clinical Trials*. Rev. 2nd ed. Chichester, UK: Wiley.

Also see

[META] **meta data** — Declare meta-analysis data

[META] **meta funnelplot** — Funnel plots

[META] **meta regress** — Meta-analysis regression

[META] **meta summarize** — Summarize meta-analysis data+

[META] **meta trimfill** — Nonparametric trim-and-fill analysis of publication bias

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta trimfill — Nonparametric trim-and-fill analysis of publication bias

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
References | Also see

Description

`meta trimfill` performs the nonparametric "trim-and-fill" method to account for publication bias in meta-analysis. The command estimates the number of studies potentially missing from a meta-analysis because of publication bias, imputes these studies, and computes the overall effect-size estimate using the observed and imputed studies. It can also provide a funnel plot, in which omitted studies are imputed.

Quick start

Perform the trim-and-fill analysis of publication bias under the default setting specified in either `meta set` or `meta esize`

```
meta trimfill
```

Same as above, and request a funnel plot

```
meta trimfill, funnel
```

Specify that the number of missing studies be estimated using the rightmost-run estimator instead of the default linear estimator

```
meta trimfill, estimator(run)
```

Specify that the estimation of the overall effect size be based on the fixed-effects inverse-variance method during the iteration step and random-effects DerSimonian–Laird method during the pooling step of the trim-and-fill algorithm

```
meta trimfill, itermethod(fixed) poolmethod(dlaird)
```

Specify that studies in the right side of the funnel plot (with large effect sizes) be suppressed and need to be imputed

```
meta trimfill, right
```

Menu

Statistics > Meta-analysis

Syntax

```
meta trimfill [if] [in] [, options]
```

options	Description
Main	
`estimator(`*estimator*`)`	estimator for the number of missing studies; default is `linear`
`left`	impute studies on the left side of the funnel plot
`right`	impute studies on the right side of the funnel plot
`funnel[(`*funnelopts*`)]`	draw funnel plot
Options	
`level(#)`	set confidence level; default is as declared for meta-analysis
eform_option	report exponentiated results
`[no]metashow`	display or suppress meta settings in the output
display_options	control column formats
Iteration	
`random[(`*remethod*`)]`	random-effects meta-analysis to use for iteration and pooling steps
`common`	common-effect meta-analysis to use for iteration and pooling steps; implies inverse-variance method
`fixed`	fixed-effects meta-analysis to use for iteration and pooling steps; implies inverse-variance method
`itermethod(`*method*`)`	meta-analysis to use for iteration step
`poolmethod(`*method*`)`	meta-analysis to use for pooling step
`iterate(#)`	maximum number of iterations for the trim-and-fill algorithm; default is `iterate(100)`
`[no]log`	display an iteration log from the trim-and-fill algorithm

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

estimator	Description
`linear`	linear estimator, L_0; the default
`run`	run estimator, R_0
`quadratic`	quadratic estimator, Q_0 (rarely used)

remethod	Description
`reml`	restricted maximum likelihood; the default
`mle`	maximum likelihood
`ebayes`	empirical Bayes
`dlaird`	DerSimonian–Laird
`sjonkman`	Sidik–Jonkman
`hedges`	Hedges
`hschmidt`	Hunter–Schmidt

Options

Main

`estimator(`*estimator*`)` specifies the type of estimator for the number of missing studies. *estimator* is one of `linear`, `run`, or `quadratic`. The default is `estimator(linear)`.

`linear` specifies that the "linear" estimator, L_0, be used to estimate the number of missing studies. This is the default estimator.

`run` specifies that the rightmost-run estimator, R_0, be used to estimate the number of missing studies.

`quadratic` specifies that the "quadratic" estimator, Q_0, be used to estimate the number of missing studies. This estimator is not recommended in the literature and provided for completeness.

Duval and Tweedie (2000a) found that the L_0 and R_0 estimators perform better in terms of mean squared error (MSE) than the Q_0 estimator, with L_0 having the smallest MSE in certain cases. They also found that R_0 tends to be conservative in some cases. Therefore, L_0 is chosen to be the default, but the authors recommend that all estimators be considered in practice. Also see *Estimating the number of missing studies* in *Methods and Formulas* for details about the estimators.

`left` and `right` specify the side of the funnel plot, where the missing studies are to be imputed. By default, the side is chosen based on the results of the traditional Egger test—if the estimated slope is positive, `left` is assumed; otherwise, `right` is assumed. Only one of `left` or `right` is allowed.

`left` assumes that the leftmost (smallest) effect sizes have been suppressed and specifies to impute them.

`right` assumes that the rightmost (largest) effect sizes have been suppressed and specifies to impute them.

`funnel` and `funnel(`*funnelopts*`)` specify to draw a funnel plot that includes the imputed studies.

funnelopts are any options as documented in [META] **meta funnelplot**, except `random[()]`, `common[()]`, `fixed[()]`, `by()`, and `[no]metashow`.

Options

`level(#)` specifies the confidence level, as a percentage, for confidence intervals. The default is as declared for the meta-analysis session; see *Declaring a confidence level for meta-analysis* in [META] **meta data**. Also see option `level()` in [META] **meta set**.

eform_option is one of `eform`, `eform(`*string*`)`, `or`, or `rr`. It reports exponentiated effect sizes and transforms their respective confidence intervals, whenever applicable. By default, the results are displayed in the metric declared with `meta set` or `meta esize` such as log odds-ratios and log risk-ratios. *eform_option* affects how results are displayed, not how they are estimated and stored. *eform_option* is not available with two-sample continuous data, one-sample binary data, and correlation data.

`eform(`*string*`)` labels the exponentiated effect sizes as *string*; the other options use default labels. The default label is specific to the chosen effect size. For example, option `eform` uses `Odds ratio` when used with log odds-ratios declared with `meta esize` or `Risk ratio` when used with the declared log risk-ratios. Option `or` is a synonym for `eform` when log odds-ratio is declared, and option `rr` is a synonym for `eform` when log risk-ratio is declared. If option `eslabel(`*eslab*`)` is specified during declaration, then `eform` will use the `exp(`*eslab*`)` label or, if *eslab* is too long, the `exp(ES)` label.

metashow and nometashow display or suppress the meta setting information. By default, this information is displayed at the top of the output. You can also specify nometashow with meta update to suppress the meta setting output for the entire meta-analysis session.

display_options: cformat(%*fmt*); see [R] **Estimation options**.

Iteration

Options random(), common, and fixed, when specified with meta trimfill, temporarily override the global model declared by meta set or meta esize during the computation. These options specify that the same method be used during both iteration and pooling steps. To specify different methods, use options itermethod() and poolmethod(). Options random(), common, and fixed may not be combined. If these options are omitted, the declared meta-analysis model is assumed; see *Declaring a meta-analysis model* in [META] **meta data**. Also see *Meta-analysis models* in [META] **Intro**.

random and random(*remethod*) specify that a random-effects model be assumed for meta-analysis during iteration and pooling steps of the trim-and-fill algorithm; see *Random-effects model* in [META] **Intro**.

remethod specifies the type of estimator for the between-study variance τ^2. *remethod* is one of reml, mle, ebayes, dlaird, sjonkman, hedges, or hschmidt. random is a synonym for random(reml). See *Options* in [META] **meta esize** for more information.

common specifies that a common-effect model be assumed for meta-analysis during iteration and pooling steps of the trim-and-fill algorithm; see *Common-effect ("fixed-effect") model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about common-effect versus fixed-effects models.

fixed specifies that a fixed-effects model be assumed for meta-analysis during iteration and pooling steps of the trim-and-fill algorithm; see *Fixed-effects model* in [META] **Intro**. It uses the inverse-variance estimation method; see *Meta-analysis estimation methods* in [META] **Intro**. Also see the discussion in [META] **meta data** about fixed-effects versus common-effect models.

itermethod(*method*) specifies the meta-analysis method to use during the iteration step of the trim-and-fill algorithm. The default is the method declared for meta-analysis; see *Declaring a meta-analysis model* in [META] **meta data**. Also see *Trim-and-fill algorithm* in *Methods and formulas*. This option may not be combined with random(), common, or fixed.

method is one of the random-effects meta-analysis methods, *remethod*; or a common-effect inverse-variance method, common; or a fixed-effects inverse-variance method, fixed; see *Options* in [META] **meta set** for details.

poolmethod(*method*) specifies the meta-analysis method to use during the pooling step of the trim-and-fill algorithm. The default is to use the method declared for meta-analysis; see *Declaring a meta-analysis model* in [META] **meta data**. Also see *Trim-and-fill algorithm* in *Methods and formulas*. This option may not be combined with random(), common, or fixed.

method is one of the random-effects meta-analysis methods, *remethod*; or a common-effect inverse-variance method, common; or a fixed-effects inverse-variance method, fixed; see *Options* in [META] **meta set** for details.

iterate(#) specifies the maximum number of iterations for the trim-and-fill algorithm. The default is iterate(100). When the number of iterations equals iterate(), the algorithm stops and presents the current results. If convergence is not reached, a warning message is also displayed. If convergence is declared before this threshold is reached, the algorithm will stop when convergence is declared.

`nolog` and `log` specify whether an iteration log showing the progress of the trim-and-fill algorithm is to be displayed. By default, the log is suppressed but you can specify `log` to display it.

Remarks and examples

Remarks are presented under the following headings:

Introduction
Using meta trimfill
Examples of using meta trimfill

Introduction

Publication bias is a serious problem in meta-analysis. It arises when the decision whether to publish a study depends on the statistical significance of the results of the study. Typically, more significant findings are more likely to get published. See *Publication bias* of [META] **Intro** for details.

Publication bias can be assessed visually with a funnel plot ([META] **meta funnelplot**). More formal tests for the presence of publication bias or, more generally, of small-study effects are also available ([META] **meta bias**). The focus of this entry is on assessing the impact of the publication bias on the results. One of the methods popular in practice is the so-called trim-and-fill method, introduced by Duval and Tweedie (2000a, 2000b).

The main goal of the trim-and-fill method is to evaluate the impact of publication bias on our final inference. The idea of the method is to iteratively estimate the number of studies potentially missing because of publication bias at the iteration stage. Then, at the final pooling stage, impute (fill in) the effect sizes and effect-size standard errors for these studies and use the completed set of studies to compute the overall effect-size estimate. For details, see *Trim-and-fill algorithm*.

Meta-analysis literature does not provide definitive recommendations for which model should be used during the iteration stage of the method. Duval (2005) points out that a random-effects model was recommended initially (National Research Council 1992; Sutton et al. 1998), but a common-effect model was found later to provide a more conservative approach. What is meant by conservative? Random-effects models tend to give more weight to less precise studies than common-effect models (Poole and Greenland 1999). But less precise (smaller) studies are more likely to exhibit publication bias. In general, neither model outperforms the other in all situations. Thus, meta-analysts are advised to try both in practice and compare the results.

Just like other methods for detecting publication bias such as funnel plots and tests for the funnel-plot asymmetry, the trim-and-fill method is sensitive to the presence of substantial between-study heterogeneity (for example, Peters et al. [2007]). The method is agnostic to the reasons for the funnel-plot asymmetry, be it publication bias or between-study heterogeneity. It merely detects the asymmetry and attempts to correct it. Unlike the tests for the funnel-plot asymmetry ([META] **meta bias**), the trim-and-fill method does not allow accounting for the moderators that could potentially explain the heterogeneity. For categorical moderators, however, you can perform the method separately for each category. In any case, you should take potential heterogeneity into account when interpreting the final estimate of the effect size from the trim-and-fill method. See the *Trim and fill* section in Deeks, Higgins, and Altman (2017) for details.

Duval (2005) states that there are researchers who are not comfortable with using imputed (fictional) studies to adjust the meta-analysis results for publication bias (Begg 1997). Indeed, one of the shortcomings of this method is that it treats "imputed" effect sizes as if they were observed in the final meta-analysis and thus potentially underestimates their corresponding standard errors.

Note that the emphasis of the trim-and-fill method is on sensitivity analyses and not on trying to recover the missing study values. The actual imputed values are not of interest. The primary goal of the method is to explore the impact of missing studies on the overall effect-size estimate. This method should be used purely as a guide to which meta-analysis appears to be more susceptible to publication bias.

Using meta trimfill

`meta trimfill` implements the trim-and-fill method of Duval and Tweedie (2000a, 2000b). It supports three estimators: `linear`, `run`, and `quadratic`, which can be specified in the `estimator()` option.

It can impute studies on the left side of the funnel plot with option `left` or on the right with option `right`. By default, it assumes the side where the missingness occurs based on the traditional Egger test; see [META] **meta bias**. Specifically, if the slope from the Egger test is positive, then `left` is assumed; otherwise, `right` is assumed.

You can use option `funnel` or `funnel()` to draw a funnel plot, which would include the imputed studies. You can customize the default look of the funnel plot by specifying `funnel()`.

You can use *eform_option* to exponentiate the results reported in the log metric such as log odds-ratios and log risk-ratios.

By default, the model declared with either `meta set` or `meta esize` is assumed during both the iteration and pooling stages. You can change this by specifying one of `random()`, `common`, or `fixed`. You can also specify a different model for the iteration stage in the `itermethod()` option and for the pooling stage in the `poolmethod()` option.

Examples of using meta trimfill

Here we use `metatrim.dta` to demonstrate the usages of the trim-and-fill method. (This dataset is based on `pupiliq.dta`; it contains a subset of variables renamed to have generic names and fewer observations.) Also see example 14 of [META] **meta** for the trim-and-fill analysis of the NSAIDS dataset.

▷ Example 1: A basic trim-and-fill analysis

Consider the dataset `metatrim.dta`. In this dataset, each study consists of two groups, and the standardized difference in means between the two groups is recorded, as well as the standard error for this difference. We begin by declaring the effect sizes and their standard errors with `meta set`.

```
. use https://www.stata-press.com/data/r18/metatrim
(Subset of pupiliq.dta)
. meta set stdmdiff se

Meta-analysis setting information

 Study information
    No. of studies: 16
       Study label: Generic
        Study size: N/A
       Effect size
             Type: <generic>
            Label: Effect size
         Variable: stdmdiff
       Precision
         Std. err.: se
               CI: [_meta_cil, _meta_ciu]
        CI level: 95%
  Model and method
            Model: Random effects
           Method: REML
```

We perform a trim-and-fill analysis to see whether any studies are estimated to be missing:

```
. meta trimfill
  Effect-size label: Effect size
        Effect size: stdmdiff
          Std. err.: se

Nonparametric trim-and-fill analysis of publication bias
Linear estimator, imputing on the left

Iteration                              Number of studies =     19
  Model: Random-effects                         observed =     16
 Method: REML                                    imputed =      3

Pooling
  Model: Random-effects
 Method: REML
```

Studies	Effect size	[95% conf.	interval]
Observed	0.119	-0.018	0.256
Observed + Imputed	0.034	-0.150	0.219

The model in the iteration and pooling steps is a random-effects model with the REML estimation as declared by `meta set`. These models may be individually controlled by using options `itermethod()` and `poolmethod()`. By default, estimation of the number of missing studies K_0 was based on the `linear` estimator. This can be changed using the `estimator()` option.

The mean effect size based on the 16 observed studies is 0.119 with a 95% CI of $[-0.018, 0.256]$. Three hypothetical studies, $\widehat{K}_0 = 19 - 16 = 3$, are estimated to be missing and are imputed. If these three studies were included in the meta-analysis, the funnel plot would be more symmetrical. After imputing the studies, we obtain an updated estimate (based on the 19 studies, observed plus imputed) of the mean effect size of 0.034 with a 95% CI $[-0.150, 0.219]$.

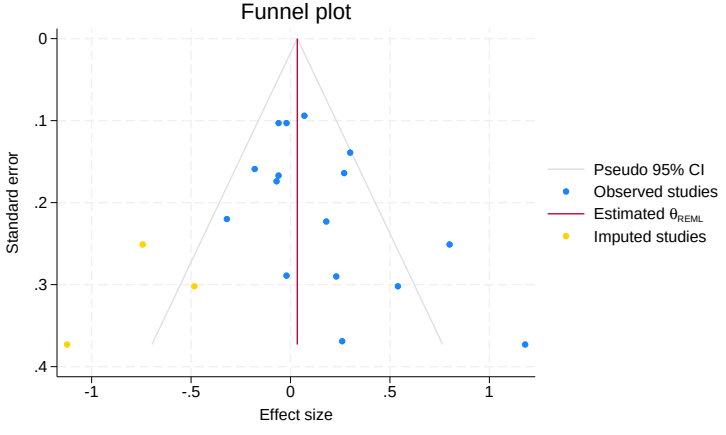
◁

▷ Example 2: Funnel plot

Continuing with example 1, we can request a funnel plot by specifying the `funnel` option.

```
. meta trimfill, funnel
  Effect-size label: Effect size
        Effect size: stdmdiff
          Std. err.: se
Nonparametric trim-and-fill analysis of publication bias
Linear estimator, imputing on the left
Iteration                             Number of studies =     19
  Model: Random-effects                       observed =     16
 Method: REML                                 imputed =      3
Pooling
  Model: Random-effects
 Method: REML
```

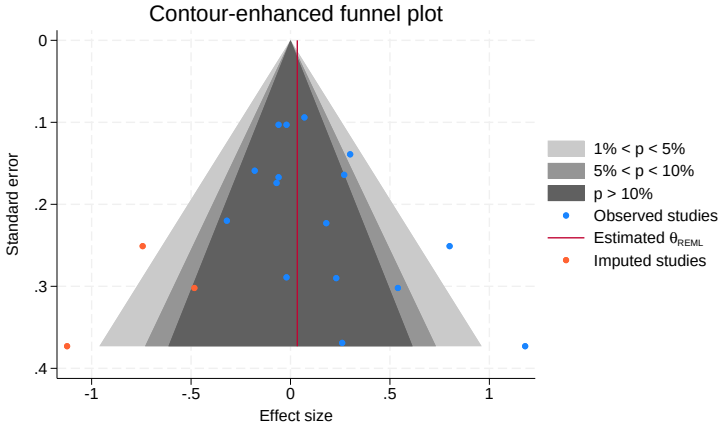
Studies	Effect size	[95% conf.	interval]
Observed	0.119	-0.018	0.256
Observed + Imputed	0.034	-0.150	0.219



The imputed studies are shown in yellow in the above funnel plot. Only one of the three imputed studies lies within the 95% pseudo CI; see [META] **meta funnelplot** for the interpretation of the funnel plot.

We may be interested in a contour-enhanced funnel plot of the completed set of studies (observed and imputed) to visually assess whether the imputed studies fall in regions of statistical significance. This is done by specifying the `contour(1 5 10)` suboption within `funnel()` in `meta trimfill`. We suppress the output from the command with `quietly`.

```
. quietly meta trimfill, funnel(contour(1 5 10))
```



The plot reveals that two of the three imputed studies fall in the white region corresponding to a p-value less than 1%; see example 5 of [META] **meta funnelplot** for more examples of contour-enhanced funnel plots.

◁

▷ Example 3: Specifying different pooling and iteration methods

By default, the same meta-analysis methods will be used for both the pooling and iteration steps, but we can specify other methods with the `poolmethod()` and `itermethod()` options. For example, below we specify the random-effects DerSimonian–Laird method for the pooling step and the fixed-effects inverse-variance method for the iteration step:

```
. meta trimfill, itermethod(fixed) poolmethod(dlaird)
  Effect-size label: Effect size
        Effect size: stdmdiff
          Std. err.: se
Nonparametric trim-and-fill analysis of publication bias
Linear estimator, imputing on the left
Iteration                             Number of studies =     19
  Model: Fixed-effects                        observed =     16
 Method: Inverse-variance                     imputed =      3
Pooling
  Model: Random-effects
 Method: DerSimonian-Laird
```

Studies	Effect size	[95% conf.	interval]
Observed	0.117	-0.016	0.249
Observed + Imputed	0.033	-0.120	0.186

The estimates are only slightly smaller than what they were in example 1, where we used the random-effects REML estimation method in both steps.

◁

▷ Example 4: Specifying the estimator

By default, the linear estimator is used to estimate the number of missing studies. Let's use the rightmost-run estimator and see how the results differ:

```
. meta trimfill, estimator(run)
  Effect-size label: Effect size
        Effect size: stdmdiff
          Std. err.: se
Nonparametric trim-and-fill analysis of publication bias
Run estimator, imputing on the left
Iteration                             Number of studies =     18
  Model: Random-effects                        observed =     16
 Method: REML                                  imputed =      2
Pooling
  Model: Random-effects
 Method: REML
```

Studies	Effect size	[95% conf.	interval]
Observed	0.119	-0.018	0.256
Observed + Imputed	0.059	-0.124	0.242

The mean effect size based on the 16 observed studies is still 0.119, as it was in example 1, but the updated estimate that includes the imputed studies is large now, 0.059. Also, the estimated number of missing studies is 2 in this example instead of 3.

◁

Stored results

`meta trimfill` stores the following in `r()`:

Scalars

`r(K_total)`	total number of studies (observed plus imputed)
`r(K_observed)`	number of observed studies
`r(K_imputed)`	number of imputed studies
`r(converged)`	`1` if trim-and-fill algorithm converged, `0` otherwise

Macros

`r(estimator)`	type of estimator for the number of missing studies
`r(side)`	side of the funnel plot with missing studies; `left` or `right`
`r(itermethod)`	meta-analysis estimation method used during iteration step
`r(poolmethod)`	meta-analysis estimation method used during final pooling step
`r(level)`	confidence level for CIs

Matrices

`r(table)`	trim-and-fill table of results
`r(imputed)`	matrix of effect sizes and their standard errors for imputed studies

Methods and formulas

For each study j, $j = 1, \ldots, K$, let $\hat{\theta}_j$ be the effect size and $\hat{\sigma}^2_j$ be its squared standard error (or within-study variance). The goal is to estimate an overall effect size, θ, from the sample of the effect sizes, accounting for (potentially) suppressed studies with extreme values from the meta-analysis.

The formulas and discussion below are based on Duval and Tweedie (2000b), Duval (2005), and Steichen (2000).

Suppose that there are K observed studies and K_0 relevant studies that are potentially missing from the meta-analysis because of publication bias. The goal is to estimate the value of K_0 as well as the effect size from the "completed" set of $K + K_0$ studies. The formulas below are based on the assumption that the K_0 effect-size values that are missing are the most extreme smallest values. That is, the studies with "nonsignificant" results are the ones being suppressed. This is also equivalent to assuming that the studies are missing from the left side of the funnel plot (option `left`). If missing studies are expected to be missing on the right side of the funnel (option `right`), Duval (2005) indicates that the same formulas below can be applied after multiplying the effect sizes by -1.

Note that the default behavior of `meta trimfill` is to assume the side where the missingness occurs based on the traditional Egger test; see *Egger's linear regression test* in [META] **meta bias**. Specifically, if the slope from the Egger test is positive, then option `left` is assumed, and vice versa.

Methods and formulas are presented under the following headings:

Estimating the number of missing studies
Trim-and-fill algorithm

Estimating the number of missing studies

We follow the description and notation from Duval and Tweedie (2000b) and Steichen (2000). Let $X_j = \hat{\theta}_j - \hat{\theta}$ and denote the ranks of $|X_j|$'s as $r_j^* \in \{1, 2, \ldots, K\}$. Let $\gamma^* \geq 0$ be the length of the rightmost run of ranks associated with positive values of X_j's such that $\gamma^* = K - r_h^*$, where r_h^* is the absolute rank of the most negative X_j with the corresponding index h; see Rothstein, Sutton, and Borenstein (2005, 137) for an example illustrating the computation of γ^* manually.

The "trimmed" rank test statistic for the observed K values is

$$T_K = \sum_{X_j > 0} r_j^*$$

The following estimators of K_0 can be considered based on the above quantities:

$$R_0 = \gamma^* - 1$$
$$L_0 = \frac{4T_K - K(K+1)}{2K - 1}$$
$$Q_0 = K - \frac{1}{2} - \sqrt{2K^2 - 4T_K + \frac{1}{4}}$$

Because K_0 must be an integer, the above estimators are rounded as follows,

$$R_0^+ = \max(0, R_0)$$
$$L_0^+ = \max\{0, \texttt{round}(L_0)\}$$
$$Q_0^+ = \max\{0, \texttt{round}(Q_0)\}$$

where `round(`x`)` is x rounded to the nearest integer.

Duval and Tweedie (2000a) compared the behavior of the MSEs of the estimators R_0^+, L_0^+, and Q_0^+ using simulated data and found that L_0^+ and R_0^+ seem to perform better than Q_0^+. They also found that R_0 tends to be conservative in some cases. In general, the authors recommend that all estimators be considered in practice. Also see Duval and Tweedie (2000b), Duval (2005), and Steichen (2016) for more details about these estimators.

Trim-and-fill algorithm

Without loss of generality, we assume that the observations, $\hat{\theta}_j$'s, are sorted in ascending order.

The steps of the trim-and-fill method below are based on Duval and Tweedie (2000b) and Steichen (2016).

Step 1. At the initial iteration $l = 1$, compute $\hat{\theta}^{(1)}$ using the meta-analysis model as declared with `meta set` or `meta esize`; the model may be changed using the `itermethod()` option.

Compute the centered values

$$\hat{\theta}_j^{(1)} = \hat{\theta}_j - \hat{\theta}^{(1)} \quad j = 1, \ldots, K$$

and estimate $\widehat{K}_0^{(1)}$ using the default L_0^+ estimator applied to the set of centered values $\hat{\theta}_j^{(1)}$; that is, set $X_j = \hat{\theta}_j^{(1)}$ in the previous section. Other estimators for $\widehat{K}_0^{(1)}$ may be specified using the `estimator()` option.

Step 2. At the iteration $l \geq 2$, remove $\widehat{K}_0^{(l-1)}$ values from the right end of the set of values $\hat{\theta}_j$, and estimate $\hat{\theta}^{(l)}$ based on the trimmed "symmetric" set of $K - \widehat{K}_0^{(l-1)}$ values: $\left\{\hat{\theta}_1, \ldots, \hat{\theta}_{K-\widehat{K}_0^{(l-1)}}\right\}$. Compute the next set of centered values

$$\hat{\theta}_j^{(l)} = \hat{\theta}_j - \hat{\theta}^{(l)} \quad j = 1, \ldots, K$$

Step 3. Repeat step 2 until an iteration $l = L$, at which $\widehat{K}_0^{(L)} = \widehat{K}_0^{(L-1)}$ (and thus, $\hat{\theta}^{(L)} = \hat{\theta}^{(L-1)}$). Set $\widehat{K}_0 = \widehat{K}_0^{(L)}$.

Step 4. Finally, compute the $\widehat{K}_0$ imputed symmetric values

$$\hat{\theta}_j^* = 2\hat{\theta}^{(L)} - \hat{\theta}_{K-j+1} \quad j = 1, \ldots, \widehat{K}_0$$

and the corresponding imputed within-study standard errors,

$$\hat{\sigma}_j^* = \hat{\sigma}_{K-j+1} \quad j = 1, \ldots, \widehat{K}_0$$

Compute the final overall effect-size estimate using the default meta-analysis method or `poolmethod()`, if specified, based on the "completed" dataset $\left\{\hat{\theta}_1, \ldots, \hat{\theta}_K, \hat{\theta}_1^*, \ldots, \hat{\theta}_{\widehat{K}_0}^*\right\}$.

References

Begg, C. B. 1997. Publication bias in meta-analysis: A Bayesian data-augmentation approach to account for issues exemplified in the passive smoking debate: Comment. *Statistical Science* 12: 241–244. https://doi.org/10.1214/ss/1030037958.

Deeks, J. J., J. P. T. Higgins, and D. G. Altman. 2017. "Addressing reporting biases". In *Cochrane Handbook for Systematic Reviews of Interventions Version 5.2.0*, edited by J. A. C. Sterne, M. Egger, and D. Moher. London: The Cochrane Collaboration. https://training.cochrane.org/handbook.

Duval, S. 2005. "The trim and fill method". In *Publication Bias in Meta-Analysis: Prevention, Assessment and Adjustments*, edited by H. R. Rothstein, A. J. Sutton, and M. Borenstein, 127–144. Chichester, UK: Wiley. https://doi.org/10.1002/0470870168.ch8.

Duval, S., and R. L. Tweedie. 2000a. Trim and fill: A simple funnel-plot–based method of testing and adjusting for publication bias in meta-analysis. *Biometrics* 56: 455–463. https://doi.org/10.1111/j.0006-341X.2000.00455.x.

———. 2000b. A nonparametric "trim and fill" method of accounting for publication bias in meta-analysis. *Journal of the American Statistical Association* 95: 89–98. https://doi.org/10.1080/01621459.2000.10473905.

National Research Council. 1992. *Combining Information: Statistical Issues and Opportunities for Research*. Washington, DC: National Academies Press. https://doi.org/10.17226/20865.

Peters, J. L., A. J. Sutton, D. R. Jones, K. R. Abrams, and L. Rushton. 2007. Performance of the trim and fill method in the presence of publication bias and between-study heterogeneity. *Statistics in Medicine* 26: 4544–4562. https://doi.org/10.1002/sim.2889.

Poole, C., and S. Greenland. 1999. Random-effects meta-analyses are not always conservative. *American Journal of Epidemiology* 150: 469–475. https://doi.org/10.1093/oxfordjournals.aje.a010035.

Rothstein, H. R., A. J. Sutton, and M. Borenstein, eds. 2005. *Publication Bias in Meta-Analysis: Prevention, Assessment and Adjustments*. Chichester, UK: Wiley. https://doi.org/10.1002/0470870168.

Steichen, T. J. 2000. sbe39: Nonparametric trim and fill analysis of publication bias in meta-analysis. *Stata Technical Bulletin* 57: 8–14. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 108–117. College Station, TX: Stata Press.

———. 2016. "Nonparametric trim and fill analysis of publication bias in meta-analysis". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 180–192. 2nd ed. College Station, TX: Stata Press.

Sutton, A. J., K. R. Abrams, D. R. Jones, T. A. Sheldon, and F. Song. 1998. Systematic reviews of trials and other studies. *Health Technology Assessment* 2(19): 1–276.

Also see

[META] **meta bias** — Tests for small-study effects in meta-analysis

[META] **meta data** — Declare meta-analysis data

[META] **meta funnelplot** — Funnel plots

[META] **meta summarize** — Summarize meta-analysis data+

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

meta meregress — Multilevel mixed-effects meta-regression

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
References | Also see

Description

`meta meregress` performs multilevel meta-regression. You can think of multilevel meta-regression as an extension of meta-regression, where effect sizes are nested within a higher grouping variable, such as district or region, and thus may be correlated. These groups may themselves be nested within another higher grouping variable, and so on. The dependencies among the observations within a group are accounted for by the inclusion of random effects at different levels of hierarchy.

If you wish to fit multilevel meta-analysis models with random intercepts only, see [META] **meta multilevel** for an alternative command with a simpler syntax.

`meta meregress` performs random-effects (RE) multilevel meta-regression with various covariance structures and estimation methods for the random effects, which include random intercepts and random coefficients. `meta meregress` is a stand-alone command in that it does not require you to declare your data as `meta` data using `meta set` or `meta esize`.

Quick start

Perform standard RE meta-analysis by expressing it as a two-level meta-analysis model of the effect-size `y` with random intercepts by `trial` and effect-size standard errors `se`

```
meta meregress y || trial:, essevariable(se)
```

As above, but perform an RE meta-regression on continuous moderator `x`

```
meta meregress y x || trial:, essevariable(se)
```

As above, but specify effect-size variances (`var`) instead of the effect-size standard errors

```
meta meregress y x || trial:, esvarvariable(var)
```

Perform a three-level meta-analysis of effect-size `y` with random intercepts by `region` and by `trial` nested within `region`

```
meta meregress y || region: || trial:, essevariable(se)
```

As above, but perform a three-level meta-regression on moderator `x`, add a random slope on `x` at the region level, and request the ML instead of the default REML estimation method

```
meta meregress y x || region: x || trial:, essevariable(se) mle
```

As above, but add a random slope on `x` at the trial-within-region level and specify an `exchangeable` covariance structure between the random slopes and intercepts at the trial-within-region level and an unstructured covariance structure between the random slopes and intercepts at the `region` level

```
meta meregress y x || region: x, covariance(unstructured) ///
   || trial: x, covariance(exchangeable) essevariable(se) mle
```

Perform a three-level meta-regression of y on x1 and x2 with random slopes for x1 and x2 at the region level, and specify a custom covariance structure for the random effects at the region level

```
matrix A = (.5,.,.a .,1,. .a,.,1)
meta meregress y x1 x2 || region: x1 x2, covariance(custom A) ///
            || trial:, essevariable(se)
```

Menu

Statistics > Meta-analysis

Syntax

```
meta meregress depvar fe_equation || re_equation [ || re_equation [...] ],
  { essevariable(varname) | esvarvariable(varname) } [ options ]
```

where the syntax of *fe_equation* is

```
[ indepvars ] [ if ] [ in ] [ , fe_options ]
```

and the syntax of *re_equation* is

```
levelvar: [ varlist ] [ , re_options ]
```

levelvar is a variable identifying the group structure for the random effects at that level. A random intercept is included in each *re_equation* unless option noconstant is specified and a random coefficient (also known as a random slope) associated with each variable in *varlist* is also added to the model.

fe_options	Description
Model	
noconstant	suppress constant term from the fixed-effects equation
constraints(*constraints*)	apply specified linear constraints

re_options	Description
Model	
covariance(*vartype*)	variance–covariance structure of the random effects
noconstant	suppress constant term from the random-effects equation
collinear	keep collinear variables

options	Description
Model	
* `essevariable(`*varname*`)`	specify effect-size (sampling) standard errors
* `esvarvariable(`*varname*`)`	specify effect-size (sampling) variances
`reml`	fit model via restricted maximum likelihood; the default
`mle`	fit model via maximum likelihood
Reporting	
`level(#)`	set confidence level; default is `level(95)`
`stddeviations`	show random-effects parameter estimates as standard deviations and correlations; the default
`variance`	show random-effects parameter estimates as variances and covariances
`estmetric`	show parameter estimates as stored in `e(b)`
`nohomtest`	suppress output for homogeneity test
`noretable`	suppress random-effects table
`nofetable`	suppress fixed-effects table
`noheader`	suppress output header
`nogroup`	suppress table summarizing groups
`nocnsreport`	do not display constraints
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
EM options	
`emiterate(#)`	number of EM iterations; default is `emiterate(20)`
`emtolerance(#)`	EM convergence tolerance; default is `emtolerance(1e-10)`
`emonly`	fit model exclusively using EM
`emlog`	show EM iteration log
`emdots`	show EM iterations as dots
Maximization	
maximize_options	control the maximization process; seldom used
`coeflegend`	display legend instead of statistics

* Either `essevariable()` or `esvarvariable()` is required.

indepvars and *varlist* may contain factor variables; see **[U] 11.4.3 Factor variables**.

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

`coeflegend` does not appear in the dialog box.

See **[U] 20 Estimation and postestimation commands** for more capabilities of estimation commands.

vartype	Description
independent	one unique standard-deviation parameter per random effect, all correlations 0; the default
exchangeable	equal standard deviations for random effects and one common pairwise correlation
identity	equal standard deviations for random effects; all correlations 0
unstructured	all standard deviations and correlations to be distinctly estimated
custom *matname*	custom matrix *matname* with fixed, free, and patterned standard deviations and correlations

Options

Model

noconstant suppresses the constant (intercept) term and may be specified for the fixed-effects equation and for any of or all the random-effects equations.

covariance(*vartype*) specifies the structure of the covariance matrix for the random effects and may be specified for each random-effects equation. *vartype* is one of the following: independent, exchangeable, identity, unstructured, or custom.

independent allows for a distinct standard deviation for each random effect within a random-effects equation and assumes that all correlations are 0. This is the default covariance structure.

exchangeable specifies one common standard deviation for all random effects and one common pairwise correlation.

identity is short for "multiple of the identity"; that is, all standard deviations are equal and all correlations are 0.

unstructured allows for all standard deviations and correlations to be distinct. If there are q random-effects terms, the unstructured covariance matrix will have $q(q+1)/2$ unique parameters.

custom *matname* specifies constraints for standard deviations (diagonal elements of *matname*) and correlations (off-diagonal elements of *matname*) of the random effects. Three types of specifications are allowed within *matname*:

1. A nonmissing value # that fixes the corresponding element at # during estimation.
2. One of .a, .b, etc., assigned to at least two diagonal or two off-diagonal elements to restrict the respective standard deviations or correlations to be the same during estimation.
3. A missing value . that allows the corresponding element to be freely estimated.

For example, assume that an *re_equation* in the model is || levelvar : x1 x2 x3 and therefore there are four random effects (one random intercept and three random slopes) at the levelvar level. Below, we describe the effect of specifying covariance(custom *matname*) with

$$matname = \begin{array}{c} \begin{array}{cccc} \texttt{x1} & \texttt{x2} & \texttt{x3} & \texttt{_cons} \end{array} \\ \begin{pmatrix} 1.2 & & & \\ 0.5 & \texttt{.a} & & \\ \texttt{.b} & \texttt{.} & \texttt{.a} & \\ \texttt{.b} & \texttt{.c} & \texttt{.c} & \texttt{.} \end{pmatrix} \end{array}$$

Let the u_j's be the random slopes of `xj`, `j` $\in \{1, 2, 3\}$ and u_0 be the random intercept. The above specification fixes the standard deviation of u_1 at 1.2 and the correlation between u_1 and u_2 at 0.5 during estimation. It also restricts the standard deviations of u_2 and u_3 to be equal (set equal to `.a`), the correlation between u_1 and u_3 to be the same as the correlation between u_1 and u_0 (both set equal to `.b`), and the correlation between u_2 and u_0 to be identical to the correlation between u_3 and u_0 (both set equal to `.c`). Furthermore, it allows the standard deviation of u_0 and the correlation between u_2 and u_3 to be freely estimated.

`essevariable(`*varname*`)` specifies a variable that stores the standard errors of the effect sizes in variable *varname*, also known as sampling standard errors. You must specify one of `essevariable()` or `esvarvariable()`.

`esvarvariable(`*varname*`)` specifies a variable that stores the variances of the effect sizes in variable *varname*, also known as sampling variances. You must specify one of `esvarvariable()` or `essevariable()`.

`reml` and `mle` specify the statistical method for fitting the model.

`reml`, the default, specifies that the model be fit using restricted maximum likelihood (REML), also known as residual maximum likelihood.

`mle` specifies that the model be fit using maximum likelihood (ML).

`constraints(`*constraints*`)`; see [R] **Estimation options**.

Reporting

`level(`#`)`; see [R] **Estimation options**.

`stddeviations`, the default, displays the random-effects parameter estimates as standard deviations and correlations.

`variance` displays the random-effects parameter estimates as variances and covariances.

`estmetric`; see [ME] **mixed**.

`nohomtest` suppresses the homogeneity test based on the Q_M statistic from the output.

`noretable`, `nofetable`, `noheader`, and `nogroup`; see [ME] **mixed**.

`nocnsreport`; see [R] **Estimation options**.

display_options: `noci`, `nopvalues`, `noomitted`, `vsquish`, `noemptycells`, `baselevels`, `allbaselevels`, `nofvlabel`, `fvwrap(`#`)`, `fvwrapon(`*style*`)`, `cformat(`%*fmt*`)`, `pformat(`%*fmt*`)`, `sformat(`%*fmt*`)`, and `nolstretch`; see [R] **Estimation options**.

EM options

`emiterate(`#`)`, `emtolerance(`#`)`, `emonly`, `emlog`, and `emdots`; see [ME] **mixed**.

Maximization

maximize_options: `difficult`, `technique(`*algorithm_spec*`)`, `iterate(`#`)`, `[no]log`, `trace`, `gradient`, `showstep`, `hessian`, `showtolerance`, `tolerance(`#`)`, `ltolerance(`#`)`, `nrtolerance(`#`)`, and `nonrtolerance`; see [R] **Maximize**. Those that require special mention for `meta meregress` are listed below.

For the `technique()` option, the default is `technique(nr)`. The `bhhh` algorithm is not available.

`matsqrt`, the default, and `matlog`; see [ME] **mixed**.

The following options are available with `meta meregress` but are not shown in the dialog box:

`collinear` specifies that `meta meregress` not omit collinear variables from the random-effects equation. Usually, there is no reason to leave collinear variables in place; in fact, doing so usually causes the estimation to fail because of the matrix singularity caused by the collinearity. However, with certain models (for example, a random-effects model with a full set of contrasts), the variables may be collinear, yet the model is fully identified because of restrictions on the random-effects covariance structure. In such cases, using the `collinear` option allows the estimation to take place with the random-effects equation intact.

`coeflegend`; see [R] **Estimation options**.

Remarks and examples

Remarks are presented under the following headings:

Introduction
Standard meta-analysis as a two-level model
Three-level random-intercepts model
Three-level model with random slopes
Using meta meregress
Examples of using meta meregress
Example 1: Standard meta-analysis as a two-level model
Example 2: Three-level meta-analysis
Example 3: Assessing multilevel heterogeneity
Example 4: Likelihood-ratio tests and information criteria
Example 5: Three-level meta-regression with random slopes
Example 6: Random-effects covariance structures
Example 7: Sensitivity multilevel meta-regression

Introduction

Multilevel meta-regression is a statistical technique used to study the relationship between effect sizes and covariates, where effect sizes may be correlated because of the clustered or multilevel (hierarchical) structure of the data. The multilevel structure can arise, for example, when we consider a meta-analysis that explores the impact of a new teaching technique on math testing scores. Studies may be conducted in separate school districts with potentially multiple studies in each school district. Each study reports an effect size that quantifies the difference between the two groups of students (those who received the new teaching technique and those who did not), such as mean difference of testing scores between the two groups. We are interested not only in synthesizing the overall effect of the new teaching technique but also in assessing the variability (heterogeneity) among the effect sizes at the district level (level 3) and among the studies within each district (level 2, also known as the studies-within-district level).

Results of studies conducted within the same school district are more likely to be similar and thus dependent given that, for example, the students therein are exposed to the same socioeconomical factors. This dependence is usually accounted for by including random effects at various levels of hierarchy in the model. By properly accounting for the dependence among the effect sizes, we can produce more accurate inference compared with performing a standard meta-analysis that ignores the hierarchical structure and the dependence among the effect sizes.

The standard meta-analysis can be viewed as a two-level meta-analysis model where the subjects or participants within studies are the level-1 observations and studies (or more precisely effect sizes reported by the studies) are the level-2 observations. The within-study standard errors or variances are assumed known; see *Standard meta-analysis as a two-level model*.

In the school example above, studies are the level-2 observations. However, this is not always the case in multilevel meta-analysis applications. For example, we may have a four-level meta-analysis with runs (level 2) nested within experiments (level 3) nested within studies (level 4); see McCurdy et al. (2020) for another example. Here studies actually define level 4 and runs define level 2. Thus, the terms "within-study standard errors" and "within-study variances" may not always be appropriate to refer to the variability at the lowest level of hierarchy, which is described by the standard errors or variances of the effect sizes. In our four-level example, the terms "within-run standard errors" and "within-run variances" would be more appropriate. To avoid any confusion, we will use the terms "sampling standard errors" and "sampling variances" to refer, respectively, to the standard errors and the variances of the effect sizes.

Multilevel meta-regression differs from standard meta-regression in two major aspects. First, a hierarchical (grouping) structure is assumed to be present in the data, and it is a main interest to decompose the total heterogeneity among the effect sizes across the different levels of hierarchy. Second, random slopes for moderators may be included in the model at different grouping levels. Recall that a standard meta-regression model incorporates only random intercepts.

Multilevel meta-regression is analogous to a multilevel mixed-effects model (Raudenbush and Bryk 2002), which is used when individual data are available, but in multilevel meta-regression, the outcome of interest is an effect size. And, because we do not have individual participant data, there are no covariates that are recorded at the lowest observation level. Also, the sampling variances, the variability at the lowest level, are assumed to be known. Having the known sampling variance allows us to include random intercepts at level 2. However, to include random slopes at level 2, the data must include repeated measures at this level; in the example of schools at level 2, we would need multiple effect sizes for each school to include random slopes at the school level. At level 3 and higher, the data will naturally have multiple lower-level groups nested within higher-level groups, so both random intercepts and random slopes can be included.

The covariates in multilevel meta-regression are known as moderators. Examples of moderators include study publication year, study test environment, and drug administration method. For a comprehensive introduction to multilevel meta-regression, see Goldstein et al. (2000); Thompson, Turner, and Warn (2001); Konstantopoulos (2011); Cheung (2014); and Sera et al. (2019).

Standard meta-analysis as a two-level model

The standard RE meta-analysis model (see [META] **meta summarize**) may be viewed as a special two-level meta-analysis where the subjects or the within-study observations (level 1) are nested within studies (level 2). These levels are

$$\begin{aligned} \text{Level 1 (within studies): } \hat{\theta}_j &= \theta_j + \epsilon_j \\ \text{Level 2 (between studies): } \theta_j &= \theta + u_j \end{aligned} \tag{1}$$

where $\epsilon_j \sim N\left(0, \hat{\sigma}_j^2\right)$ and $u_j \sim N\left(0, \tau^2\right)$. Here $\hat{\sigma}_j^2$ is the sampling variance (effect-size variance) for the jth study, which is assumed known (it is assumed to be estimated with adequate accuracy within each study, hence the hat notation). τ^2 is the variance of the random effects (the u_j's), also known as the between-study variance. The sampling errors (the ϵ_j's) and the random effects (the u_j's) are assumed to be independent. Similarly, the classical RE meta-regression (see [META] **meta regress**) can be obtained by incorporating moderators into (1) as follows:

$$\begin{aligned}\text{Level 1 (within studies): } \hat{\theta}_j &= \theta_j + \epsilon_j \\ \text{Level 2 (between studies): } \theta_j &= \beta_0 + \beta_1 x_{1,j} + \cdots + \beta_{p-1} x_{p-1,j} + u_j \\ &= \mathbf{x}_j\boldsymbol{\beta} + u_j\end{aligned}$$

where $\boldsymbol{\beta} = \left(\beta_0, \beta_1, \ldots, \beta_{p-1}\right)'$ is a $p \times 1$ vector of unknown regression (fixed-effects) coefficients.

Three-level random-intercepts model

Next we will discuss extensions of (1) to higher levels of hierarchy. Given its prevalence in practice, we will start by mathematically describing the three-level random-intercepts meta-analysis model with a single observation per level-2 group (level-2 groups have no repeated measures). The model can be expressed as

$$\begin{aligned}\text{Level 1 (within studies): } \hat{\theta}_{jk} &= \theta_{jk} + \epsilon_{jk} \\ \text{Level 2: } \theta_{jk} &= \theta_j + u_{jk}^{(2)} \\ \text{Level 3: } \theta_j &= \theta + u_j^{(3)}\end{aligned} \tag{2}$$

where $j = 1, 2, \ldots, M$, $k = 1, 2, \ldots, m_j$, $u_j^{(3)} \sim N(0, \tau_3^2)$, $u_{jk}^{(2)} \sim N(0, \tau_2^2)$, and $\epsilon_{jk} \sim N(0, \hat{\sigma}_{jk}^2)$, with the $\hat{\sigma}_{jk}^2$'s being known as sampling variances (or more generally within-level-2 variances). The $u_j^{(3)}$'s, $u_{jk}^{(2)}$'s, and ϵ_{jk}'s are independent, and τ_3^2 and τ_2^2 are the random-effects variances at the third and second levels, respectively. Model (2) assumes that there is one effect-size observation per group at level 2. This is the most common setting in practice. For the general setting that accounts for multiple observations per group at level 2, see *Three-level model with random slopes*.

In a single-equation notation, (2) can be written as

$$\hat{\theta}_{jk} = \theta + u_j^{(3)} + u_{jk}^{(2)} + \epsilon_{jk}$$

When we include a $1 \times p$ vector of moderators, $\mathbf{x}_{jk} = (1, x_{1,jk}, \ldots, x_{p-1,jk})$, the three-level meta-analysis model described in (2) becomes a three-level meta-regression model

$$\begin{aligned}\hat{\theta}_{jk} &= \beta_0 + \beta_1 x_{1,jk} + \cdots + \beta_{p-1} x_{p-1,jk} + u_j^{(3)} + u_{jk}^{(2)} + \epsilon_{jk} \\ &= \mathbf{x}_{jk}\boldsymbol{\beta} + u_j^{(3)} + u_{jk}^{(2)} + \epsilon_{jk}\end{aligned} \tag{3}$$

Model (3) includes only random intercepts. It does not include any random slopes for moderators $\mathbf{x}_{jk}$. The above model can be extended to more than three levels of hierarchy; see Sera et al. (2019) for details. If you would like to fit a model like (3) or its higher-level analogs, you can use the `meta multilevel` command, which has a simpler syntax than `meta meregress`.

Three-level model with random slopes

Incorporating random slopes at any level of hierarchy (other than level 1, where observations or individual participants are not available) requires repeated measures to be available at that level. For example, it is not possible to include random slopes at level 2 in meta-analysis for any of the moderators in (3) because there is one observation per group at that level. In that case, if you attempt to include random slopes at level 2, `meta meregress` will produce estimates of their standard deviations that are practically 0. If you are familiar with the concept of random slopes in the context of multilevel meta-analysis, then you may skip the rest of this section and go to *Using meta meregress*.

Below, we modify the notation used in (3) to introduce a third subscript, r, that accounts for the repeated measures at level 2. Assume there are q_l random effects (1 random intercept and $q_l - 1$ random slopes) at level $l = 2$ and $l = 3$; random slopes may then be introduced into (3) by writing

$$\widehat{\theta}_{jkr} = \mathbf{x}_{jkr}\boldsymbol{\beta} + \mathbf{z}^{(3)}_{jkr}\mathbf{u}^{(3)}_{j} + \mathbf{z}^{(2)}_{jkr}\mathbf{u}^{(2)}_{jk} + \epsilon_{jkr} \tag{4}$$

where $j = 1, 2, \ldots, M$, $k = 1, 2, \ldots, m_j$, and $r = 1, 2, \ldots, m_{jk}$. The subscript r was not needed in (3), because m_{jk} was assumed to equal 1. Here $\mathbf{x}_{jkr} = (1, x_{1,jkr}, \ldots, x_{p-1,jkr})$ is a $1 \times p$ vector of moderators associated with $\boldsymbol{\beta}$, and $\mathbf{z}^{(3)}_{jkr}$ is a $1 \times q_3$ vector of moderators associated with the level-3 $q_3 \times 1$ vector of random effects $\mathbf{u}^{(3)}_{j}$ (1 intercept and $q_3 - 1$ slopes), where $\mathbf{u}^{(3)}_{j} \sim N(\mathbf{0}, \boldsymbol{\Sigma}^{(3)})$. Similarly, $\mathbf{z}^{(2)}_{jkr}$ is a $1 \times q_2$ vector of moderators associated with the level-2 (within-level-3) $q_2 \times 1$ vector of random effects $\mathbf{u}^{(2)}_{jk}$, where $\mathbf{u}^{(2)}_{jk} \sim N(\mathbf{0}, \boldsymbol{\Sigma}^{(2)})$. The ϵ_{jkr}'s are the within-level-2 error terms following a $N(0, \hat{\sigma}^2_{jkr})$ distribution. $\boldsymbol{\Sigma}^{(3)}$ and $\boldsymbol{\Sigma}^{(2)}$ are the random-effects covariance matrices at levels 3 and 2, respectively.

The above model can be extended to more than three levels of hierarchy with the possibility to include random slopes at any level $l > 1$; see Sera et al. (2019) for details.

Using meta meregress

`meta meregress` fits various multilevel meta-regression models. Suppose variable `es` records effect sizes and variable `se` records the sampling standard errors for effect sizes.

Standard meta-analysis model as a two-level model. The standard RE meta-analysis model can be expressed as a two-level meta-analysis model. Suppose variable `study` stores study IDs; we can then fit a standard RE meta-analysis model using

```
. meta meregress es || study:, essevariable(se)
```

Recall that in meta-analysis, the sampling standard errors are treated as known. We specify them in the `essevariable()` option. If you have variances instead, you can specify them in the `esvarvariable()` option. The above specification should produce the same results as if we had typed `meta set es se` followed by `meta summarize, nostudies`; see [META] **meta summarize** and example 1.

Two-level meta-regression. Suppose we have two moderators, `x1` and `x2`. If we assume that the effects of moderators are constant across studies, we can fit a standard RE meta-regression as a two-level meta-regression without random coefficients (random slopes) for moderators:

```
. meta meregress es x1 x2 || study:, essevariable(se)
```

The above specification produces the same results as if we had typed `meta set es se` followed by `meta regress x1 x2`; see [META] **meta regress** and example 1.

Alternatively, we can allow the effects of moderators to vary across studies by including random slopes for the moderators:

```
. meta meregress es x1 x2 || study: x1, essevariable(se)
```

Recall that this is possible only if there are multiple observations (effect sizes) per study; otherwise, the estimated standard deviations of the random slopes will be estimated as zeros. In other words, in the context of standard meta-regression (where one effect size per study is reported), the above specification will produce a zero estimate for the standard deviation of random slopes; see *Three-level model with random slopes* for more details.

We can include random slopes for all or a subset of moderators by specifying the desired subset in the random-effects equation (the `|| study:` equation in our example).

Three-level meta-analysis model. Suppose we have schools (level 2) and each school records repeated observations on effect sizes. Also, suppose that the schools are nested within districts (level 3). We can incorporate potential dependence among the effect sizes within schools and within districts by fitting a three-level meta-analysis model with `district` as the top (third) level:

```
. meta meregress es || district: || school:, essevariable(se)
```

Three-level meta-regression. Continuing with our three-level school data, if we also have moderators, say, `x1` and `x2`, we can incorporate them in our three-level meta-analysis model in various ways.

We can specify them only in the fixed-effects equation, assuming their effects do not vary across districts or schools within districts:

```
. meta meregress es x1 x2 || district: || school:, essevariable(se)
```

We can specify them in all equations to allow them to vary within all levels:

```
. meta meregress es x1 x2 || district: x1 x2 || school: x1 x2, essevariable(se)
```

Or, if there is only one effect size reported per school, then we can eliminate the random slopes from the `school` level:

```
. meta meregress es x1 x2 || district: x1 || school:, essevariable(se)
```

For illustration, in the above we included random slopes only for `x1`.

Three-level meta-regression with various covariance structures. In the presence of random slopes, we can specify various covariance structures to model the dependencies between random effects at a specific level. By default, the random effects are assumed to be independent. This default is chosen out of computational feasibility, in case the model includes many random slopes. In practice, you will often want to verify that this assumption is reasonable for your data. You can do this by specifying other covariance structures such as `exchangeable`, `unstructured`, or `custom` *matname* in the `covariance()` option. For instance, we now assume an unstructured (completely unrestricted) covariance for the random effects at the `district` level:

```
. meta meregress es x1 x2 || district: x1, covariance(unstructured)
      || school:, essevariable(se)
```

In some applications, you may need to fix or constrain some elements of the random-effects variance–covariance matrix. This is also useful to perform sensitivity analysis; see example 7. You can do this by using the `custom` *matname* covariance structure.

Covariance structure `custom` *matname* provides a flexible way to restrict specific random-effects standard deviations and correlations during estimation while allowing the remaining parameters to be freely estimated. This option can be seen as a generalization of option `tau2()` in [META] **meta regress** and thus can be used to perform sensitivity analysis; see `covariance(custom` *matname*`)`.

Similarly, we can build other models. With more levels, we can specify different covariance structures at different levels of hierarchy:

```
. meta meregress es x1 x2 || state: x1 x2, covariance(unstructured)
      || district: x2 , covariance(exchangeable) || school: ,  essevariable(se)
```

By default, `meta meregress` uses the REML method to estimate model parameters. This method produces unbiased estimates of the random-effects covariance parameters by accounting for the loss of degrees of freedom from estimating the fixed-effects vector β. You can specify the `mle` option to instead estimate parameters using ML.

Examples of using meta meregress

Examples are presented under the following headings:

Example 1: Standard meta-analysis as a two-level model
Example 2: Three-level meta-analysis
Example 3: Assessing multilevel heterogeneity
Example 4: Likelihood-ratio tests and information criteria
Example 5: Three-level meta-regression with random slopes
Example 6: Random-effects covariance structures
Example 7: Sensitivity multilevel meta-regression

▷ Example 1: Standard meta-analysis as a two-level model

Recall the pupil IQ data (Raudenbush and Bryk 1985; Raudenbush 1984) described in *Effects of teacher expectancy on pupil IQ (pupiliq.dta)* of [META] **meta**. Here we will use its declared version (declared with `meta set`) to illustrate how to specify a standard meta-analysis model as a two-level random-intercepts model.

```
. use https://www.stata-press.com/data/r18/pupiliqset
(Effects of teacher expectancy on pupil IQ; set with -meta set-)
. meta query, short
-> meta set stdmdiff se , studylabel(studylbl) eslabel(Std. mean diff.)
  Effect-size label: Std. mean diff.
   Effect-size type: Generic
        Effect size: stdmdiff
          Std. err.: se
              Model: Random effects
             Method: REML
```

In these data, the effect sizes are standardized mean differences stored in variable `stdmdiff`, with their respective standard errors stored in variable `se`. To perform standard meta-analysis, we type the following, suppressing the individual study results for brevity:

```
. meta summarize, nostudies
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
        Study label: studylbl
Meta-analysis summary                           Number of studies =     19
Random-effects model                            Heterogeneity:
Method: REML                                                tau2 =  0.0188
                                                         I2 (%) =   41.84
                                                             H2 =    1.72
        theta: Overall Std. mean diff.
------------------------------------------------------------------------------
             |      Estimate   Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       theta |      .0836946   .0516536     1.62   0.105    -.0175447     .1849338
------------------------------------------------------------------------------
Test of homogeneity: Q = chi2(18) = 35.83                 Prob > Q = 0.0074
```

The overall effect-size estimate is 0.0837 with the standard error of 0.052, and the estimated between-study variance `tau2` is 0.0188.

The standard meta-analysis model for this dataset can be expressed as

$$\texttt{stdmdiff}_j = \theta + u_j + \epsilon_j \tag{5}$$

with $u_j \sim N(0, \tau^2)$ and $\epsilon_j \sim N(0, \texttt{se}_j^2)$. This model can be fit using `meta meregress` as follows.

We specify the response variable and the fixed-effects portion of the model by typing the outcome variable (`stdmdiff`) and some independent variables (moderators) of interest (in this example, there are no moderators) after the command `meta meregress`. We then type `|| study:` to specify random effects at the `study` level. We did not specify any variables after the colon (`:`), because we wanted to incorporate only random intercepts; see example 5 for random slopes. We also specify the variable containing the sampling standard errors using option `essevariable()`.

```
. meta meregress stdmdiff || study:, essevariable(se)

Performing EM optimization ...

Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -33.419194  (not concave)
Iteration 1:  Log restricted-likelihood = -10.213945  (not concave)
Iteration 2:  Log restricted-likelihood = -3.8361073
Iteration 3:  Log restricted-likelihood = -3.7393756
Iteration 4:  Log restricted-likelihood = -3.7365412
Iteration 5:  Log restricted-likelihood = -3.7365412

Computing standard errors ...

Multilevel REML meta-analysis                     Number of obs    =  19
Group variable: study                             Number of groups =  19
                                                  Obs per group:
                                                               min =   1
                                                               avg = 1.0
                                                               max =   1
                                                  Wald chi2(0)     =   .
Log restricted-likelihood = -3.7365412            Prob > chi2      =   .
```

stdmdiff	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
_cons	.0836939	.0516531	1.62	0.105	-.0175444	.1849322

```
Test of homogeneity: Q_M = chi2(18) = 35.83                  Prob > Q_M = 0.0074
```

Random-effects parameters	Estimate
study: Identity	
sd(_cons)	.1372184

The output shows information about the optimization algorithm, the iteration log, and the method (REML) used for estimating τ^2. There are 19 observations (effect sizes) and 19 groups (studies) with one observation per group, which is the case for standard meta-analysis. The reported model Wald test is missing because we do not have moderators in our model.

The first table displays the fixed-effect parameter estimate from the two-level meta-analysis. Here the fixed-effect parameter is a constant term denoted by `_cons`, which represents θ in (5) and `theta` in the output from `meta summarize`. The estimate of θ is 0.0837 with a standard error of 0.052 and the 95% CI of $[-0.0175,\ 0.1849]$. The test of homogeneity, which tests that all effect sizes are equal, reports the Q_{M} statistic of 35.83 with a p-value of 0.0074. The second table shows the estimated value of τ (standard deviation of the random effects u_j's) labeled as `sd(_cons)` in the output.

The results for the fixed-effect parameter are virtually identical. `meta summarize` reported an estimate of the variance of the random intercepts $\hat{\tau}^2 = 0.0188$, whereas `meta meregress` reports the standard deviation ($\hat{\tau} = 0.1372$) by default. We can display the variance by specifying the option `variance` on replay. We also use options `noheader` and `nofetable` to suppress the header and the fixed-effects table.

```
. meta meregress, variance noheader nofetable
```

Random-effects parameters	Estimate
study: Identity	
var(_cons)	.0188289

Alternatively, we could have used `estat sd, variance` to obtain the same output; see [META] **estat sd** and example 6.

Similarly, we can fit a standard meta-regression model as a two-level random-intercepts regression model. First, we use `meta regress` ([META] **meta regress**) to fit a standard meta-regression model:

```
. meta regress weeks
  Effect-size label: Std. mean diff.
        Effect size: stdmdiff
          Std. err.: se
Random-effects meta-regression                  Number of obs  =        19
Method: REML                                    Residual heterogeneity:
                                                           tau2 =  .01117
                                                         I2 (%) =   29.36
                                                             H2 =    1.42
                                                R-squared (%) =   40.70
                                                Wald chi2(1)   =    7.51
                                                Prob > chi2    =  0.0061
```

_meta_es	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
weeks	-.0157453	.0057447	-2.74	0.006	-.0270046	-.0044859
_cons	.1941774	.0633563	3.06	0.002	.0700013	.3183535

```
Test of residual homogeneity: Q_res = chi2(17) = 27.66   Prob > Q_res = 0.0490
```

Next we fit this same model using `meta meregress`. We simply list the moderator (`weeks`) in the fixed-effects portion of the model after the outcome variable `stdmdiff`.

```
. meta meregress stdmdiff weeks || study:, essevariable(se) variance
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -31.596287  (not concave)
Iteration 1:  Log restricted-likelihood = -8.6658459  (not concave)
Iteration 2:  Log restricted-likelihood = -1.1427859  (not concave)
Iteration 3:  Log restricted-likelihood = -.71416907
Iteration 4:  Log restricted-likelihood = -.71388211
Iteration 5:  Log restricted-likelihood = -.71388211
Computing standard errors ...
Multilevel REML meta-regression                 Number of obs    =     19
Group variable: study                           Number of groups =     19
                                                Obs per group:
                                                             min =      1
                                                             avg =    1.0
                                                             max =      1
                                                Wald chi2(1)     =   7.51
Log restricted-likelihood = -.71388211          Prob > chi2      = 0.0061
```

stdmdiff	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
weeks	-.0157453	.0057447	-2.74	0.006	-.0270046	-.0044859
_cons	.1941769	.0633561	3.06	0.002	.0700012	.3183527

```
Test of homogeneity: Q_M = chi2(17) = 27.66              Prob > Q_M = 0.0490
```

Random-effects parameters	Estimate
study: Identity	
var(_cons)	.011166

The estimates for the fixed-effects coefficients (reported in the first table) and τ^2 (labeled `var(_cons)` in the second table above) are almost the same as from `meta regress`.

◁

▷ Example 2: Three-level meta-analysis

Consider a dataset from Cooper, Valentine, and Melson (2003) on schools that modified their calendars without prolonging the school year. A version of this dataset was also analyzed by Konstantopoulos (2011) and will be used below. The dataset consists of 56 studies that were conducted in 11 school districts.

Some schools adopted modified calendars that feature shorter breaks more frequently throughout the year (for example, 12 weeks of school followed by 4 weeks off), as opposed to the traditional calendar with a longer summer break and shorter winter and spring breaks. The studies compared the academic achievement of students on a traditional calendar with those on a modified calendar. The effect size (`stmdiff`) was the standardized mean difference with positive values indicating higher achievement, on average, in the group on the modified calendar. The standard error (`se`) of `stmdiff` was also reported by each study. Let's first describe our dataset:

```
. use https://www.stata-press.com/data/r18/schoolcal, clear
(Effect of modified school calendar on student achievement)
. describe
Contains data from https://www.stata-press.com/data/r18/schoolcal.dta
 Observations:            56                  Effect of modified school
                                                calendar on student achievement
    Variables:             8                  19 Jan 2023 21:44
                                              (_dta has notes)
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
district        int     %12.0g                District ID
school          byte    %9.0g                 School ID
study           byte    %12.0g                Study ID
stdmdiff        double  %10.0g                Standardized difference in means
                                                of achievement test scores
var             double  %10.0g                Within-study variance of stdmdiff
year            int     %12.0g                Year of the study
se              double  %10.0g                Within-study standard-error of
                                                stdmdiff
year_c          byte    %9.0g                 Year of the study centered around
                                                1990
-------------------------------------------------------------------------------
Sorted by: district
```

Because the schools are nested within districts, we fit a three-level random-intercepts model. This model can also be fit using command `meta multilevel`; see example 1 of [META] **meta multilevel**. The model can be expressed as

$$\texttt{stdmdiff}_{jk} = \theta + u_j^{(3)} + u_{jk}^{(2)} + \epsilon_{jk} \tag{6}$$

with $u_j^{(3)} \sim N(0, \tau_3^2)$, $u_{jk}^{(2)} \sim N(0, \tau_2^2)$, and $\epsilon_{jk} \sim N(0, \texttt{se}_{jk}^2)$. Here there is one observation (effect size) reported per school (level-2 group). Fitting a three-level model requires that you specify two random-effects equations: one for level 3 (identified by variable `district`) and one for level 2 (identified by variable `school`). This model can be fit using `meta meregress` as follows:

```
. meta meregress stdmdiff || district: || school:, essevariable(se)
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood =  -104.8525  (not concave)
Iteration 1:  Log restricted-likelihood = -49.423271  (not concave)
Iteration 2:  Log restricted-likelihood = -25.793841  (not concave)
Iteration 3:  Log restricted-likelihood = -21.310018
Iteration 4:  Log restricted-likelihood = -9.1236345
Iteration 5:  Log restricted-likelihood = -8.2625776
Iteration 6:  Log restricted-likelihood = -7.9588561
Iteration 7:  Log restricted-likelihood = -7.9587239
Iteration 8:  Log restricted-likelihood = -7.9587239
Computing standard errors ...
Multilevel REML meta-analysis                          Number of obs = 56
        Grouping information
```

Group variable	No. of groups	Observations per group Minimum	Average	Maximum
district	11	3	5.1	11
school	56	1	1.0	1

```
                                                       Wald chi2(0)  =  .
Log restricted-likelihood = -7.9587239                 Prob > chi2   =  .
```

stdmdiff	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
_cons	.1847132	.0845559	2.18	0.029	.0189866	.3504397

```
Test of homogeneity: Q_M = chi2(55) = 578.86            Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
district: Identity	
sd(_cons)	.2550724
school: Identity	
sd(_cons)	.1809324

We first store the results of the model so we can use them later in example 4 to perform likelihood-ratio tests.

```
. estimates store main_model
```

As in example 1, our fixed-effects equation contains only the dependent variable (effect sizes `stdmdiff`). But we have two random-effects equations. The first represents random intercepts [the $u_j^{(3)}$'s in (6)] at the `district` level (level 3), and the second represents random intercepts [the $u_{jk}^{(2)}$'s in (6)] at the `school` level (level 2). The order in which these are specified (from left to right) is important—`meta meregress` assumes that `school` is nested within `district`. Below, we describe each portion of the output in detail.

The output first displays information about the optimization, including an iteration log. The top of the header shows the method (REML) used for estimation and also displays the total number of observations, which is 56 in our example.

The information on groups at different levels of hierarchy is displayed as a table with one row for each grouping (level of hierarchy). For example, there are 11 groups (districts) at the `district` level. Each group contains somewhere between 3 to 11 level-2 groups (schools). You can suppress this table with the `nogroup` or the `noheader` option, which will also suppress the rest of the header.

The second table displays the fixed-effects coefficients. In our example, there is only an intercept corresponding to the term θ in (6). The value of $\widehat{\theta}$ is 0.185 with a 95% CI of $[0.019, 0.35]$. This means that, on average, students following the modified school calendar achieved higher scores than those who did not.

The third table displays the random-effects parameters, traditionally known as variance components in the context of multilevel or mixed-effects models. The variance-component estimates are now organized and labeled according to each level. By default, `meta meregress` reports standard deviations of the random intercepts (and correlations if they existed in the model) at each level. But you can instead specify the `variance` option to report variances (and covariances if they existed in the model). We have $\widehat{\tau_3} = 0.255$ and $\widehat{\tau_2} = 0.181$. These values are the building blocks for assessing heterogeneity across different hierarchical levels and are typically interpreted in that context; see example 3 and *Higgins–Thompson heterogeneity statistics* in *Methods and formulas* in [META] **estat heterogeneity (me)** for details. In general, the higher the value of τ_l, the more heterogeneity is expected among the groups within level l.

◁

▷ Example 3: Assessing multilevel heterogeneity

Continuing with example 2, let's use the postestimation command `estat heterogeneity` to quantify the multilevel heterogeneity among the effect sizes captured by the three-level meta-analysis model.

```
. estat heterogeneity

Method: Cochran
Joint:
  I2 (%) = 90.50

Method: Higgins-Thompson
district:
  I2 (%) = 63.32

school:
  I2 (%) = 31.86

Total:
  I2 (%) = 95.19
```

Cochran's I^2_{Q} quantifies the amount of heterogeneity jointly for all levels of hierarchy. It is a direct extension to the multilevel setting of the classical I^2 statistic based on the DerSimonian–Laird method and thus has the same interpretation. For instance, $I^2_{\text{Q}} = 90.50\%$ means that 90.50% of the variability among the effect sizes is due to true heterogeneity in our data as opposed to the sampling variability. See *Heterogeneity measures* in *Methods and formulas* in [META] **meta summarize** and *Residual heterogeneity measures* in *Methods and formulas* in [META] **meta regress** for details.

The value of the Cochran statistic is the same for all multilevel models with the same fixed-effects structure. This is because its computation is based on the Cochran multivariate Q statistic, which is calculated based only on the fixed-effects model; see *Cochran heterogeneity statistic* in *Methods and formulas* in [META] **estat heterogeneity (me)** for details.

Unlike the Cochran I^2_{Q} statistic, the multilevel Higgins–Thompson I^2 statistics (Nakagawa and Santos 2012) provide ways to assess the contribution of each level of hierarchy to the total heterogeneity, in addition to their joint contribution. For example, between-schools heterogeneity or heterogeneity within

districts (level-2 heterogeneity) is the lowest, accounting for about 32% of the total variation in our data, whereas between-districts heterogeneity (level-3 heterogeneity) accounts for about 63% of the total variation. This is a direct consequence of the estimate of τ^2_3 being greater than that of τ^2_2 in example 2. See *Higgins–Thompson heterogeneity statistics* in *Methods and formulas* in [META] **estat heterogeneity (me)** for details.

◁

▷ Example 4: Likelihood-ratio tests and information criteria

Suppose we wish to test whether there is a nonnegligible amount of heterogeneity within districts (that is, heterogeneity between the schools within a district). This amounts to testing H_0: $\tau^2_2 = 0$. We need to fit a model with $\tau^2_2 = 0$ and compare it with the model from example 2. This is a two-level model with `district` as the second level of hierarchy (we eliminate the `school` level). We fit this model and store its results under the name `school_effect`. Recall that we had already saved our results for the three-level model in example 2 under the name `main_model`. So we can use the `lrtest` command to conduct a likelihood-ratio test of our H_0.

```
. quietly meta meregress stdmdiff || district: , essevariable(se)
. estimates store school_effect
. lrtest main_model school_effect
Likelihood-ratio test
Assumption: school_effect nested within main_model
 LR chi2(1) =  48.52
Prob > chi2 = 0.0000
Note: The reported degrees of freedom assumes the null hypothesis is not on
      the boundary of the parameter space. If this is not true, then the
      reported test is conservative.
Note: LR tests based on REML are valid only when the fixed-effects
      specification is identical for both models.
```

Because the null hypothesis value of τ^2_2 is at the boundary of the parameter space, the asymptotic distribution of the test statistic is a mixture of the χ^2_0 (a point mass at zero) and χ^2_1 distributions (Verbeke and Molenberghs 2000; Self and Liang 1987; and Gutierrez, Carter, and Drukker 2001), with each having an equal weight of 0.5. To elaborate on the first note reported by `lrtest`, the exact p-value can therefore be computed as

$$p = 0.5 \times P(\chi^2_0 > 48.52) + 0.5 \times P(\chi^2_1 > 48.52) = 0.5 \times P(\chi^2_1 > 48.52)$$

which is half of what is reported above. The second equality holds because the χ^2 distribution with zero degrees of freedom, χ^2_0, places all probability mass at zero, and therefore $0.5 \times P(\chi^2_0 > 48.52) = 0$. This updated p-value computation does not affect our conclusion regarding the test result, which is that we reject the hypothesis that schools are homogeneous within districts.

Similarly, we may also wish to test whether there is a nonnegligible amount of heterogeneity between districts, which amounts to testing H_0: $\tau^2_3 = 0$. This is equivalent to fitting a standard RE meta-analysis where all 56 effect sizes are assumed independent. Hence, we use variable `study` as the grouping level in our model specification. Had we used `school`, the model would have clustered our 56 effect sizes into 11 groups, which would violate the independence assumption.

```
. quietly meta meregress stdmdiff || study: , essevariable(se)
. estimates store dist_effect
. lrtest main_model dist_effect
Likelihood-ratio test
Assumption: dist_effect nested within main_model
 LR chi2(1) =  17.77
Prob > chi2 = 0.0000
Note: The reported degrees of freedom assumes the null hypothesis is not on
      the boundary of the parameter space. If this is not true, then the
      reported test is conservative.
Note: LR tests based on REML are valid only when the fixed-effects
      specification is identical for both models.
```

The results of the test provide strong evidence that there is significant between-districts heterogeneity. Similar discussion applies to the computation of the exact p-value as above.

We can compare our models using information criteria by using the `estimates stats` command. We use option `all` to request AICc and CAIC in addition to the default AIC and BIC. We also use option `n()` to use $n - p = 55$ instead of $n = 56$ as the number of observations in the computation of BIC, AICc, and CAIC because our models used REML estimation.

```
. estimates stats main_model dist_effect school_effect, all n(55)
Information criteria
```

Model	N	ll(null)	ll(model)	df
main_model	55	.	-7.958724	3
dist_effect	55	.	-16.8455	2
school_eff~t	55	.	-32.21648	2

Model	AIC	BIC	AICc	CAIC
main_model	21.91745	27.93945	22.38804	30.93945
dist_effect	37.691	41.70566	37.92177	43.70566
school_eff~t	68.43295	72.44762	68.66372	74.44762

```
Legend: AIC is Akaike's information criterion.
        BIC is Bayesian information criterion.
        AICc is corrected Akaike's information criterion.
        CAIC is consistent Akaike's information criterion.
```

All measures of information criteria favor the three-level model `main_model`.

◁

▷ Example 5: Three-level meta-regression with random slopes

For illustration purposes, we will use variable `year_c` to conduct a three-level meta-regression and include random slopes (corresponding to variable `year_c`) at the `district` level. We will not include random slopes at the `school` level, because there is only one observation (effect size) per school; otherwise, we will get an estimate that is practically zero for the standard deviation of the random slope of `year_c` at the `school` level; see *Three-level model with random slopes*. The model can be described as follows:

$$\texttt{stdmdiff}_{jk} = \beta_0 + \beta_1 \texttt{year_c}_{jk} + u^{(3)}_{0j} + u^{(3)}_{1j} \texttt{year_c}_{jk} + u^{(2)}_{jk} + \epsilon_{jk} \qquad (7)$$

with $(u_{0j}^{(3)}, u_{1j}^{(3)})' \sim N(\mathbf{0}, \mathbf{\Sigma}^{(3)})$, $u_{jk}^{(2)} \sim N(0, \tau_2^2)$, and $\epsilon_{jk} \sim N(0, \texttt{var}_{jk})$. By default, the 2×2 matrix $\mathbf{\Sigma}^{(3)}$ is assumed diagonal, which means that the $u_{0j}^{(3)}$'s and $u_{1j}^{(3)}$'s are assumed independent. Other covariance structures can be specified with the `covariance()` option; see example 6.

```
. meta meregress stdmdiff year_c || district: year_c || school:,
> esvarvariable(var)

Performing EM optimization ...

Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -101.95646  (not concave)
Iteration 1:  Log restricted-likelihood = -94.506522  (not concave)
Iteration 2:  Log restricted-likelihood = -27.473266  (not concave)
Iteration 3:  Log restricted-likelihood = -9.8063483
Iteration 4:  Log restricted-likelihood = -7.2135276
Iteration 5:  Log restricted-likelihood =  -7.210109  (not concave)
Iteration 6:  Log restricted-likelihood = -7.2100808  (not concave)
Iteration 7:  Log restricted-likelihood = -7.2100609  (not concave)
Iteration 8:  Log restricted-likelihood = -7.2098538
Iteration 9:  Log restricted-likelihood = -7.2095961
Iteration 10: Log restricted-likelihood = -7.2095347
Iteration 11: Log restricted-likelihood = -7.2095302
Iteration 12: Log restricted-likelihood = -7.2095301

Computing standard errors ...

Multilevel REML meta-regression                     Number of obs =     56
        Grouping information
------------------------------------------------------------------------
                        |     No. of       Observations per group
 Group variable         |    groups    Minimum    Average    Maximum
------------------------+-----------------------------------------------
               district |          11          3        5.1         11
                 school |          56          1        1.0          1
------------------------------------------------------------------------
                                                    Wald chi2(1)  =   0.55
Log restricted-likelihood = -7.2095301              Prob > chi2   = 0.4577
------------------------------------------------------------------------------
    stdmdiff | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
      year_c |   .0096026   .0129302     0.74   0.458    -.0157402    .0349453
       _cons |   .1609574   .0823083     1.96   0.051    -.0003639    .3222787
------------------------------------------------------------------------------
Test of homogeneity: Q_M = chi2(54) = 550.26              Prob > Q_M = 0.0000

------------------------------------------------------------------------------
  Random-effects parameters |   Estimate
----------------------------+-------------------------------------------------
district: Independent       |
                  sd(year_c) |   .033531
                   sd(_cons) |  .0643482
----------------------------+-------------------------------------------------
school: Identity            |
                   sd(_cons) |  .1808125
------------------------------------------------------------------------------
```

The estimate of the regression coefficient of variable `year_c` is 0.010 with a 95% CI of $[-0.016, 0.035]$. We do not see any evidence for the association between `stdmdiff` and `year_c` ($p = 0.458$). The estimates of the standard deviations of $u_{1j}^{(3)}$ and $u_{0j}^{(3)}$ (at the `district` level) are labeled in the output as `sd(year_c)` and `sd(_cons)` and are estimated to be 0.034 and 0.064, respectively. These values are the estimates of the square root of the diagonal elements of $\mathbf{\Sigma}^{(3)}$. The covariance

structure at the district level is labeled as Independent, which is the default assumption. You may display the 2×2 matrix $\boldsymbol{\Sigma}^{(3)}$ using the estat recovariance command; see example 6. The estimate of τ_2 is 0.181.

◁

Although year_c did not explain the heterogeneity, we continue to include it as a moderator in our subsequent examples (example 6 and example 7) for illustration purposes.

▷ Example 6: Random-effects covariance structures

Continuing with example 5, we will explore different random-effects covariance structures for $\boldsymbol{\Sigma}^{(3)}$ instead of the default independent structure. The default independent covariance structure is chosen out of computational feasibility. In multilevel modeling, it is important to start with an unrestricted covariance first, whenever feasible. It is also important to have meaningful baseline values for the moderators to make variance components interpretable. Here we include year_c, which is centered on 1990, instead of year, so that the intercept can be interpreted as the expected value in 1990 and the variance components can also be interpreted relative to this year. Let's specify the covariance(unstructured) option first. This assumes that all random effects have distinct standard deviations and correlations. We suppress the header and the iteration log and display results with 3 decimal points using the noheader, nolog, and cformat(%9.3f) options, which we store in the local macro 'options' for syntactical convenience.

```
. local options noheader nolog cformat(%9.3f)
. meta meregress stdmdiff year_c || district: year_c,
> covariance(unstructured) || school:, esvarvariable(var) 'options'
```

stdmdiff	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
year_c	0.007	0.010	0.71	0.479	-0.013	0.028
_cons	0.160	0.076	2.12	0.034	0.012	0.308

Test of homogeneity: Q_M = chi2(54) = 550.26 Prob > Q_M = 0.0000

Random-effects parameters	Estimate
district: Unstructured	
sd(year_c)	0.028
sd(_cons)	0.082
corr(year_c,_cons)	1.000
school: Identity	
sd(_cons)	0.180

The random-effects covariance structure at the district level is now labeled Unstructured:. The correlation between the random slope and the random intercept is labeled as corr(year_c, _cons). The estimated correlation value is 1 because, as we mentioned in example 5, variable year_c did not explain any heterogeneity and was included here for illustration purposes only.

Instead of specifying one of the standard covariance structures (independent, identity, exchangeable, or unstructured), you may request a custom covariance structure where you can fix specific standard deviations or correlations while allowing others to be estimated. For example, the following matrix A fixes the correlation between $u_{0j}^{(3)}$ and $u_{1j}^{(3)}$ at 0.5 and allows for their standard deviations to be estimated from the data. See covariance(custom *matname*) for details.

```
. matrix A = (. ,.5 \ .5 ,.)

. meta meregress stdmdiff year_c || district: year_c, covariance(custom A)
> || school:, esvarvariable(var) 'options'
```

stdmdiff	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
year_c	0.007	0.011	0.67	0.500	-0.014	0.028
_cons	0.170	0.082	2.08	0.038	0.010	0.330

Test of homogeneity: Q_M = chi2(54) = 550.26 Prob > Q_M = 0.0000

Random-effects parameters	Estimate
district: Custom	
sd(year_c)	0.026
sd(_cons)	0.116
corr(year_c,_cons)	0.500*
school: Identity	
sd(_cons)	0.180

(*) fixed during estimation

Note the asterisk that is appended next to the `corr(year_c, _cons)` value to emphasize that it was fixed during estimation.

You may additionally wish to constrain the two standard deviations of the random intercept and random slope to be the same (both specified equal to `.a`):

```
. matrix B = (.a ,.5 \ .5 ,.a)

. meta meregress stdmdiff year_c || district: year_c, covariance(custom B)
> || school:, esvarvariable(var) 'options'
```

stdmdiff	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
year_c	0.010	0.012	0.79	0.427	-0.014	0.034
_cons	0.154	0.076	2.02	0.043	0.005	0.304

Test of homogeneity: Q_M = chi2(54) = 550.26 Prob > Q_M = 0.0000

Random-effects parameters	Estimate
district: Custom	
sd(year_c _cons)	0.033
corr(year_c,_cons)	0.500*
school: Identity	
sd(_cons)	0.181

(*) fixed during estimation

We can display the random-effects covariance matrices $\boldsymbol{\Sigma}^{(3)}$ (at the `district` level) and $\boldsymbol{\Sigma}^{(2)}$ (at the `school` level), which is a scalar in our example, using the `estat recovariance` command ([META] **estat recovariance**). This is particularly useful if we specify a complicated custom covariance structure in our model using the `covariance(custom` *matname*`)` option (think 3×3 or larger covariance matrices).

```
. estat recovariance
Random-effects covariance matrix for level district
             |    year_c      _cons
-------------+----------------------
      year_c |  .0010852
       _cons |  .0005426   .0010852
Fixed parameter: corr(year_c,_cons)=.5.
Random-effects covariance matrix for level school
             |     _cons
-------------+----------
       _cons |  .0326401
```

To see the corresponding correlation matrix, specify the `correlation` option.

You may also use `estat sd` ([META] **estat sd**) to display the variance-components parameters as variances and covariances (instead of the default standard deviations and correlations). This will also group together any parameters that were constrained to be the same.

```
. estat sd, variance
------------------------------------------------------------------------------
  Random-effects parameters  |   Estimate
-----------------------------+------------------------------------------------
district: Custom             |
           var(year_c _cons) |    .0010852
           cov(year_c,_cons) |    .0005426
-----------------------------+------------------------------------------------
school: Identity             |
                 var(_cons)  |    .0326401
------------------------------------------------------------------------------
Note: corr(year_c,_cons)=.5 at district
      level fixed during estimation.
```

◁

▷ Example 7: Sensitivity multilevel meta-regression

It is quite common in multilevel meta-regression to produce unstable estimates, especially when the number of observations is small relative to the number of parameters to be estimated. In this case, our goal may shift toward assessing the impact of different magnitudes of random-effects covariance parameters on the estimates of regression coefficients to evaluate the robustness of our results.

Continuing with (7) from example 5, we can investigate the effect of no correlation, moderate correlation (0.4), and high correlation (0.8) between the random intercepts (the $u_{0j}^{(3)}$'s) and the random slopes (the $u_{1j}^{(3)}$'s) at the `district level` on the regression coefficient estimates. We will allow for the random-effects standard deviations to be estimated from the data. Thus, our fixed custom random-effects covariance matrices for the three scenarios are

```
. matrix Sigma1 = (.,0\0,.)
. matrix Sigma2 = (.,.4\.4,.)
. matrix Sigma3 = (.,.8\.8,.)
```

We fit the first model using the correlations of 0 and store the estimation results as `corr0`.

```
. quietly meta meregress stdmdiff year_c
> || district: year_c, covariance(custom Sigma1)
> || school:, esvarvariable(var)
. estimates store corr0
```

Next we fit the model with correlations of 0.4 and store results as corr4 and the model with correlations of 0.8 and store results as corr8. For brevity, we suppressed the output from all commands by running them quietly.

```
. quietly meta meregress stdmdiff year_c
> || district: year_c, covariance(custom Sigma2)
> || school:, esvarvariable(var)
. estimates store corr4

. quietly meta meregress stdmdiff year_c
> || district: year_c, covariance(custom Sigma3)
> || school:, esvarvariable(var)
. estimates store corr8
```

We compare the estimates side by side by using estimates table:

```
. estimates table corr0 corr4 corr8,
> keep(stdmdiff:year_c stdmdiff:_cons) b(%8.3f) se(%8.3f)
```

Variable	corr0	corr4	corr8
year_c	0.006	0.007	0.007
	0.011	0.011	0.011
_cons	0.181	0.172	0.164
	0.090	0.083	0.078

Legend: b/se

As the correlation between the random intercepts and the random slopes at the district level increases, the coefficient estimate for _cons decreases. Also, the estimate becomes more precise (has a smaller standard error) as the correlation increases. Note also how the various magnitudes of correlations had little to no impact on the estimation of year_c (all values are near 0) because, as we saw in example 5, variable year_c did not explain any heterogeneity and should have been excluded from the model.

◁

Stored results

meta meregress stores the following in e():

Scalars

e(N)	total number of observations
e(k)	number of parameters
e(k_f)	number of fixed-effects parameters
e(k_r)	number of random-effects parameters
e(k_rs)	number of variances
e(k_rc)	number of covariances
e(ll)	log (restricted) likelihood
e(rank)	rank of e(V)
e(ic)	number of iterations
e(df_m)	model degrees of freedom
e(chi2)	model χ^2 Wald test statistic
e(p)	p-value for model test
e(Q_M)	multilevel Cochran Q_M residual homogeneity test statistic
e(df_Q_M)	degrees of freedom for residual homogeneity test
e(p_Q_M)	p-value for residual homogeneity test
e(converged)	1 if converged, 0 otherwise

Macros

e(cmd)	meta meregress
e(cmdline)	command as typed
e(method)	REML or ML
e(title)	title in estimation output
e(chi2type)	Wald; type of model χ^2 test
e(depvar)	name of dependent variable
e(ivars)	grouping variables
e(indepvars)	names of independent variables (moderators)
e(esvarvariable)	variable containing sampling variances (when esvarvariable() is specified)
e(essevariable)	variable containing sampling standard errors (when essevariable() is specified)
e(redim)	random-effects dimensions
e(vartypes)	variance-structure types
e(revars)	random-effects covariates
e(technique)	maximization technique
e(datasignature)	the checksum
e(datasignaturevars)	variables used in calculation of checksum
e(emonly)	emonly, if specified
e(ml_method)	type of ml method
e(opt)	type of optimization
e(optmetric)	matsqrt or matlog; random-effects matrix parameterization
e(properties)	b V
e(predict)	program used to implement predict
e(estat_cmd)	program used to implement estat
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices

e(b)	coefficient vector
e(V)	variance–covariance matrix of the estimators
e(cov_cust_#)	custom random-effects covariance matrix (when covariance(custom *matname*) is specified)
e(Cns)	constraints matrix
e(N_g)	group counts
e(g_min)	group-size minimums
e(g_avg)	group-size averages
e(g_max)	group-size maximums

Functions

e(sample)	marks estimation sample

In addition to the above, the following is stored in r():

Matrices

r(table)	matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

When the esvarvariable() option is specified, meta meregress creates a system variable, _meta_mereg_se, that contains the sampling standard errors.

Methods and formulas

Methods and formulas are presented under the following headings:

Three-level meta-regression
Methods for estimating $\boldsymbol{\Sigma}^{(2)}$ and $\boldsymbol{\Sigma}^{(3)}$
Random-effects covariance structures
Multilevel meta-analysis
Residual homogeneity test

For an overview of the statistical models behind multilevel meta-regression, see Konstantopoulos (2011) and Sera et al. (2019).

Three-level meta-regression

The model for the three-level meta-regression can be expressed as

$$\hat{\theta}_{jkr} = \mathbf{x}_{jkr}\boldsymbol{\beta} + \mathbf{z}^{(3)}_{jkr}\mathbf{u}^{(3)}_{j} + \mathbf{z}^{(2)}_{jkr}\mathbf{u}^{(2)}_{jk} + \epsilon_{jkr}$$

for $j = 1, 2, \ldots, M$, $k = 1, 2, \ldots, m_j$, and $r = 1, 2, \ldots, m_{jk}$. In this case, $\mathbf{x}_{jkr} = (1, x_{1,jkr}, \ldots, x_{p-1,jkr})$ is a $1 \times p$ vector of moderators and $\boldsymbol{\beta}$ is the corresponding $p \times 1$ vector of unknown fixed-effects parameters. $\mathbf{z}^{(3)}_{jkr}$ is a $1 \times q_3$ vector of moderators associated with the level-3 $q_3 \times 1$ vector of random effects $\mathbf{u}^{(3)}_{j}$ (1 intercept and $q_3 - 1$ slopes), where $\mathbf{u}^{(3)}_{j} \sim N(\mathbf{0}, \boldsymbol{\Sigma}^{(3)})$. Similarly, $\mathbf{z}^{(2)}_{jkr}$ is a $1 \times q_2$ vector of moderators associated with the level-2 (within-level-3) $q_2 \times 1$ vector of random effects $\mathbf{u}^{(2)}_{jk}$, where $\mathbf{u}^{(2)}_{jk} \sim N(\mathbf{0}, \boldsymbol{\Sigma}^{(2)})$. $\epsilon_{jkr} \sim N(0, \hat{\sigma}^2_{jkr})$ with the $\hat{\sigma}^2_{jkr}$'s being the sampling variances.

Define the $m_{jk} \times p$ matrix $\mathbf{X}_{jk} = (\mathbf{x}'_{jk1}, \mathbf{x}'_{jk2}, \ldots, \mathbf{x}'_{jkm_{jk}})'$ and the $m_{jk} \times 1$ vectors $\hat{\boldsymbol{\theta}}_{jk} = (\hat{\theta}_{jk1}, \hat{\theta}_{jk2}, \ldots, \hat{\theta}_{jkm_{jk}})'$ and $\boldsymbol{\epsilon}_{jk} = \left(\epsilon_{jk1}, \epsilon_{jk2}, \ldots, \epsilon_{jkm_{jk}}\right)'$. The above model can now be written as

$$\hat{\boldsymbol{\theta}}_{jk} = \mathbf{X}_{jk}\boldsymbol{\beta} + \mathbf{Z}^{(3)}_{jk}\mathbf{u}^{(3)}_{j} + \mathbf{Z}^{(2)}_{jk}\mathbf{u}^{(2)}_{jk} + \boldsymbol{\epsilon}_{jk}$$

where $m_{jk} \times q_3$ matrix $\mathbf{Z}^{(3)}_{jk} = (\mathbf{z}^{(3)'}_{jk1}, \mathbf{z}^{(3)'}_{jk2}, \ldots, \mathbf{z}^{(3)'}_{jkm_{jk}})'$ and $m_{jk} \times q_2$ matrix $\mathbf{Z}^{(2)}_{jk} = (\mathbf{z}^{(2)'}_{jk1}, \mathbf{z}^{(2)'}_{jk2}, \ldots, \mathbf{z}^{(2)'}_{jkm_{jk}})'$. The $\boldsymbol{\epsilon}_{jk}$'s have an m_{jk}-variate normal distribution with zero mean vector and a diagonal $m_{jk} \times m_{jk}$ covariance matrix $\mathrm{Var}(\boldsymbol{\epsilon}_{jk}) = \boldsymbol{\Lambda}_{jk}$ with diagonal elements $\hat{\sigma}^2_{jkr}$, $r = 1, 2, \ldots, m_{jk}$. The covariance matrices (the $\boldsymbol{\Lambda}_{jk}$'s) are treated as known and do not require estimation. The $\boldsymbol{\Lambda}_{jk}$'s reduce to $\hat{\sigma}^2_{j}$ in the case of standard meta-analysis; see *Methods and formulas* of [META] **meta summarize**.

Let $m_{j.} = \sum_{k=1}^{m_j} m_{jk}$ be the number of observations belonging to the jth level-3 group and define the $m_{j.} \times p$ matrix $\mathbf{X}_j = (\mathbf{X}'_{j1}, \mathbf{X}'_{j2}, \ldots, \mathbf{X}'_{jm_j})'$ and the $m_{j.} \times 1$ vectors $\hat{\boldsymbol{\theta}}_j = (\hat{\boldsymbol{\theta}}'_{j1}, \hat{\boldsymbol{\theta}}'_{j2}, \ldots, \hat{\boldsymbol{\theta}}'_{jm_j})'$ and $\boldsymbol{\epsilon}_j = (\boldsymbol{\epsilon}'_{j1}, \boldsymbol{\epsilon}'_{j2}, \ldots, \boldsymbol{\epsilon}'_{jm_j})'$ with $m_{j.} \times m_{j.}$ covariance matrix $\mathrm{Var}(\boldsymbol{\epsilon}_j) = \boldsymbol{\Lambda}_j = \oplus_{k=1}^{m_j}\boldsymbol{\Lambda}_{jk}$, where $\oplus$ is the Kronecker sum. The previous model can now be expressed as

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \mathbf{Z}^{(3)}_{j}\mathbf{u}^{(3)}_{j} + \mathbf{Z}^{(2)}_{j}\mathbf{u}^{(2)}_{j} + \boldsymbol{\epsilon}_j$$

where $m_{j.} \times m_j q_2$ block-diagonal matrix $\mathbf{Z}^{(2)}_{j} = \oplus_{k=1}^{m_j}\mathbf{Z}^{(2)}_{jk}$, $m_{j.} \times q_3$ matrix $\mathbf{Z}^{(3)}_{j} = (\mathbf{Z}^{(3)'}_{j1}, \mathbf{Z}^{(3)'}_{j2}, \ldots, \mathbf{Z}^{(3)'}_{jm_j})$, and $m_j q_2 \times 1$ vector of random effects at level 2 $\mathbf{u}^{(2)}_{j} = (\mathbf{u}^{(2)'}_{j1}, \mathbf{u}^{(2)'}_{j2}, \ldots, \mathbf{u}^{(2)'}_{jm_j})' \sim N(\mathbf{0}, \mathbf{I}_{m_j} \otimes \boldsymbol{\Sigma}^{(2)})$, where $\otimes$ is the Kronecker product.

We may eliminate the explicit reference to specific levels of hierarchy and express the previous model more compactly as

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \mathbf{Z}_j\mathbf{u}_j + \boldsymbol{\epsilon}_j$$

where $m_{j.} \times (q_3 + m_j q_2)$ matrix $\mathbf{Z}_j = (\mathbf{Z}_j^{(3)}, \mathbf{Z}_j^{(2)})$ and $(q_3 + m_j q_2) \times 1$ vector $\mathbf{u}_j = (\mathbf{u}_j^{(3)\prime}, \mathbf{u}_j^{(2)\prime})'$, with a $(q_3 + m_j q_2) \times (q_3 + m_j q_2)$ covariance matrix $\boldsymbol{\Sigma}_j$,

$$\boldsymbol{\Sigma}_j = \text{Var}\left(\mathbf{u}_j\right) = \begin{bmatrix} \boldsymbol{\Sigma}^{(3)} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{m_j} \otimes \boldsymbol{\Sigma}^{(2)} \end{bmatrix}$$

Note that $\boldsymbol{\Sigma}_j$ depends on j only through its dimension. In other words, if estimates for $\boldsymbol{\Sigma}^{(2)}$ and $\boldsymbol{\Sigma}^{(3)}$ are available, then estimates for $\boldsymbol{\Sigma}_j$, $j = 1, 2, \ldots, M$ are also available.

Let $\widehat{\boldsymbol{\Sigma}}_j$ be an estimate of the random-effects covariance matrix $\boldsymbol{\Sigma}_j$ (to be discussed later), and let $\mathbf{W}_j = (\mathbf{Z}_j\widehat{\boldsymbol{\Sigma}}_j\mathbf{Z}_j' + \boldsymbol{\Lambda}_j)^{-1}$. The vector of fixed-effects regression coefficients $\boldsymbol{\beta}$ can be estimated as

$$\widehat{\boldsymbol{\beta}} = \left(\sum_{j=1}^{M} \mathbf{X}_j'\mathbf{W}_j\mathbf{X}_j\right)^{-1} \sum_{j=1}^{M} \mathbf{X}_j'\mathbf{W}_j\hat{\boldsymbol{\theta}}_j$$

The corresponding covariance matrix is given by

$$\text{Var}(\widehat{\boldsymbol{\beta}}) = \left(\sum_{j=1}^{M} \mathbf{X}_j'\mathbf{W}_j\mathbf{X}_j\right)^{-1}$$

In the following section, we outline the estimation of the random-effects covariance matrices $\boldsymbol{\Sigma}^{(2)}$ and $\boldsymbol{\Sigma}^{(3)}$ (and thus of $\boldsymbol{\Sigma}_j$) for the ML and REML methods.

Methods for estimating $\Sigma^{(2)}$ and $\Sigma^{(3)}$

The two estimators described below do not have a closed-form solution, and an iterative algorithm is needed to estimate $\boldsymbol{\Sigma}^{(2)}$ and $\boldsymbol{\Sigma}^{(3)}$.

The joint log-likelihood function of $\boldsymbol{\beta}$, $\boldsymbol{\Sigma}^{(2)}$, and $\boldsymbol{\Sigma}^{(3)}$ for a random-effects multivariate meta-regression can be expressed as

$$\ln L_{\text{ML}}\left(\boldsymbol{\beta}, \boldsymbol{\Sigma}^{(2)}, \boldsymbol{\Sigma}^{(3)}\right) = -\frac{1}{2}\left\{n\ln(2\pi) + \sum_{j=1}^{M}\ln\left|\mathbf{V}_j\right| + \sum_{j=1}^{M}\left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\boldsymbol{\beta}\right)'\mathbf{V}_j^{-1}\left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\boldsymbol{\beta}\right)\right\}$$

where $\mathbf{V}_j = \mathbf{Z}_j\boldsymbol{\Sigma}_j\mathbf{Z}_j' + \boldsymbol{\Lambda}_j$, $|\mathbf{V}_j|$ is the determinant of $\mathbf{V}_j$, and $n = \sum_{j=1}^{M}\sum_{k=1}^{m_j} m_{jk}$ is the total number of observations $\hat{\theta}_{jkr}$.

The random-effects covariance matrices $\boldsymbol{\Sigma}^{(2)}$ and $\boldsymbol{\Sigma}^{(3)}$ are estimated by maximizing the profile log-likelihood function obtained by treating $\boldsymbol{\beta}$ as known and plugging $\widehat{\boldsymbol{\beta}}$ into $\ln L_{\text{ML}}(\boldsymbol{\beta}, \boldsymbol{\Sigma})$ in place of $\boldsymbol{\beta}$ (Pinheiro and Bates [2000, chap. 2]):

$$\ln L_{\text{ML}}\left(\boldsymbol{\Sigma}^{(2)}, \boldsymbol{\Sigma}^{(3)}\right) = -\frac{1}{2}\left\{n\ln(2\pi) + \sum_{j=1}^{M}\ln\left|\mathbf{V}_j\right| + \sum_{j=1}^{M}\left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}\right)'\mathbf{V}_j^{-1}\left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}\right)\right\}$$

The MLE of $\boldsymbol{\Sigma}^{(2)}$ and $\boldsymbol{\Sigma}^{(3)}$ does not incorporate the uncertainty about the unknown regression coefficients $\boldsymbol{\beta}$ and thus can be negatively biased.

The REML estimator of $\boldsymbol{\Sigma}^{(2)}$ and $\boldsymbol{\Sigma}^{(3)}$ maximizes the restricted log-likelihood function

$$\ln L_{\text{REML}}\left(\boldsymbol{\Sigma}^{(2)}, \boldsymbol{\Sigma}^{(3)}\right) = \ln L_{\text{ML}}\left(\boldsymbol{\Sigma}^{(2)}, \boldsymbol{\Sigma}^{(3)}\right) - \frac{1}{2}\ln\left|\sum_{j=1}^{M}\mathbf{X}_j'\mathbf{V}_j^{-1}\mathbf{X}_j\right| + \frac{p}{2}\ln(2\pi)$$

The REML method estimates $\boldsymbol{\Sigma}^{(2)}$ and $\boldsymbol{\Sigma}^{(3)}$ by accounting for the uncertainty in the estimation of $\boldsymbol{\beta}$, which leads to a nearly unbiased estimate of $\boldsymbol{\Sigma}^{(2)}$ and $\boldsymbol{\Sigma}^{(3)}$. The optimization of the above log-likelihood functions can be done using the machinery of the mixed-effects models to obtain the estimates $\widehat{\boldsymbol{\beta}}$, $\boldsymbol{\Sigma}^{(2)}$, and $\boldsymbol{\Sigma}^{(3)}$. For details, see Pinheiro and Bates (2000) and *Methods and formulas* of [ME] **mixed**. When there are only two levels of hierarchy in the model and no random slopes, that is, in the context of standard meta-analysis, the above ML and REML estimators reduce to their counterparts as reported by `meta regress`.

Random-effects covariance structures

Several covariance structures may be assumed for the $q_l \times q_l$ random-effects covariance matrix $\boldsymbol{\Sigma}^{(l)}$ at a specific level of hierarchy l. The default covariance structure is `independent`, which assumes there are q_l standard deviations to be estimated corresponding to the q_l random effects at level l. Other covariance structures are `exchangeable`, `identity`, `unstructured`, and `custom` *matname*. Structures that allow the random effects to be correlated (`unstructured`, `exchangeable`, and potentially `custom` *matname*) should be used only when adequate observations are available in order to produce stable estimates of the correlations.

For example, when there are 3 random effects at level l ($q_l = 3$), the covariance structures are

$$\texttt{independent}\quad \boldsymbol{\Sigma}^{(l)} = \begin{bmatrix}\sigma_{11} & & \\ 0 & \sigma_{22} & \\ 0 & 0 & \sigma_{33}\end{bmatrix}$$

$$\texttt{exchangeable}\quad \boldsymbol{\Sigma}^{(l)} = \begin{bmatrix}\sigma_{11} & & \\ \sigma_{21} & \sigma_{11} & \\ \sigma_{21} & \sigma_{21} & \sigma_{11}\end{bmatrix}$$

$$\texttt{identity}\quad \boldsymbol{\Sigma}^{(l)} = \begin{bmatrix}\sigma_{11} & & \\ 0 & \sigma_{11} & \\ 0 & 0 & \sigma_{11}\end{bmatrix}$$

$$\texttt{unstructured}\quad \boldsymbol{\Sigma}^{(l)} = \begin{bmatrix}\sigma_{11} & & \\ \sigma_{21} & \sigma_{22} & \\ \sigma_{31} & \sigma_{32} & \sigma_{33}\end{bmatrix}$$

For the custom covariance structure, see `covariance(custom` *matname*`)`.

Multilevel meta-analysis

The formulas presented so far are derived for the general case of multilevel meta-regression. Methods and formulas for the special case of multilevel meta-analysis (when no moderators are included) can be obtained by taking $\mathbf{x}_{jkr} = 1$, $\mathbf{z}^{(3)}_{jkr} = 1$, $\mathbf{z}^{(2)}_{jkr} = 1$, and $p = 1$. This model can be expressed as

$$\hat{\theta}_{jkr} = \beta_0 + u^{(3)}_j + u^{(2)}_{jk} + \epsilon_{jkr}$$

where $j = 1, 2, \ldots, M$, $k = 1, 2, \ldots, m_j$, and $r = 1, 2, \ldots, m_{jk}$. When there are only two levels of hierarchy in the model, the REML and ML estimators reduce to the classical REML and ML estimators described in [META] **meta summarize** for constant-only models.

Residual homogeneity test

Consider a test of residual homogeneity, which mathematically translates to H_0: $\boldsymbol{\Sigma}^{(l)} = \mathbf{0}_{q_l \times q_l}$, $l = 2, 3$, for the multilevel meta-regression. This test is based on the multivariate residual weighted sum of squares, Q_{M}, defined as

$$Q_{\mathrm{M}} = \sum_{j=1}^{M} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j \widehat{\boldsymbol{\beta}}_f\right)' \boldsymbol{\Lambda}_j^{-1} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j \widehat{\boldsymbol{\beta}}_f\right)$$

where $\widehat{\boldsymbol{\beta}}_f$ is a fixed-effects estimator obtained by fitting a standard fixed-effects meta-regression (see [META] **meta regress**) of the $\hat{\theta}_{jkr}$'s on the moderators defining the $\mathbf{X}_j$ matrix.

Under the null hypothesis of residual homogeneity, Q_{M} follows a χ^2 distribution with $n - p$ degrees of freedom (Seber and Lee 2003, sec. 2.4). The Q_{M} statistic reduces to the classical residual homogeneity test statistic, Q_{res}, when there are two levels of hierarchy and no random slopes in the model (see *Residual homogeneity test* in *Methods and formulas* in [META] **meta regress**). It also reduces to the classical homogeneity statistic Q when no moderators are included (see *Homogeneity test* in *Methods and formulas* in [META] **meta summarize**).

References

Cheung, M. W.-L. 2014. Modeling dependent effect sizes with three-level meta-analyses: A structural equation modeling approach. *Psychological Methods* 19: 211–229. https://doi.org/10.1037/a0032968.

Cooper, H., J. C. Valentine, and A. Melson. 2003. The effects of modified school calendars on student achievement and on school and community attitudes. *Review of Educational Research* 73: 1–52. https://doi.org/10.3102/00346543073001001.

Goldstein, H., M. Yang, R. Z. Omar, R. M. Turner, and S. G. Thompson. 2000. Meta-analysis using multilevel models with an application to the study of class size effects. *Journal of the Royal Statistical Society*, C ser., 49: 399–412. https://doi.org/10.1111/1467-9876.00200.

Gutierrez, R. G., S. L. Carter, and D. M. Drukker. 2001. sg160: On boundary-value likelihood-ratio tests. *Stata Technical Bulletin* 60: 15–18. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 269–273. College Station, TX: Stata Press.

Konstantopoulos, S. 2011. Fixed effects and variance components estimation in three-level meta-analysis. *Research Synthesis Methods* 2: 61–76. https://doi.org/10.1002/jrsm.35.

McCurdy, M. P., W. Viechtbauer, A. M. Sklenar, A. N. Frankenstein, and E. D. Leshikar. 2020. Theories of the generation effect and the impact of generation constraint: A meta-analytic review. *Psychonomic Bulletin and Review* 27: 1139–1165. https://doi.org/10.3758/s13423-020-01762-3.

Nakagawa, S., and E. S. A. Santos. 2012. Methodological issues and advances in biological meta-analysis. *Evolutionary Ecology* 26: 1253–1274. https://doi.org/10.1007/s10682-012-9555-5.

Orsini, N. 2021. Weighted mixed-effects dose–response models for tables of correlated contrasts. *Stata Journal* 21: 320–347.

Pinheiro, J. C., and D. M. Bates. 2000. *Mixed-Effects Models in S and S-PLUS*. New York: Springer. https://doi.org/10.1007/b98882.

Raudenbush, S. W. 1984. Magnitude of teacher expectancy effects on pupil IQ as a function of the credibility of expectancy induction: A synthesis of findings from 18 experiments. *Journal of Educational Psychology* 76: 85–97. https://doi.org/10.1037/0022-0663.76.1.85.

Raudenbush, S. W., and A. S. Bryk. 1985. Empirical Bayes meta-analysis. *Journal of Educational Statistics* 10: 75–98. https://doi.org/10.2307/1164836.

———. 2002. *Hierarchical Linear Models: Applications and Data Analysis Methods*. 2nd ed. Thousand Oaks, CA: Sage.

Rose, C. J. 2024. Multivariate random-effects meta-analysis for sparse data using smvmeta. *Stata Journal* 24: 301–328.

Seber, G. A. F., and A. J. Lee. 2003. *Linear Regression Analysis*. 2nd ed. Hoboken, NJ: Wiley. https://doi.org/10.1002/9780471722199.

Self, S. G., and K.-Y. Liang. 1987. Asymptotic properties of maximum likelihood estimators and likelihood ratio tests under nonstandard conditions. *Journal of the American Statistical Association* 82: 605–610. https://doi.org/10.2307/2289471.

Sera, F., B. Armstrong, M. Blangiardo, and A. Gasparrini. 2019. An extended mixed-effects framework for meta-analysis. *Statistics in Medicine* 38: 5429–5444. https://doi.org/10.1002/sim.8362.

Thompson, S. G., R. M. Turner, and D. E. Warn. 2001. Multilevel models for meta-analysis, and their application to absolute risk differences. *Statistical Methods in Medical Research* 10: 375–392. https://doi.org/10.1177/096228020101000602.

Verbeke, G., and G. Molenberghs. 2000. *Linear Mixed Models for Longitudinal Data*. New York: Springer. https://doi.org/10.1007/978-1-4419-0300-6.

Also see

[META] **meta me postestimation** — Postestimation tools for multilevel mixed-effects meta-analysis

[META] **meta regress** — Meta-analysis regression

[META] **meta summarize** — Summarize meta-analysis data+

[META] **meta multilevel** — Multilevel random-intercepts meta-regression

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

[U] 20 Estimation and postestimation commands

meta multilevel — Multilevel random-intercepts meta-regression

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
References | Also see

Description

`meta multilevel` fits random-intercepts meta-analysis models, which are commonly used in practice. For fitting more complicated multilevel meta-analysis models, including random slopes, see [META] **meta meregress**. `meta multilevel` is a convenience wrapper for `meta meregress`.

`meta multilevel` is a stand-alone command in that it does not require you to declare your data as `meta` data using `meta set` or `meta esize`.

Quick start

Perform standard RE meta-analysis by expressing it as a two-level meta-analysis model of the effect size `y` with random intercepts by `trial` and effect-size standard errors (`se`)

```
meta multilevel y, relevels(trial) essevariable(se)
```

As above, but perform a RE meta-regression on continuous moderator `x`

```
meta multilevel y x, relevels(trial) essevariable(se)
```

As above, but specify effect-size variances (`var`) instead of the effect-size standard errors

```
meta multilevel y x, relevels(trial) esvarvariable(var)
```

Perform a three-level meta-analysis of `y` with random intercepts by `region` and by `trial` nested within `region`, and request the ML instead of the default REML estimation method

```
meta multilevel y, relevels(region trial) essevariable(se) mle
```

Perform a three-level meta-regression of `y` on `x1` and `x2` and specify a fixed standard deviation for the `trial`-within-`region` random intercepts

```
meta multilevel y x1 x2, relevels(region trial, sd(. .2)) ///
  essevariable(se)
```

Menu

Statistics > Meta-analysis

Syntax

```
meta multilevel depvar [indepvars] [if] [in], relevels(relevspec)
  { essevariable(varname) | esvarvariable(varname) } [options]
```

options	Description
Model	
noconstant	suppress constant term from the fixed-effects equation
* relevels(*relevspec*)	specify the grouping structure of the model
† essevariable(*varname*)	specify effect-size (sampling) standard errors
† esvarvariable(*varname*)	specify effect-size (sampling) variances
reml	fit model via restricted maximum likelihood; the default
mle	fit model via maximum likelihood
constraints(*constraints*)	apply specified linear constraints
Reporting	
level(#)	set confidence level; default is level(95)
stddeviations	show random-effects parameter estimates as standard deviations and correlations; the default
variance	show random-effects parameter estimates as variances and covariances
estmetric	show parameter estimates as stored in e(b)
nohomtest	suppress output for homogeneity test
noretable	suppress random-effects table
nofetable	suppress fixed-effects table
noheader	suppress output header
nogroup	suppress table summarizing groups
nocnsreport	do not display constraints
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
EM options	
emiterate(#)	number of EM iterations; default is emiterate(20)
emtolerance(#)	EM convergence tolerance; default is emtolerance(1e-10)
emonly	fit model exclusively using EM
emlog	show EM iteration log
emdots	show EM iterations as dots
Maximization	
maximize_options	control the maximization process; seldom used
collinear	keep collinear variables
coeflegend	display legend instead of statistics

*relevels() is required. The full specification is relevels(*varlist*[, sd(# [# [...]])]).

†Either essevariable() or esvarvariable() is required.

indepvars may contain factor variables; see **[U] 11.4.3 Factor variables**.

collect is allowed; see **[U] 11.1.10 Prefix commands**.

collinear and coeflegend do not appear in the dialog box.

See **[U] 20 Estimation and postestimation commands** for more capabilities of estimation commands.

Options

Model

noconstant suppresses the constant (intercept) term from the fixed-effects model.

relevels(*varlist*[, sd(# [# [...]])]) specifies the grouping structure of the multilevel model. A random intercept corresponding to each level variable in *varlist* is included in the model. The order of *varlist* is important. The first variable is assumed to be the highest grouping level, and each subsequent variable is assumed to be nested within the previous one. For example, relevels(region study) assumes that variable region is the highest grouping level and that study is nested within region. relevels() is required.

sd(# [# [...]]) specifies fixed values for the standard deviations of the random intercepts during estimation. The order of the values # [# [...]] should correspond to the order of variables in relevels(). A missing value (.) means that the standard deviation of the corresponding random intercept is to be estimated. This suboption is useful for exploring the sensitivity of the results to different magnitudes of random-intercepts standard deviations.

essevariable(*varname*) specifies a variable that stores the standard errors of the effect sizes in variable *varname*, also known as sampling standard errors. You must specify one of essevariable() or esvarvariable().

esvarvariable(*varname*) specifies a variable that stores the variances of the effect sizes in variable *varname*, also known as sampling variances. You must specify one of esvarvariable() or essevariable().

reml and mle specify the statistical method for fitting the model.

reml, the default, specifies that the model be fit using restricted maximum likelihood (REML), also known as residual maximum likelihood.

mle specifies that the model be fit using maximum likelihood (ML).

constraints(*constraints*); see [R] **Estimation options**.

Reporting

level(#); see [R] **Estimation options**.

stddeviations, the default, displays the random-effects parameter estimates as standard deviations and correlations.

variance displays the random-effects parameter estimates as variances and covariances.

estmetric; see [ME] **mixed**.

nohomtest suppresses the homogeneity test based on the Q_M statistic from the output.

noretable, nofetable, noheader, and nogroup; see [ME] **mixed**.

nocnsreport; see [R] **Estimation options**.

display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(#), fvwrapon(*style*), cformat(%*fmt*), pformat(%*fmt*), sformat(%*fmt*), and nolstretch; see [R] **Estimation options**.

EM options

emiterate(#), emtolerance(#), emonly, emlog, and emdots; see [ME] **mixed**.

Maximization

maximize_options: difficult, technique(*algorithm_spec*), iterate(#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(#), ltolerance(#), nrtolerance(#), and nonrtolerance; see [R] **Maximize**. Those that require special mention for meta multilevel are listed below.

For the technique() option, the default is technique(nr). The bhhh algorithm is not available.

matsqrt, the default, and matlog; see [ME] **mixed**.

The following options are available with meta multilevel but are not shown in the dialog box:

collinear, coeflegend; see [R] **Estimation options**.

Remarks and examples

Remarks are presented under the following headings:

Introduction
Examples of using meta multilevel

Introduction

For an introduction to the general multilevel meta-regression model, see *Introduction* in [META] **meta meregress**.

Let $\mathbf{x}_{jkr} = (1, x_{1,jkr}, \ldots, x_{p-1,jkr})$ be a $1 \times p$ vector of moderators and $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_{p-1})'$ be the corresponding $p \times 1$ vector of unknown fixed-effects regression coefficients. The three-level random-intercepts meta-regression model (Goldstein et al. [2000]; Thompson, Turner, and Warn [2001]; and Konstantopoulos [2011]) can be expressed as

$$\begin{aligned}\widehat{\theta}_{jkr} &= \beta_0 + \beta_1 x_{1,jkr} + \cdots + \beta_{p-1} x_{p-1,jkr} + u^{(3)}_j + u^{(2)}_{jk} + \epsilon_{jk} \\ &= \mathbf{x}_{jkr}\boldsymbol{\beta} + u^{(3)}_j + u^{(2)}_{jk} + \epsilon_{jkr}\end{aligned} \tag{1}$$

where $j = 1, 2, \ldots, M$, $k = 1, 2, \ldots, m_j$, and $r = 1, 2, \ldots, m_{jk}$. $u^{(3)}_j \sim N(0, \tau^2_3)$, $u^{(2)}_{jk} \sim N(0, \tau^2_2)$, and $\epsilon_{jkr} \sim N(0, \widehat{\sigma}^2_{jkr})$, with the $\widehat{\sigma}^2_{jkr}$'s being the known sampling variances (variances of the effect sizes). The random intercepts (the $u^{(3)}_j$'s, $u^{(2)}_{jk}$'s) and the sampling errors (the ϵ_{jk}'s) are independent. τ^2_3 and τ^2_2 are the random-intercepts variances at the third and second levels, respectively. Model (1) and its higher-level extensions are precisely the models that meta multilevel was designed to fit. If you wish to fit models that incorporate random slopes, see the more general command [META] **meta meregress**.

`meta multilevel` fits multilevel random-intercepts meta-regression. By default, the REML method is used to estimate the random-intercepts variances τ^2_3 and τ^2_2. Use the `mle` option to request ML estimation. REML is typically preferred over ML because it produces unbiased estimates of the random-effects variance parameters by accounting for the loss of degrees of freedom from estimating the fixed-effects vector $\boldsymbol{\beta}$.

The `relevels()` option specifies the variables that identify the different levels of hierarchy that are present in the model. For each level of hierarchy, a random intercept is added to the model. The order of the specified variables is important. The first variable is assumed to be the highest grouping level, and each subsequent variable is assumed to be nested within the previous one.

The `sd()` suboption within `relevels()` provides a flexible way to restrict specific random-intercepts standard deviations during estimation while allowing the remaining parameters to be freely estimated. This option can be seen as a generalization of option `tau2()` in [META] **meta regress** and thus can be used to perform sensitivity analysis; see suboption `sd()` in *Options*.

The sampling variances (the $\hat{\sigma}^2_{jkr}$'s) are treated as known and do not require estimation. The variable that stores these values is specified in the `esvarvariable()` option. Alternatively, if the sampling standard errors (the $\hat{\sigma}_{jkr}$'s) are available, then option `essevariable()` can be used instead.

For example, suppose we specify the following in Stata:

```
. meta multilevel y x1 x2, relevels(lev3var lev2var) esvarvariable(var)
```

Consider how the above specification relates to the components of (1). Variable `y` stores the values of the $\hat{\theta}_{jkr}$'s, and the variables `x1` and `x2` represent the fixed-effects component of the model, $\mathbf{x}_{jkr}\boldsymbol{\beta}$. Three fixed-effects parameters will need to be estimated: an intercept and two coefficients corresponding to variables `x1` and `x2`, respectively. The `relevels(lev3var lev2var)` option specifies that two random intercepts are to be included in the model: one at level 3 (identified by variable `lev3var`) and another one at level 2 (identified by variable `lev2var`). These are the $u^{(3)}_j$ and $u^{(2)}_{jk}$ terms in (1). Level 1 corresponds to the participant or subject-level data, which are not available in meta-analysis. In general, if you specify L variables within `relevels()`, then $L + 1$ levels of hierarchy will be present in the model, with the leftmost variable corresponding to the highest level. The `esvarvariable(var)` option specifies the variable name (`var` in our example) that stores the sampling variances (the $\hat{\sigma}^2_{jkr}$'s) of the ϵ_{jkr}'s.

You may also use suboption `sd()` within `relevels()` to fix certain random-intercepts standard deviations at specified values during estimation, while allowing the remaining standard deviations to be freely estimated as follows:

```
. meta multilevel y x1 x2, relevels(lev3var lev2var, sd(.4 .)) esvarvariable(var)
```

Option `sd(.4 .)` specifies that the standard deviation of $u^{(3)}_j$ is to be fixed at .4 during estimation and that the standard deviation of $u^{(2)}_{jk}$ is to be estimated.

Examples of using meta multilevel

Examples are presented under the following headings:

Example 1: Three-level meta-analysis
Example 2: Sensitivity multilevel meta-analysis

Example 1: Three-level meta-analysis

Continuing with example 2 of [META] **meta meregress**, which explored the effect of a modified school calendar on student achievement (Cooper, Valentine, and Melson 2003), we will fit the same model using the syntax of `meta multilevel`. Recall that this command fits models that contain random intercepts only (no random slopes). Our three-level random-intercepts model is given by

$$\texttt{stdmdiff}_{jk} = \theta + u_j^{(3)} + u_{jk}^{(2)} + \epsilon_{jk} \tag{2}$$

with $u_j^{(3)} \sim N(0, \tau_3^2)$, $u_{jk}^{(2)} \sim N(0, \tau_2^2)$, and $\epsilon_{jk} \sim N(0, \texttt{se}_{jk}^2)$. Here there is one observation (effect size) reported per school (level-2 group); therefore, $m_{jk} = 1$ in formula (1) in *Introduction*. This model can be fit using `meta multilevel` as follows:

```
. use https://www.stata-press.com/data/r18/schoolcal
(Effect of modified school calendar on student achievement)
. meta multilevel stdmdiff, relevels(district school) essevariable(se)
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood =  -104.8525  (not concave)
Iteration 1:  Log restricted-likelihood = -49.423271  (not concave)
Iteration 2:  Log restricted-likelihood = -25.793841  (not concave)
Iteration 3:  Log restricted-likelihood = -21.310018
Iteration 4:  Log restricted-likelihood = -9.1236345
Iteration 5:  Log restricted-likelihood = -8.2625776
Iteration 6:  Log restricted-likelihood = -7.9588561
Iteration 7:  Log restricted-likelihood = -7.9587239
Iteration 8:  Log restricted-likelihood = -7.9587239
Computing standard errors ...
Multilevel REML meta-analysis                          Number of obs = 56
        Grouping information
```

Group variable	No. of groups	Observations per group Minimum	Average	Maximum
district	11	3	5.1	11
school	56	1	1.0	1

```
                                                       Wald chi2(0)  =  .
Log restricted-likelihood = -7.9587239                 Prob > chi2   =  .
```

stdmdiff	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
_cons	.1847132	.0845559	2.18	0.029	.0189866	.3504397

```
Test of homogeneity: Q_M = chi2(55) = 578.86            Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
district: Identity	
sd(_cons)	.2550724
school: Identity	
sd(_cons)	.1809324

By typing `stdmdiff` after `meta multilevel`, we specified the response variable (`stdmdiff`) and the fixed-effects portion of our model, which consists of a constant term (fixed-effect intercept), denoted by θ in (2). We could have specified `stdmdiff` *indepvars* to include additional moderators (independent variables) in the same way that we would if we were using any other estimation command. The `relevels(district school)` option defines two levels of hierarchy (the model will then have three levels, given that level 1 always corresponds to effect sizes) and includes random intercepts at both levels [the $u_j^{(3)}$ and $u_{jk}^{(2)}$ terms in (2)]. The order in which the variables are specified within `relevels()` (from left to right) is important—`meta multilevel` assumes that `school` is nested within `district`. This model was specified as follows in example 2 of [META] **meta meregress** (see that example for output interpretation):

```
. meta meregress stdmdiff || district: || school:, essevariable(se)
```

In other words, the `relevels(district school)` specification in `meta multilevel` is equivalent to the `|| district: || school:` specification in `meta meregress`.

◁

▷ Example 2: Sensitivity multilevel meta-analysis

We may often wish to fix certain random-intercepts standard deviations at specified values during estimation, while allowing the remaining standard deviations to be freely estimated. This could be done as a form of sensitivity analysis to assess the impact of certain random-effects parameters on estimation overall. The `sd()` suboption within `relevels()` can be a useful tool for this. For example, we may fix the value of τ_3^2 at the value reported in example 1 (0.2550724) and check that the estimates of the other parameters match with what was reported in that example. We use options `nolog` and `noheader` to suppress the log and header output for a more compact display of the results.

```
. meta multilevel stdmdiff, relevels(district school, sd(.2550724 .))
> essevariable(se) nolog noheader
```

stdmdiff	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
_cons	.1847132	.0845559	2.18	0.029	.0189866	.3504397

Test of homogeneity: Q_M = chi2(55) = 578.86 Prob > Q_M = 0.0000

Random-effects parameters	Estimate
district: Custom	
sd(_cons)	.2550724*
school: Identity	
sd(_cons)	.1809324

(*) fixed during estimation

The order in which you specify values in `sd()` corresponds to the order in which the variables were specified within `relevels()`. In other words, the first value corresponds to the standard deviation of the random effects at the `district` level and the second value to that at the `school` level. The second `.` in `sd(.2550724 .)` means that the standard deviation of the random intercepts at the `school` level, τ_2, is free and needs to be estimated. The two outputs are essentially identical, as expected. Notice the starred note to indicate which parameter was fixed during estimation.

Next we will assess the impact of five different magnitudes in increasing order of the value of the random-effects standard deviations at the `school` level on the estimation of the other model parameters (θ and τ_3). We fit five models corresponding to fixing τ_2 at each element of matrix `val` in a loop and store their results under the names `fixsd1`, `fixsd2`, and so on.

```
. matrix val = (.01, .08, .18, .3, .6)
. forvalues i=1/5 {
  2.          quietly meta multilevel stdmdiff,
>              relevels(district school, sd(. `=val[1,`i']')) essevariable(se)
  3.          estimates store fixsd`i'
  4. }
```

We then use `estimates table` to report $\widehat{\theta}$ (option `keep(stdmdiff:_cons)`) and its standard error from the five models for ease of comparison.

```
. estimates table _all, stats(sd2) keep(stdmdiff:_cons) b(%8.3f) se(%8.3f)
```

Variable	fixsd1	fixsd2	fixsd3	fixsd4	fixsd5
_cons	0.196	0.193	0.185	0.172	0.123
	0.090	0.088	0.085	0.081	0.083
sd2	0.010	0.080	0.180	0.300	0.600

Legend: b/se

As τ_2 (`sd2` in the output) increases from 0.01 to 0.6, $\widehat{\theta}$ (`_cons` in the output) decreases from 0.196 to 0.123 and seems to be estimated with more precision (its standard error decreases). This suggests that increased variability among schools leads to a smaller overall standardized mean difference, resulting in less benefit from the modified-calendar program. Recall that a positive mean difference corresponds to higher student achievement in the group on the modified calendar.

The next table shows the estimates of $\tau_3 = \sqrt{\mathrm{Var}(u_j^{(3)})}$ for the different fixed values of τ_2. The term `lns1_1_1:_cons` (used within option `keep()`) stores the value of $\log(\tau_3)$, so we use the `eform` option to report the exponentiated value.

```
. estimates table _all, stats(sd2) keep(lns1_1_1:_cons) b(%8.3f) eform
```

Variable	fixsd1	fixsd2	fixsd3	fixsd4	fixsd5
_cons	0.288	0.278	0.255	0.215	0.000
sd2	0.010	0.080	0.180	0.300	0.600

As τ_2 (`sd2`) increases from 0.01 to 0.6, $\widehat{\tau_3}$ (`_cons`) decreases from 0.288 to nearly 0, indicating that as τ_2 increases, it will eventually capture all the variability (excluding sampling error) among the effect sizes. In this case, the district level (level 3) may be dropped from the model.

◁

Stored results

meta multilevel stores the following in e():

Scalars

e(N)	total number of observations
e(k)	number of parameters
e(k_f)	number of fixed-effects parameters
e(k_r)	number of random-effects parameters
e(k_rs)	number of variances
e(ll)	log (restricted) likelihood
e(rank)	rank of e(V)
e(ic)	number of iterations
e(sd#)	user-specified random-intercepts standard deviation (when suboption sd() of relevels() is specified)
e(df_m)	model degrees of freedom
e(chi2)	model χ^2 Wald test statistic
e(p)	p-value for model test
e(Q_M)	multilevel Cochran Q_M residual homogeneity test statistic
e(df_Q_M)	degrees of freedom for residual homogeneity test
e(p_Q_M)	p-value for residual homogeneity test
e(converged)	1 if converged, 0 otherwise

Macros

e(cmd)	meta multilevel
e(cmdline)	command as typed
e(method)	REML or ML
e(title)	title in estimation output
e(chi2type)	Wald; type of model χ^2 test
e(depvar)	name of dependent variable
e(ivars)	grouping variables
e(indepvars)	names of independent variables (moderators)
e(esvarvariable)	variable containing sampling variances (when esvarvariable() is specified)
e(essevariable)	variable containing sampling standard errors (when essevariable() is specified)
e(technique)	maximization technique
e(datasignature)	the checksum
e(datasignaturevars)	variables used in calculation of checksum
e(emonly)	emonly, if specified
e(ml_method)	type of ml method
e(opt)	type of optimization
e(optmetric)	matsqrt or matlog; random-effects matrix parameterization
e(properties)	b V
e(predict)	program used to implement predict
e(estat_cmd)	program used to implement estat
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices

e(b)	coefficient vector
e(V)	variance–covariance matrix of the estimators
e(Cns)	constraints matrix
e(N_g)	group counts
e(g_min)	group-size minimums
e(g_avg)	group-size averages
e(g_max)	group-size maximums

Functions

e(sample)	marks estimation sample

In addition to the above, the following is stored in `r()`:

Matrices

`r(table)` matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in `r()` are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

When the `esvarvariable()` option is specified, `meta multilevel` creates a system variable, `_meta_mereg_se`, that contains the sampling standard errors.

Methods and formulas

Let $\mathbf{X}_j$, $\hat{\boldsymbol{\theta}}_j$, and $\boldsymbol{\epsilon}_j$ be defined as in *Methods and formulas* of [META] **meta meregress**. If we eliminate the explicit reference to specific levels of hierarchy, then (1) can be expressed compactly as

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \dot{\mathbf{Z}}_j\dot{\mathbf{u}}_j + \boldsymbol{\epsilon}_j$$

where $m_{j.} \times (m_j + 1)$ matrix $\dot{\mathbf{Z}}_j = (\mathbf{1}_{m_{j.}}, \oplus_{k=1}^{m_j}\mathbf{1}_{m_{jk}})$ and $(m_j + 1) \times 1$ vector $\dot{\mathbf{u}}_j = (u_j^{(3)}, u_{j1}^{(2)}, u_{j2}^{(2)}, \ldots, u_{jm_j}^{(2)})'$, with a $(m_j + 1) \times (m_j + 1)$ covariance matrix $\dot{\boldsymbol{\Sigma}}_j$, defined as

$$\dot{\boldsymbol{\Sigma}}_j = \mathrm{Var}\left(\dot{\mathbf{u}}_j\right) = \begin{bmatrix} \tau_3^2 & \mathbf{0} \\ \mathbf{0} & \tau_2^2\mathbf{I}_{m_j} \end{bmatrix}$$

The formulas used by `meta multilevel` to estimate $\boldsymbol{\beta}$, τ_3^2, and τ_2^2 are described in *Methods and formulas* of [META] **meta meregress** with $\boldsymbol{\Sigma}_j = \dot{\boldsymbol{\Sigma}}_j$, $\mathbf{Z}_j = \dot{\mathbf{Z}}_j$, and $\mathbf{u}_j = \dot{\mathbf{u}}_j$.

References

Cooper, H., J. C. Valentine, and A. Melson. 2003. The effects of modified school calendars on student achievement and on school and community attitudes. *Review of Educational Research* 73: 1–52. https://doi.org/10.3102/00346543073001001.

Goldstein, H., M. Yang, R. Z. Omar, R. M. Turner, and S. G. Thompson. 2000. Meta-analysis using multilevel models with an application to the study of class size effects. *Journal of the Royal Statistical Society*, C ser., 49: 399–412. https://doi.org/10.1111/1467-9876.00200.

Konstantopoulos, S. 2011. Fixed effects and variance components estimation in three-level meta-analysis. *Research Synthesis Methods* 2: 61–76. https://doi.org/10.1002/jrsm.35.

Thompson, S. G., R. M. Turner, and D. E. Warn. 2001. Multilevel models for meta-analysis, and their application to absolute risk differences. *Statistical Methods in Medical Research* 10: 375–392. https://doi.org/10.1177/096228020101000602.

Also see

[META] **meta me postestimation** — Postestimation tools for multilevel mixed-effects meta-analysis

[META] **meta meregress** — Multilevel mixed-effects meta-regression

[META] **meta regress** — Meta-analysis regression

[META] **meta summarize** — Summarize meta-analysis data+

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

[U] 20 Estimation and postestimation commands

Postestimation commands | predict | margins
Remarks and examples | Methods and formulas | References
Also see

Postestimation commands

`meta meregress` and `meta multilevel` allow the same postestimation commands. The following postestimation commands are of special interest after `meta meregress` and `meta multilevel`:

Command	Description
`estat group`	summarize the composition of the nested groups
`estat heterogeneity`	compute multilevel heterogeneity statistics
`estat recovariance`	display the estimated random-effects covariance matrices
`estat sd`	display variance components as standard deviations and correlations

The following standard postestimation commands are also available:

Command	Description
`contrast`	contrasts and ANOVA-style joint tests of parameters
`estat ic`	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC, respectively)
`estat summarize`	summary statistics for the estimation sample
`estat vce`	variance–covariance matrix of the estimators (VCE)
`estimates`	cataloging estimation results
`etable`	table of estimation results
`lincom`	point estimates, standard errors, testing, and inference for linear combinations of parameters
`lrtest`	likelihood-ratio test
`margins`	marginal means, predictive margins, marginal effects, and average marginal effects
`marginsplot`	graph the results from margins (profile plots, interaction plots, etc.)
`nlcom`	point estimates, standard errors, testing, and inference for nonlinear combinations of parameters
`predict`	predictions and their SEs, leverage statistics, etc.
`predictnl`	point estimates, standard errors, testing, and inference for generalized predictions
`pwcompare`	pairwise comparisons of parameters
`test`	Wald tests of simple and composite linear hypotheses
`testnl`	Wald tests of nonlinear hypotheses

predict

Description for predict

`predict` creates a new variable containing predictions such as linear predictions, standard errors, fitted values, residuals, and standardized residuals. You can also obtain predictions of random effects and estimates of their standard errors.

Menu for predict

Statistics > Postestimation

Syntax for predict

Syntax for obtaining predictions other than best linear unbiased predictions (BLUPs) of random effects

`predict` [*type*] *newvar* [*if*] [*in*] [, *statistic* `relevel(`*levelvar*`)`]

Syntax for obtaining BLUPs of random effects and the BLUPs' standard errors

`predict` [*type*] { *stub*`*` | *newvarlist* } [*if*] [*in*] , `reffects` [`relevel(`*levelvar*`)`
`reses(`*resesspec*`)`]

statistic	Description
Main	
`xb`	linear prediction for the fixed portion of the model only; the default
`stdp`	standard error of the fixed-portion linear prediction
`fitted`	fitted values, fixed-portion linear prediction plus contributions based on predicted random effects
`residuals`	residuals, response minus fitted values
* `rstandard`	standardized residuals

Unstarred statistics are available both in and out of sample; type `predict ... if e(sample) ...` if wanted only for the estimation sample. Starred statistics are calculated only for the estimation sample, even when `if e(sample)` is not specified.

Options for predict

Main

`xb`, the default, calculates the linear prediction $\mathbf{X}_j\widehat{\boldsymbol{\beta}}$ based on the estimated fixed effects (coefficients) in the model. This is equivalent to fixing all random effects in the model to their theoretical mean value of 0.

`stdp` calculates the standard error of the linear prediction $\mathbf{X}_j\widehat{\boldsymbol{\beta}}$.

`fitted` calculates fitted values, which are equal to the fixed-portion linear predictor *plus* contributions based on predicted random effects, $\mathbf{X}_j\widehat{\boldsymbol{\beta}} + \mathbf{Z}_j\hat{\mathbf{u}}_j$. By default, the fitted values account for random effects from all levels in the model; however, if the `relevel(`*levelvar*`)` option is specified, then the fitted values are fit beginning with the topmost level down to and including level *levelvar*. For example, if `trials` are nested within `regions`, then typing

```
. predict yhat_region, fitted relevel(region)
```

would produce region-level predictions. That is, the predictions would incorporate region-specific random effects but not those for each trial nested within each region.

`residuals` calculates residuals, equal to the responses minus fitted values, $\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}} - \mathbf{Z}_j\hat{\mathbf{u}}_j$. By default, the fitted values account for random effects from all levels in the model; however, if the `relevel(`*levelvar*`)` option is specified, then the fitted values are fit beginning at the topmost level down to and including level *levelvar*.

`rstandard` calculates standardized residuals, equal to the residuals multiplied by the inverse square root of the estimated error covariance matrix.

`reffects` calculates best linear unbiased predictions (BLUPs) of the random effects. By default, BLUPs for all random effects in the model are calculated. However, if the `relevel(`*levelvar*`)` option is specified, then BLUPs for only level *levelvar* in the model are calculated. For example, if `trials` are nested within `regions`, then typing

```
. predict b*, reffects relevel(region)
```

would produce BLUPs at the region level. You must specify q new variables, where q is the number of random-effects terms in the model (or level). However, it is much easier to just specify *stub*`*` and let Stata name the variables *stub*`1`, *stub*`2`, …, *stub*q for you.

`relevel(`*levelvar*`)` specifies the level in the model at which predictions involving random effects are to be obtained; see the options above for the specifics. *levelvar* is the name of the variable describing the grouping at that level. This option is not allowed with statistic `xb` or `stdp`.

`reses(`*resesspec*`)` calculates the standard errors of the random effects, where *resesspec* is

stub`*` | *newvarlist* [, `comparative` | `diagnostic`]

`comparative`, the default, computes comparative random-effects standard errors. For linear multilevel models, these correspond to posterior standard deviations of random effects and to standard errors of marginal prediction errors $\hat{\mathbf{u}}_j - \mathbf{u}_j$. These standard errors are used for inference about the random effects.

`diagnostic` computes diagnostic random-effects standard errors. These correspond to marginal standard errors of BLUPs, $\mathrm{SE}(\hat{\mathbf{u}}_j)$. These standard errors are used for model diagnostics.

By default, standard errors for all BLUPs in the model are calculated. However, if the `relevel(`*levelvar*`)` option is specified, then standard errors for only level *levelvar* in the model are calculated; see the `reffects` option.

You must specify q new variables, where q is the number of random-effects terms in the model (or level). However, it is much easier to just specify *stub*`*` and let Stata name the variables *stub*`1`, *stub*`2`, …, *stub*q for you. The new variables will have the same storage type as the corresponding random-effects variables.

The `reffects` and `reses()` options often generate multiple new variables at once. When this occurs, the random effects (or standard errors) contained in the generated variables correspond to the order in which the variance components are listed in the output of `meta meregress` or `meta multilevel`. Still, examining the variable labels of the generated variables (with the `describe` command, for instance) can be useful in deciphering which variables correspond to which terms in the model.

margins

Description for margins

`margins` estimates margins of response for linear predictions.

Menu for margins

Statistics > Postestimation

Syntax for margins

`margins` [*marginlist*] [, *options*]

`margins` [*marginlist*] , `predict(`*statistic* ...`)` [*options*]

statistic	Description
`xb`	linear predictor for the fixed portion of the model only; the default
`stdp`	not allowed with `margins`
`fitted`	not allowed with `margins`
`residuals`	not allowed with `margins`
`rstandard`	not allowed with `margins`
`reffects`	not allowed with `margins`

Statistics not allowed with `margins` are functions of stochastic quantities other than `e(b)`.

For the full syntax, see [R] **margins**.

Remarks and examples

Various predictions and statistics are available after fitting a multilevel meta-regression using `meta multilevel` or `meta meregress`. Below, we will discuss how to obtain BLUPs of the random effects. Random effects at different levels of hierarchy are not estimated when the model is fit but rather must be predicted after the estimation of the model parameters. The estimates of the random effects are in turn used to obtain other statistics such as the fitted values and residuals. These are useful for checking model assumptions and may be used in general as model-building tools.

▷ Example 1: Obtaining predictions of random effects

In example 2 of [META] **meta meregress**, we conducted a multilevel meta-analysis to assess the effect of modifying the school calendar on students' achievement test scores. For completeness, we refit that model here:

```
. use https://www.stata-press.com/data/r18/schoolcal
(Effect of modified school calendar on student achievement)
. meta meregress stdmdiff || district: || school:, essevariable(se)
  (output omitted)
```

The above model can also be fit by using the `meta multilevel` command as follows:

```
. meta multilevel stdmdiff, relevels(district school) essevariable(se)
```

We can use `estat group` to see how the data are broken down by district and school:

```
. estat group
```

Group variable	No. of groups	Observations per group Minimum	Average	Maximum
district	11	3	5.1	11
school	56	1	1.0	1

We are reminded that each district had somewhere between 3 to 11 schools and that each school reported one effect size in our dataset.

Below, we predict the random effects using `predict, reffects` and obtain their diagnostic standard errors by specifying the `reses(, diagnostic)` option. Because we have two random effects in our model (at levels 2 and 3), we need to specify two new variable names with `predict` and two new variable names within `reses()`. Although it is typically much easier to specify a stubname, say, `u*`, where `predict` constructs variables `u1` and `u2` for you, here we will explicitly specify the names `u3` and `u2` for the variables corresponding to the random intercepts at level 3 and level 2, respectively. We will also specify `se_u3` and `se_u2` within the `reses()` option instead of a stubname `u_se*`. And we will use suboption `diagnostic` of the `reses()` option to request the diagnostic standard errors instead of the default comparative standard errors. The diagnostic standard errors are used for model diagnostics (Goldstein 2011; Skrondal and Rabe-Hesketh 2009).

```
. predict double u3 u2, reffects reses(se_u3 se_u2, diagnostic)
. by district, sort: generate tolist = (_n==1)
. list district u3 se_u3 if tolist
```

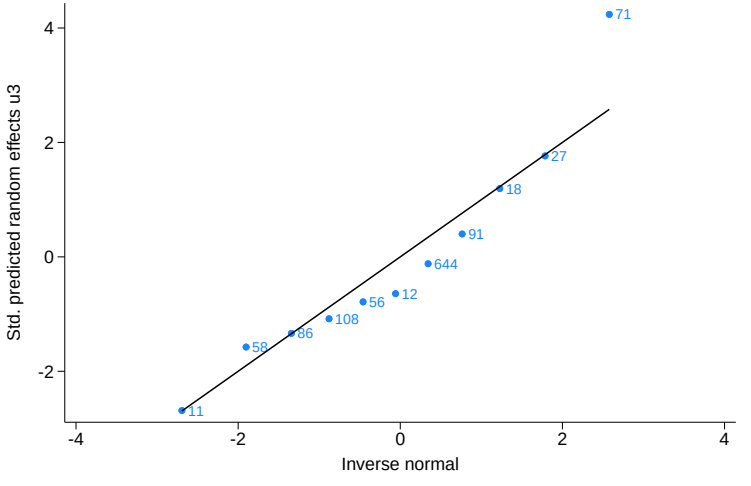
	district	u3	se_u3
1.	11	-.18998595	.07071818
5.	12	-.08467077	.13168501
9.	18	.1407273	.11790486
12.	27	.24064814	.13641505
16.	56	-.1072942	.13633364
20.	58	-.23650899	.15003184
31.	71	.5342778	.12606073
34.	86	-.2004695	.1499012
42.	91	.05711692	.14284823
48.	108	-.14168396	.13094894
53.	644	-.01215679	.10054689

As an example, we listed the random-intercepts variable `u3` with the corresponding standard error variable `se_u3` for the $M = 11$ school districts. The purpose of variable `tolist` is to list only the unique values of `u3` and `se_u3` for each district. Had we not added the `if tolist` qualifier, row j ($j = 1, \ldots, M$) would have been repeated m_j times, where m_j is the number of schools within the jth district. The random intercepts are district-specific deviations from the overall mean effect size. For example, for district

18, the predicted standardized mean difference is 0.1407 higher than the overall effect size $\hat{b}_0 = 0.1847$, estimated in example 2 of [META] **meta meregress**, conditional on zero contribution from level-2 random intercepts.

Let's use the predicted random effects and their standard errors to compute a standardized random-effects variable, `ustan3`, to check for outliers. This new variable corresponds to the standardized random effects at the `district` level (level 3). We will use the `qnorm` command (see [R] **Diagnostic plots**) to obtain the normal quantile plot.

```
. generate double ustan3 = u3/se_u3
. label variable ustan3 "Std. predicted random effects u3"
. qnorm ustan3 if tolist, mlabel(district) xtitle("Inverse normal")
```



From the plot, district 71 appears to be an outlier. Let's list the values for districts 71 and, for comparison, 27.

```
. list district school stdmdiff if inlist(district, 71, 27), separator(4)
```

	district	school	stdmdiff
12.	27	1	.16
13.	27	2	.65
14.	27	3	.36
15.	27	4	.6
31.	71	1	.3
32.	71	2	.98
33.	71	3	1.19

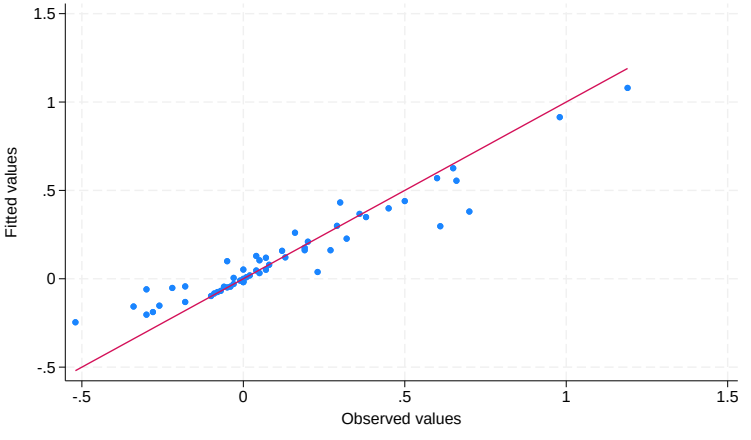
District 71 has 3 schools with students following the modified calendar scoring substantially higher ($\hat{\theta}_{71,1} = 0.3$, $\hat{\theta}_{71,2} = 0.98$, and $\hat{\theta}_{71,3} = 1.19$) compared with students from schools with modified calendars in other districts such as district 27.

◁

▷ Example 2: Checking model fit

Continuing with example 1, we specify the `fitted` option to obtain the fitted values and plot them against the observed values of `stdmdiff`. By default, for the kth school within the jth district, the fitted values, $\hat{\beta}_0+\hat{u}_j^{(3)}+\hat{u}_{jk}^{(2)}$, are computed based on random-effects contributions from all levels of hierarchy. Alternatively, we could specify the `relevel(district)` option to compute the fitted values, $\hat{\beta}_0+\hat{u}_j^{(3)}$, based on contributions from level-3 random-effects only.

```
. predict double fit, fitted
. twoway (scatter fit stdmdiff)
>        (function y = x, range(stdmdiff)),
> legend(off) xtitle(Observed values)  ytitle(Fitted values)
```



In the above code, we computed the fitted values using `predict, fitted`. We then produced a scatterplot of the fitted values versus the observed values of `stdmdiff`. We added a reference line $y=x$ to assess goodness of fit. Studies that are close to the reference line have their fitted values close to the observed values. Overall, it seems that goodness of fit is satisfactory.

You could also use the `rstandard` option with `predict` to compute standardized residuals. In theory, the standardized residuals are useful for checking the normality assumption of the level-1 error terms.

◁

Methods and formulas

Continuing with the notation in *Methods and formulas* of [META] **meta meregress**, the three-level meta-regression model can be expressed compactly as

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \mathbf{Z}_j\mathbf{u}_j + \boldsymbol{\epsilon}_j, \quad j = 1,\ldots,M$$

where $m_{j.} \times (q_3 + m_j q_2)$ matrix $\mathbf{Z}_j = (\mathbf{Z}_j^{(3)}, \mathbf{Z}_j^{(2)})$ and $(q_3 + m_j q_2) \times 1$ vector $\mathbf{u}_j = (\mathbf{u}_j^{(3)\prime}, \mathbf{u}_j^{(2)\prime})'$ with a $(q_3 + m_j q_2) \times (q_3 + m_j q_2)$ covariance matrix $\boldsymbol{\Sigma}_j$

$$\boldsymbol{\Sigma}_j = \mathrm{Var}\left(\mathbf{u}_j\right) = \begin{bmatrix} \boldsymbol{\Sigma}^{(3)} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{m_j} \otimes \boldsymbol{\Sigma}^{(2)} \end{bmatrix}$$

If we let $\mathbf{X} = (\mathbf{X}_1', \mathbf{X}_2', \ldots, \mathbf{X}_M')'$, $\mathbf{\Lambda} = \oplus_{j=1}^{M} \mathbf{\Lambda}_j$, $\mathbf{Z} = \oplus_{j=1}^{M} \mathbf{Z}_j$, and $\mathbf{u} = (\mathbf{u}_1', \mathbf{u}_2', \ldots, \mathbf{u}_M')'$, then the formulas used by `predict` for predicting random effects, residuals, etc. are described in *Methods and formulas* of [ME] **mixed postestimation** with $\mathbf{G} = \oplus_{j=1}^{M} \mathbf{\Sigma}_j$, $\mathbf{R} = \mathbf{\Lambda}$, and $\sigma_\epsilon^2 = 1$.

When the `reses()` option is specified with `reffects`, the estimated covariance matrix of $\hat{\mathbf{u}}_j - \mathbf{u}_j$ is computed:

$$\widehat{\mathrm{Var}}\left(\hat{\mathbf{u}}_j - \mathbf{u}_j\right) = \widehat{\mathbf{\Sigma}} - \widehat{\mathbf{\Sigma}}\mathbf{Z}_j'\mathbf{W}_j\left\{\left(\mathbf{W}_j\right)^{-1} - \mathbf{X}_j\mathrm{Var}(\widehat{\boldsymbol{\beta}})\mathbf{X}_j'\right\}\mathbf{W}_j\mathbf{Z}_j\widehat{\mathbf{\Sigma}}$$

The comparative standard errors of the random effects can be obtained by taking the square root of the diagonal elements of $\widehat{\mathrm{Var}}(\hat{\mathbf{u}}_j - \mathbf{u}_j)$.

If the `diagnostic` suboption is specified within `reses()`, then the estimated covariance matrix of $\hat{\mathbf{u}}_j$ is computed:

$$\widehat{\mathrm{Var}}\left(\hat{\mathbf{u}}_j\right) = \widehat{\mathbf{\Sigma}}\mathbf{Z}_j'\mathbf{W}_j\left\{\left(\mathbf{W}_j\right)^{-1} - \mathbf{X}_j\mathrm{Var}(\widehat{\boldsymbol{\beta}})\mathbf{X}_j'\right\}\mathbf{W}_j\mathbf{Z}_j\widehat{\mathbf{\Sigma}}$$

The diagnostic standard errors of the random effects can be obtained by taking the square root of the diagonal elements of $\widehat{\mathrm{Var}}(\hat{\mathbf{u}}_j)$.

See Goldstein (2011), Skrondal and Rabe-Hesketh (2009), and Rabe-Hesketh and Skrondal (2022) for more details.

References

Goldstein, H. 2011. *Multilevel Statistical Models*. 4th ed. Chichester, UK: Wiley. https://doi.org/10.1002/9780470973394.

Rabe-Hesketh, S., and A. Skrondal. 2022. *Multilevel and Longitudinal Modeling Using Stata*. 4th ed. College Station, TX: Stata Press.

Skrondal, A., and S. Rabe-Hesketh. 2009. Prediction in multilevel generalized linear models. *Journal of the Royal Statistical Society*, A ser., 172: 659–687. https://doi.org/10.1111/j.1467-985X.2009.00587.x.

Also see

[META] **meta meregress** — Multilevel mixed-effects meta-regression

[META] **meta multilevel** — Multilevel random-intercepts meta-regression

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

[U] 20 Estimation and postestimation commands

meta mvregress — Multivariate meta-regression

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Methods and formulas
References | Also see

Description

`meta mvregress` performs multivariate meta-regression. You can think of multivariate meta-regression as an extension of meta-regression, where multiple potentially dependent effect sizes are available for each study. `meta mvregress` performs both random-effects and fixed-effects multivariate meta-regression with various covariance structures and estimation methods for the random effects. `meta mvregress` is a stand-alone command in the sense that it does not require you to declare your data as `meta` data using `meta set` or `meta esize`.

Quick start

Perform random-effects multivariate meta-analysis of the effect-size variables `y1` and `y2` with within-study covariance structure defined by variables `v11`, `v12`, and `v22`

```
meta mvregress y1 y2, wcovvariables(v11 v12 v22)
```

Same as above, but perform random-effects multivariate meta-regression on continuous variable `x1` and factor variable `x2`

```
meta mvregress y1 y2 = x1 i.x2, wcovvariables(v11 v12 v22)
```

Same as above, but estimate random-effects using ML instead of the default REML

```
meta mvregress y1 y2 = x1 i.x2, wcovvariables(v11 v12 v22) random(mle)
```

Same as above, but specify an independent random-effects covariance structure instead of the default unstructured covariance matrix

```
meta mvregress y1 y2 = x1 i.x2, wcovvariables(v11 v12 v22) ///
random(mle, covariance(independent))
```

Same as above, but use a truncated Jackson–Riley adjustment to the standard errors of coefficients

```
meta mvregress y1 y2 = x1 i.x2, wcovvariables(v11 v12 v22) ///
random(mle, covariance(independent) se(truncjriley))
```

Perform a fixed-effects multivariate meta-analysis of variables `y1` and `y2` with standard error variables `s1` and `s2`, and assume a within-study correlation value of 0

```
meta mvregress y1 y2, fixed wsevariables(s1 s2) wcorrelations(0)
```

Perform multivariate meta-analysis of three effect-size variables `y1`, `y2`, and `y3` with six within-study variance–covariance variables `v11`, `v12`, `v13`, `v22`, `v23`, and `v33`

```
meta mvregress y1 y2 y3, wcovvariables(v11 v12 v13 v22 v23 v33)
```

Same as above, but using *varlist* shortcut notations and assuming the variables appear in the dataset in the order shown above

```
meta mvregress y1-y3, wcovvariables(v11-v33)
meta mvregress y*, wcovvariables(v*)
```

Menu

Statistics > Meta-analysis

Syntax

Random-effects multivariate meta-regression

```
meta mvregress depvars = moderators [if] [in], wcovspec [random(randomspec)
    options]
```

Fixed-effects multivariate meta-regression

```
meta mvregress depvars = moderators [if] [in], wcovspec fixed [options]
```

Multivariate meta-analysis (constant-only model)

```
meta mvregress depvars [if] [in], wcovspec [modelopts]
```

wcovspec	Description
Model	
* wcovvariables(*varlist*)	specify within-study variance and covariance variables
* wsevariables(*varlist*)	specify within-study standard-error variables
* wcorrelations(# | *numlist*)	specify within-study correlation values

Either `wcovvariables()` or both `wsevariables()` and `wcorrelations()` are required.

For `random(`*randomspec*`)`, the syntax of *randomspec* is

remethod [, covariance(*recov*) se(*seadj*)]

remethod	Description
reml	restricted maximum likelihood; the default
mle	maximum likelihood
jwriley	Jackson–White–Riley

recov	Description
unstructured	all variances and covariances to be distinctly estimated; the default
independent	one unique variance parameter per random effect; all covariances 0
exchangeable	equal variances for random effects and one common pairwise covariance
identity	equal variances for random effects; all covariances 0
fixed(*matname*)	fixed random-effects covariance matrix *matname*

seadj	Description
jriley	Jackson–Riley standard-error adjustment
truncjriley	truncated Jackson–Riley standard-error adjustment

options	Description
Model	
noconstant	suppress constant term
tdistribution(#)	compute t tests instead of z tests for regression coefficients
Reporting	
level(#)	set confidence level; default is level(95)
stddeviations	show random-effects parameter estimates as standard deviations and correlations; the default
variance	show random-effects parameter estimates as variances and covariances
nohomtest	suppress output for homogeneity test
noretable	suppress random-effects table
nofetable	suppress fixed-effects table
estmetric	show parameter estimates as stored in e(b)
noheader	suppress output header
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Maximization	
maximize_options	control the maximization process; seldom used
coeflegend	display legend instead of statistics

moderators may contain factor variables; see **[U] 11.4.3 Factor variables**.

collect is allowed; see **[U] 11.1.10 Prefix commands**.

coeflegend does not appear in the dialog box.

See **[U] 20 Estimation and postestimation commands** for more capabilities of estimation commands.

modelopts is any of *options* except noconstant.

Options

Model

wcovvariables(*varlist*) or wsevariables(*varlist*) and wcorrelations(#|*numlist*) specify information about the within-study covariance matrices $\mathbf{\Lambda}_j$, which are required for multivariate meta-regression.

wcovvariables(*varlist*) specifies variables that define the within-study covariance matrices $\mathbf{\Lambda}_j$. If d is the number of *depvars*, then $d(d+1)/2$ variables must be provided. The order in which the variables are specified is important. For example, if we have $d = 3$ dependent variables y1, y2, and y3, then 6 variables must be provided within wcovvariables() in the following order: Var(y1), Cov(y1, y2), Cov(y1, y3), Var(y2), Cov(y2, y3), and Var(y3). This option may not be combined with options wsevariables() and wcorrelations().

wsevariables(*varlist*) specifies variables that define the within-study standard errors of *depvars*. This option is useful, in combination with wcorrelations(), when the within-study covariances are not reported but only standard errors are available for *depvars*. If d is the number of *depvars*, then d variables must be specified, which represent the within-study standard errors of each variable in *depvars*. The order of the variables must follow the order in which *depvars* were spec-

ified. This option must be specified in combination with option `wcorrelations()`, which together define the within-study covariance matrices. `wsevariables()` may not be combined with `wcovvariables()`.

`wcorrelations(`#|*numlist*`)` specifies values for the within-study correlations between *depvars*. This option is also used to specify assumed correlations when only within-study standard errors are available, which are specified in option `wsevariables()`. If `wcorrelations(`#`)` is specified, # is assumed to be the common within-study correlation value between all *depvars*. If *numlist* is specified, then $d(d-1)/2$ values must be provided, where d is the number of *depvars*. The order in which the correlation values are specified is important. For example, if we have $d = 3$ dependent variables `y1`, `y2`, and `y3`, then 3 values must be provided in the following order: Corr(`y1`, `y2`), Corr(`y1`, `y3`), and Corr(`y2`, `y3`). This option must be specified in combination with option `wsevariables()`, which together define the within-study covariance matrices. `wcorrelations()` may not be combined with `wcovvariables()`.

`random` and `random(`*randomspec*`)` specify that a random-effects model be assumed for the multivariate meta-regression. The syntax for *randomspec* is *remethod* [`, covariance(`*recov*`) se(`*seadj*`)`].

remethod specifies the type of estimator for the between-study covariance matrix $\boldsymbol{\Sigma}$. *remethod* is one of `reml`, `mle`, or `jwriley`. `random` is a synonym for `random(reml)`.

`reml`, the default, specifies that the REML method (Jackson, Riley, and White 2011) be used to estimate $\boldsymbol{\Sigma}$. This method produces an unbiased positive semidefinite estimate of the between-study covariance matrix and is commonly used in practice. The `reml` method requires iteration.

`mle` specifies that the ML method (Jackson, Riley, and White 2011) be used to estimate $\boldsymbol{\Sigma}$. It produces a positive semidefinite estimate of the between-study covariance matrix. With a few studies or small studies, this method may produce biased estimates. With many studies, the ML method is more efficient than the REML method. Method `mle` requires iteration.

`jwriley` specifies that the Jackson–White–Riley method (Jackson, White, and Riley 2013) be used to estimate $\boldsymbol{\Sigma}$. This method is a multivariate generalization of the popular DerSimonian–Laird method in univariate meta-analysis. The method does not make any assumptions about the distribution of random effects and does not require iteration. But it may produce an estimate of $\boldsymbol{\Sigma}$ that is not positive semidefinite and is thus "truncated" (via spectral decomposition) in that case.

`covariance(`*recov*`)` specifies the structure of the covariance matrix for the random effects. *recov* is one of the following: `unstructured`, `independent`, `exchangeable`, `identity`, or `fixed(`*matname*`)`.

`unstructured` allows for all variances and covariances to be distinct. If there are d random-effects terms (corresponding to the d *depvars*), the unstructured covariance matrix will have $d(d+1)/2$ unique parameters. This is the default covariance structure.

`independent` allows for a distinct variance for each random effect corresponding to a dependent variable and assumes that all covariances are 0.

`exchangeable` specifies one common variance for all random effects and one common pairwise covariance.

`identity` is short for "multiple of the identity"; that is, all variances are equal and all covariances are 0.

`fixed(`*matname*`)` specifies a fixed (known) $\boldsymbol{\Sigma} =$ *matname*. This covariance structure requires no iteration.

se(*seadj*) specifies that the adjustment *seadj* be applied to the standard errors of the regression coefficients. Additionally, the tests of significance of the regression coefficients are based on a Student's t distribution instead of the normal distribution. The Jackson–Riley adjustments are multivariate generalizations of the Knapp–Hartung standard-error adjustments in univariate meta-regression. *seadj* is one of jriley or truncjriley.

jriley specifies that the Jackson–Riley adjustment (Jackson and Riley 2014) be applied to the standard errors of the coefficients.

truncjriley specifies that the truncated Jackson–Riley adjustment (Jackson and Riley 2014) be applied to the standard errors of the coefficients.

fixed specifies that a fixed-effects model be assumed for the multivariate meta-regression. In this case, $\mathbf{\Sigma} = \mathbf{0}$, and no iteration is performed to estimate the random-effects parameters.

noconstant; see [R] **Estimation options**. This option is not allowed with constant-only multivariate meta-regression.

tdistribution(#) computes t tests instead of z tests for the regression coefficients. The t tests are based on # degrees of freedom, which does not have to be an integer.

Reporting

level(#); see [R] **Estimation options**.

stddeviations, variance; see [ME] **mixed**.

nohomtest suppresses the homogeneity test based on the Q statistic from the output.

noretable, nofetable, estmetric, noheader; see [ME] **mixed**.

display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(#), fvwrapon(*style*), cformat(%*fmt*), pformat(%*fmt*), sformat(%*fmt*), and nolstretch; see [R] **Estimation options**.

Maximization

maximize_options: difficult, technique(*algorithm_spec*), iterate(#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(#), ltolerance(#), nrtolerance(#), and nonrtolerance; see [R] **Maximize**. Those that require special mention for meta mvregress are listed below.

For the technique() option, the default is technique(nr). The bhhh algorithm is not available.

matsqrt, the default, and matlog; see [ME] **mixed**, except meta mvregress implies a single model level.

maximize_options are not available with fixed-effects multivariate meta-regression.

The following option is available with meta mvregress but is not shown in the dialog box:

coeflegend; see [R] **Estimation options**.

Remarks and examples

Remarks are presented under the following headings:

Introduction
Examples of using meta mvregress

Introduction

Multivariate meta-regression is a technique used to study the relationship between multiple, usually dependent, effect sizes reported for each study and covariates. Multivariate meta-regression is analogous to multivariate regression that is used when individual data are available, but in multivariate meta-regression, the observations are the studies, the outcomes of interest are effect sizes, and the covariates are recorded at the study level. The study-level covariates in meta-regression are known as moderators. Examples of moderators include study publication year, study test environment, and drug administration method. For a comprehensive introduction to multivariate meta-regression, see Gleser and Olkin (2009) and Jackson, Riley, and White (2011).

A study may report multiple effect sizes in two different scenarios. In the first scenario, a study may compare various treatment groups against a common control group. For example, in a study that investigates the effect of multiple dietary regimens on weight loss, independent groups of individuals may be assigned to one of several diets: Keto diet, vegan diet, high-protein diet, or intermittent fasting. Multiple effect sizes that compare each of these diets with a control group (not following an assigned diet) can be computed. These effect sizes are usually correlated because they share a common control group. Studies falling under this category are called "multiple-treatment studies" or "mixed-treatment studies" in the multivariate meta-analysis literature.

In the second scenario, subjects are allocated to a treatment group or a control group as in the case of univariate meta-analysis, but multiple outcomes (endpoints) are compared across the two groups. For example, consider a study that explores the impact of a new teaching technique on math (outcome 1), physics (outcome 2), and chemistry (outcome 3) testing scores. Students are randomly assigned to one of two groups: those who were taught using the new technique (treatment group) and those who were not (control group). Three effect sizes that compare the three testing scores across the two groups are computed. These effect sizes are dependent because they were reported on the same set of students. Studies of this kind are referred to as "multiple-endpoint studies" in the literature.

Traditionally, the standard approach for handling multiple effect sizes reported per study was to perform separate univariate meta-analysis for each effect size. This approach ignores the dependence between the effect sizes and usually leads to biased pooled effects with overestimated variances. Another approach (Rosenthal and Rubin 1986) is to summarize the multiple effects by a single value for each study and then combine these values via standard univariate meta-analysis. This approach will result in information loss because of data reduction and may yield univariate summaries that are difficult to interpret in light of the original dependent effect sizes.

By properly accounting for the dependence between the effect sizes, multivariate meta-regression often provides parameter estimators with more optimal properties when compared with the previous two approaches. This is because it exploits the correlation between the multiple effect sizes, and thus the dependent effect sizes may borrow strength from each other to produce pooled effect sizes with smaller variances (Jackson, Riley, and White 2011).

As is the case with meta-regression, the goal of multivariate meta-regression is also to explore and explain the between-study heterogeneity as a function of moderators. Two types of multivariate regression models, fixed-effects and random-effects, are available. A fixed-effects multivariate meta-regression as-

sumes that all heterogeneity between study effect sizes can be accounted for by the included moderators. A random-effects multivariate meta-regression accounts for potential additional variability unexplained by the included moderators, also known as residual heterogeneity.

`meta mvregress` fits multivariate meta-regression. The default model assumed by `meta mvregress` is a random-effects model using the REML method with an unstructured between-study covariance matrix. Use the `random()` option to specify other random-effects methods such as the MLE or a noniterative Jackson–White–Riley method, which can be viewed as an extension of the univariate DerSimonian–Laird method to the multivariate setting. You may also use the `random()` option to specify an alternative covariance structure such as `exchangeable`, `independent`, `identity`, or `fixed()` in the `covariance()` suboption.

Covariance structure `fixed()` specifies a fixed between-study covariance matrix and thus can be used to perform sensitivity analysis similarly to option `tau2()` in [META] **meta regress**. Specifying a covariance structure other than the default `unstructured` is particularly useful when the number of observations, n, is small relative to the number of estimated fixed-effects parameters and variance components.

Jackson and Riley (2014) proposed an adjustment to the standard errors of the fixed-effects parameters that provides more accurate inference when the number of studies is relatively small. This adjustment is available with the `se()` option. The Jackson–Riley adjustment can be seen as a multivariate extension of the Knapp–Hartung adjustment (Knapp and Hartung 2003) in univariate meta-regression, and the two adjustments are identical when there is only one effect-size variable.

Consider data from K independent studies and d outcomes (effect sizes). Let $\hat{\theta}_{ij}$ be the estimated effect size reported by study j for outcome i, and let a $d \times 1$ vector $\hat{\boldsymbol{\theta}}_j = (\hat{\theta}_{1j}, \hat{\theta}_{2j}, \ldots, \hat{\theta}_{dj})'$ be an estimate of the true population effect size $\boldsymbol{\theta}_j$ for study j.

A model for the fixed-effects multivariate meta-regression (Raudenbush, Becker, and Kalaian 1988) can be expressed as

$$\hat{\theta}_{ij} = \beta_{i0} + \beta_{i1}x_{1j} + \cdots + \beta_{i,p-1}x_{p-1,j} + \epsilon_{ij} = \mathbf{x}_j\boldsymbol{\beta}_i + \epsilon_{ij}$$

for outcome $i = 1, \ldots, d$ and study $j = 1, \ldots, K$. Here $\mathbf{x}_j = (1, x_{1j}, \ldots, x_{p-1,j})$ is a $1 \times p$ vector of categorical and continuous moderators (covariates), $\boldsymbol{\beta}_i$ is an outcome-specific $p \times 1$ vector of unknown regression coefficients, and $\boldsymbol{\epsilon}_j = (\epsilon_{1j}, \epsilon_{2j}, \ldots, \epsilon_{dj})'$ is a $d \times 1$ vector of within-study errors that have a d-variate normal distribution with zero mean vector and a $d \times d$ covariance matrix $\mathrm{Var}(\boldsymbol{\epsilon}_j) = \boldsymbol{\Lambda}_j$. The within-study covariance matrices $\boldsymbol{\Lambda}_j$'s are treated as known and do not require estimation. The values of these matrices are specified as variables in the `wcovvariables()` option or in a combination of the `wsevariables()` and `wcorrelations()` options.

In a matrix notation, the above fixed-effects model can be defined as

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \boldsymbol{\epsilon}_j, \quad \boldsymbol{\epsilon}_j \sim N_d(\mathbf{0}, \boldsymbol{\Lambda}_j)$$

where $\mathbf{X}_j = \mathbf{x}_j \otimes I_d$ ($\otimes$ is the Kronecker product) is a $d \times dp$ matrix and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \ldots, \boldsymbol{\beta}_d')'$ is a $dp \times 1$ vector of all unknown regression coefficients.

Residual heterogeneity may be accounted for by including an additive between-study covariance component, $\boldsymbol{\Sigma}$, that leads to a random-effects multivariate meta-regression (Berkey et al. 1998):

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \boldsymbol{\epsilon}_j^* = \mathbf{X}_j\boldsymbol{\beta} + \mathbf{u}_j + \boldsymbol{\epsilon}_j, \quad \text{where } \boldsymbol{\epsilon}_j^* \sim N_d\left(\mathbf{0}, \boldsymbol{\Lambda}_j + \boldsymbol{\Sigma}\right)$$

As we mentioned earlier, a random-effects multivariate meta-regression assumes that the moderators explain only part of heterogeneity, and random effects $\mathbf{u}_j = (u_{1j}, u_{2j}, \ldots, u_{dj})' \sim N_d(\mathbf{0}, \boldsymbol{\Sigma})$ $(j = 1, \ldots, K)$ account for the remainder.

Harbord and Higgins (2016) point out that some authors (Thompson and Sharp 1999; Higgins and Thompson 2004) argue that a fixed-effects meta-regression should not be used because, in practice, the included moderators rarely capture all the between-study heterogeneity and that the failure of the fixed-effects regression to capture the extra between-study heterogeneity can lead to excessive type I errors. This observation is also echoed by Jackson, Riley, and White (2011) in the multivariate setting.

Examples of using meta mvregress

Examples are presented under the following headings:

Example 1: Univariate versus multivariate meta-analysis
Example 2: Random-effects multivariate meta-regression
Example 3: Identical results from univariate and multivariate analyses
Example 4: Heterogeneity statistics
Example 5: Jackson–White–Riley random-effects method
Example 6: Jackson–Riley standard-error adjustment
Example 7: When within-study covariances are not available
Example 8: Missing outcome data
Example 9: Between-study covariance structures
Example 10: Sensitivity meta-analysis
Example 11: Fixed-effects multivariate meta-regression

▷ Example 1: Univariate versus multivariate meta-analysis

Consider a dataset from Antczak-Bouckoms et al. (1993) of five randomized controlled trials that explored the impact of two procedures (surgical and nonsurgical) for treating periodontal disease. This dataset was also analyzed by Berkey et al. (1998).

In these trials, subjects' mouths were split into sections. These sections were randomly allocated to the two treatment procedures. At least one section was treated surgically and at least one other section was treated nonsurgically for each patient. The main objectives of the periodontal treatment were to reduce probing depths and increase attachment levels (Berkey et al. 1998).

Two outcomes of interest are improvements from baseline (pretreatment) in probing depth (`y1`) and attachment level (`y2`) around the teeth. Because the two outcomes `y1` and `y2` are measured on the same subject, they should not be treated as independent. This is an example of multiple-endpoint studies where multiple outcomes (two in this case) are compared across two groups (surgical versus nonsurgical). We first describe our dataset.

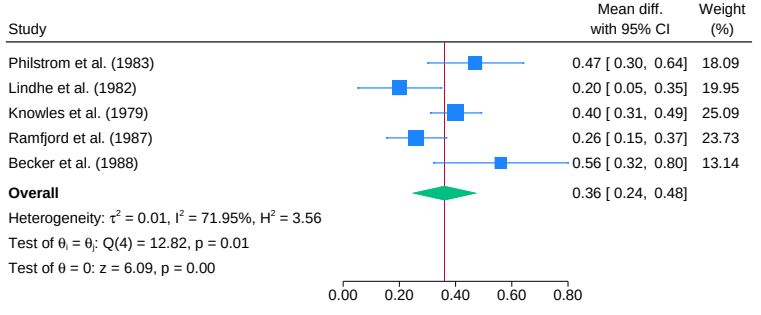
```
. use https://www.stata-press.com/data/r18/periodontal
(Treatment of moderate periodontal disease)
. describe
Contains data from https://www.stata-press.com/data/r18/periodontal.dta
 Observations:             5                  Treatment of moderate
                                                periodontal disease
    Variables:             9                  13 Jan 2023 18:11
                                              (_dta has notes)
```

Variable name	Storage type	Display format	Value label	Variable label
trial	str23	%23s		Trial label
pubyear	byte	%9.0g		Publication year centered at 1983
y1	float	%6.2f		Mean improvement in probing depth (mm)
y2	float	%6.2f		Mean improvement in attachment level (mm)
v11	float	%6.4f		Variance of y1
v12	float	%6.4f		Covariance of y1 and y2
v22	float	%6.4f		Variance of y2
s1	double	%10.0g		Standard error of y1
s2	double	%10.0g		Standard error of y2

```
Sorted by:
```

We will start by performing a separate meta-analysis for each outcome. We declare our data as meta data using the `meta set` command and then construct a forest plot for each outcome; see [META] **meta set** and [META] **meta forestplot**, respectively.

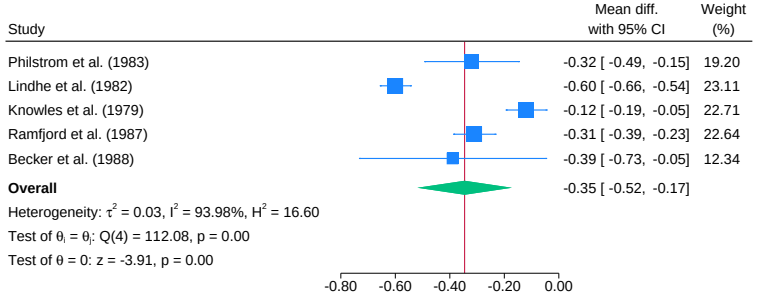
```
. quietly meta set y1 s1, studylabel(trial) eslabel("Mean diff.")
. meta forestplot, esrefline
  Effect-size label: Mean diff.
        Effect size: y1
          Std. err.: s1
        Study label: trial
```



Positive `y1` values indicate that the mean improvement (reduction) in probing depth for the surgical group is larger than that for the nonsurgical group. It appears that the surgical treatment performs consistently better (`y1` > 0) across all studies. The overall mean difference is 0.36 with a 95% CI of $[0.24, 0.48]$, which means that, on average, the reduction in probing depth was 0.36 mm higher than that for the nonsurgical group.

Similarly, we will construct a forest plot for variable y2.

```
. quietly meta set y2 s2, studylabel(trial) eslabel("Mean diff.")
. meta forestplot, esrefline
  Effect-size label: Mean diff.
        Effect size: y2
          Std. err.: s2
        Study label: trial
```



Negative y2 values indicate that the mean improvement (increase) in attachment level for the surgical group is smaller than that for the nonsurgical group. Because $\texttt{y2} < 0$ across all studies, the nonsurgical treatment performs consistently better in terms of attachment level. It appears that there is considerable heterogeneity in attachment levels (y2) based on the nonoverlapping CIs in the forest plot and a large value of the I^2 statistic (93.98%).

Notice that the obtained heterogeneity statistics are from univariate meta-analyses conducted separately. In example 4, we show how to assess heterogeneity from a multivariate analysis by using the `estat heterogeneity` command.

The two separate meta-analyses do not account for the dependence between y1 and y2. Let's fit a bivariate meta-analysis (constant-only bivariate meta-regression) using the meta mvregress command.

```
. meta mvregress y1 y2, wcovvariables(v11 v12 v22)
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood =  2.0594015
Iteration 1:  Log restricted-likelihood =  2.0822862
Iteration 2:  Log restricted-likelihood =  2.0823276
Iteration 3:  Log restricted-likelihood =  2.0823276
Multivariate random-effects meta-analysis       Number of obs     =         10
Method: REML                                    Number of studies =          5
                                                Obs per study:
                                                              min =          2
                                                              avg =        2.0
                                                              max =          2
                                                Wald chi2(0)      =          .
Log restricted-likelihood = 2.0823276           Prob > chi2       =          .
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
y1						
_cons	.3534282	.0588486	6.01	0.000	.238087	.4687694
y2						
_cons	-.3392152	.0879051	-3.86	0.000	-.5115061	-.1669243

```
Test of homogeneity: Q_M = chi2(8) = 128.23                 Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Unstructured:	
sd(y1)	.1083191
sd(y2)	.1806968
corr(y1,y2)	.6087987

The output shows information about the optimization algorithm, the iteration log, and the model (random-effects) and method (REML) used for estimation. It also displays the number of studies, $K = 5$, and the total number of observations on the outcomes, $n = 10$, which is equal to Kd because there are no missing observations. The minimum, maximum, and average numbers of observations per study are also reported. Because there were no missing observations, all of these numbers are identical and equal to 2.

The first table displays the regression (fixed-effects) coefficient estimates from the bivariate meta-analysis. These estimates correspond to the overall bivariate effect size $\widehat{\boldsymbol{\theta}} = (\widehat{\theta}_1, \widehat{\theta}_2)'$. The estimates are close to the univariate ones reported on the forest plots. But from a bivariate analysis, we obtained slightly narrower 95% CIs for the overall effect sizes. The multivariate homogeneity test, which tests whether $\boldsymbol{\theta}_j = (\theta_{1j}, \theta_{2j})'$ is constant across studies, is rejected ($p < 0.0001$). This agrees with earlier univariate results, particularly from the second forest plot, which exhibited considerable heterogeneity.

The second table displays the random-effects parameters, traditionally known as variance components in the context of multilevel or mixed-effects models. By default, similar to the mixed command, meta mvregress reports standard deviations of y1 and y2 and their correlation: sd(y1), sd(y2), and corr(y1,y2), respectively. But you can instead specify the variance option to report variances and the covariance.

◁

▷ Example 2: Random-effects multivariate meta-regression

Berkey et al. (1998) noted that although the meta-analysis of Antczak-Bouckoms et al. (1993) accounted for many factors that could potentially lead to heterogeneity, a substantive variability was still present, as we highlighted in example 1. They suggested to use the year of publication centered at 1983 (pubyear), a surrogate for the time when the trial was performed, as a moderator to explain a portion of this heterogeneity. They reasoned that as the surgical experience accumulates, the surgical procedure will become more efficient so the most recent studies may show greater surgical benefits.

Let's first perform separate univariate meta-regressions for outcomes y1 and y2 with pubyear as a moderator. We can do this by specifying only one dependent variable with meta mvregress or by using meta regress. We will use meta mvregress because it does not require setting the data.

```
. meta mvregress y1 = pubyear, wsevariables(s1)
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -1.6637351
Iteration 1:  Log restricted-likelihood = -1.6426005
Iteration 2:  Log restricted-likelihood = -1.6414308
Iteration 3:  Log restricted-likelihood = -1.6414292
Iteration 4:  Log restricted-likelihood = -1.6414292
Multivariate random-effects meta-regression     Number of obs     =          5
Method: REML                                    Number of studies =          5
                                                Obs per study:
                                                              min =          1
                                                              avg =        1.0
                                                              max =          1
                                                Wald chi2(1)      =       0.04
Log restricted-likelihood = -1.6414292          Prob > chi2       =     0.8332
------------------------------------------------------------------------------
          y1 | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
     pubyear |    .004542   .021569     0.21   0.833    -.0377325    .0468165
       _cons |    .362598   .0725013    5.00   0.000     .2204981     .504698
------------------------------------------------------------------------------
Test of homogeneity: Q_M = chi2(3) = 11.80                Prob > Q_M = 0.0081

------------------------------------------------------------------------------
   Random-effects parameters |   Estimate
-----------------------------+------------------------------------------------
Identity:                    |
                      sd(y1) |    .1406077
------------------------------------------------------------------------------
```

```
. meta mvregress y2 = pubyear, wsevariables(s2)
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -2.4661957
Iteration 1:  Log restricted-likelihood = -2.3230318
Iteration 2:  Log restricted-likelihood = -2.3229928
Iteration 3:  Log restricted-likelihood = -2.3229928
Multivariate random-effects meta-regression     Number of obs     =          5
Method: REML                                    Number of studies =          5
                                                Obs per study:
                                                              min =          1
                                                              avg =        1.0
                                                              max =          1
                                                Wald chi2(1)      =       0.20
Log restricted-likelihood = -2.3229928          Prob > chi2       =     0.6524
```

y2	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
pubyear	-.0134909	.0299534	-0.45	0.652	-.0721985	.0452167
_cons	-.3399793	.0978864	-3.47	0.001	-.5318331	-.1481256

```
Test of homogeneity: Q_M = chi2(3) = 108.29                  Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Identity:	
sd(y2)	.201787

Here we specified the standard error variables `s1` and `s2` in the `wsevariables()` options to match the univariate setup more closely, but we could have used `wcovvariables(v11)` and `wcovvariables(v22)`, following example 1.

Results from the univariate meta-regressions suggest that variable `pubyear` does not seem to explain the between-study heterogeneity between the effect sizes `y1` and `y2`; the p-values for testing the `pubyear` coefficients to be 0 are $p = 0.833$ and $p = 0.652$, respectively.

The two separate meta-regressions do not account for the dependence between y1 and y2. Below, we fit a bivariate meta-regression that accounts for this dependence.

```
. meta mvregress y1 y2 = pubyear, wcovvariables(v*)
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -3.5544446
Iteration 1:  Log restricted-likelihood = -3.5402086
Iteration 2:  Log restricted-likelihood = -3.5399568
Iteration 3:  Log restricted-likelihood = -3.5399567
Multivariate random-effects meta-regression     Number of obs     =         10
Method: REML                                    Number of studies =          5
                                                Obs per study:
                                                              min =          2
                                                              avg =        2.0
                                                              max =          2
                                                Wald chi2(2)      =       0.40
Log restricted-likelihood = -3.5399567          Prob > chi2       =     0.8197
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
y1						
pubyear	.0048615	.0218511	0.22	0.824	-.0379658	.0476888
_cons	.3587569	.07345	4.88	0.000	.2147975	.5027163
y2						
pubyear	-.0115367	.0299635	-0.39	0.700	-.070264	.0471907
_cons	-.3357368	.0979979	-3.43	0.001	-.5278091	-.1436645

```
Test of homogeneity: Q_M = chi2(6) = 125.76                  Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Unstructured:	
sd(y1)	.1429917
sd(y2)	.2021314
corr(y1,y2)	.561385

Instead of listing all the variance–covariance variables v11, v12, and v22 in the wcovvariables() option, we used the stub notation v* to refer to all of them. This notation is especially convenient for models with more dependent variables. You just need to make sure that these are the only variables starting with v in the dataset and that the variables are properly ordered (think of a vectorized upper triangle of the variance–covariance matrix) before using the stub notation; see the description of wcovvariables().

The estimates of the regression coefficients of variable pubyear are 0.0049 with a 95% CI of $[-0.0380, 0.0477]$ for outcome y1 and -0.0115 with a 95% CI of $[-0.0703, 0.0472]$) for outcome y2. The coefficients are not significant according to the z tests with the respective p-values $p = 0.824$ and $p = 0.7$.

Although pubyear did not explain the between-study heterogeneity, we continue to include it as a moderator in our subsequent examples (example 3–example 6) for illustration purposes.

◁

▷ Example 3: Identical results from univariate and multivariate analyses

At this point, it may be interesting to explore when the results from a multivariate meta-regression can match the results of separate univariate meta-analyses. Theoretically, if the within-study covariances (and thus correlations) in $\mathbf{\Lambda}_j$ are equal to 0 and the between-study covariances in $\mathbf{\Sigma}$ are also equal to 0, then performing a multivariate meta-regression is equivalent to performing separate univariate meta-regressions for each outcome.

Continuing with example 2, we specify the `wsevariables(s1 s2)` and `wcorrelations(0)` options to assume there is no within-study correlation between `y1` and `y2`. We also assume that the between-study covariances are 0 by specifying an independent covariance structure for the random effects with the `covariance(independent)` suboption of the `random()` option.

```
. meta mvregress y1 y2 = pubyear, wsevariables(s1 s2) wcorrelations(0)
> random(reml, covariance(independent))

Performing EM optimization ...

Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -3.9946242
Iteration 1:  Log restricted-likelihood = -3.9656463
Iteration 2:  Log restricted-likelihood = -3.9644233
Iteration 3:  Log restricted-likelihood =  -3.964422
Iteration 4:  Log restricted-likelihood =  -3.964422

Multivariate random-effects meta-regression     Number of obs     =         10
Method: REML                                    Number of studies =          5
                                                Obs per study:
                                                              min =          2
                                                              avg =        2.0
                                                              max =          2
                                                Wald chi2(2)      =       0.25
Log restricted-likelihood = -3.964422           Prob > chi2       =     0.8837
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
y1						
pubyear	.004542	.021569	0.21	0.833	-.0377325	.0468165
_cons	.362598	.0725013	5.00	0.000	.2204981	.504698
y2						
pubyear	-.0134909	.0299534	-0.45	0.652	-.0721985	.0452167
_cons	-.3399793	.0978864	-3.47	0.001	-.5318331	-.1481256

```
Test of homogeneity: Q_M = chi2(6) = 120.10                  Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Independent:	
sd(y1)	.1406077
sd(y2)	.201787

The results for regression coefficients and variance components are identical to those from separate univariate meta-regressions in example 2. Note that the multivariate homogeneity statistic $Q_{\mathrm{M}} = 120.10$ is the sum of the univariate statistics $Q_{\mathrm{M}} = Q_{\mathrm{res}} = 11.8$ and $Q_{\mathrm{M}} = Q_{\mathrm{res}} = 108.3$, where Q_{res} is the univariate version of Q_{M} defined in [META] **meta regress**.

◁

▷ Example 4: Heterogeneity statistics

Continuing with example 2, let's refit the model and use the postestimation command `estat heterogeneity` to quantify heterogeneity after the bivariate meta-regression. Assessing the residual between-study variability is important in the context of random-effects multivariate meta-regression, so we will discuss various heterogeneity measures in detail in this example.

```
. quietly meta mvregress y1 y2 = pubyear, wcovvariables(v*)
. estat heterogeneity
Method: Cochran
Joint:
  I2 (%) = 95.23
      H2 = 20.96
Method: Jackson-White-Riley
y1:
  I2 (%) = 85.26
       R =  2.60
y2:
  I2 (%) = 95.85
       R =  4.91
Joint:
  I2 (%) = 91.57
       R =  3.44
```

By default, the Cochran and Jackson–White–Riley heterogeneity statistics are reported, but the White heterogeneity statistic is also available, as we demonstrate later in this example.

Cochran I^2_{Q} and H^2_{Q} are direct extensions to the multivariate setting of the univariate I^2 and H^2 statistics based on the DerSimonian–Laird method and thus have the same interpretations; see *Heterogeneity measures* in *Methods and formulas* in [META] **meta summarize** and *Residual heterogeneity measures* in *Methods and formulas* in [META] **meta regress**. For instance, $I^2_{\text{Q}} = 95.23\%$ means that 95.23% of the residual heterogeneity, heterogeneity not accounted for by the moderator `pubyear`, is due to true heterogeneity between the studies as opposed to the sampling variability. The high value for this statistic is not surprising because, as we saw in example 2, `pubyear` did not explain much heterogeneity between the studies.

The values of Cochran statistics are the same for all random-effects methods because they are based on the Cochran multivariate Q statistic, which is calculated based on the fixed-effects model; see *Cochran heterogeneity statistics* in *Methods and formulas* in [META] **estat heterogeneity (mv)** for details. One potential shortcoming of the Cochran statistics is that they quantify the amount of heterogeneity jointly for all outcomes. The Jackson–White–Riley statistics (Jackson, White, and Riley 2012) provide ways to assess the contribution of each outcome to the total heterogeneity, in addition to their joint contribution.

You can also investigate the impact of any subset of outcomes on heterogeneity by specifying the subset of outcomes in the `jwriley()` option of `estat heterogeneity`; see example 1 of [META] **estat heterogeneity (mv)**. These statistics are also the only truly multivariate heterogeneity statistics in the sense that their definitions stem from purely multivariate concepts rather than from univariate concepts applied to the multivariate setting.

The Jackson–White–Riley statistics measure the variability of the random-effects estimator relative to the fixed-effects estimator. The larger the values, the more between-study heterogeneity is left unexplained after accounting for moderators. The R_{JWR} statistic is an absolute measure ($R_{\text{JWR}} \geq 1$), and

I^2_{JWR} is defined based on R_{JWR} as a percentage increase in the variability of the random-effects estimates relative to the fixed-effects estimates; see *Jackson–White–Riley heterogeneity statistics* in *Methods and formulas* in [META] **estat heterogeneity (mv)** for technical details.

$R_{\text{JWR}} = 1$, and consequently $I^2_{\text{JWR}} = 0\%$, means that the moderators have accounted for all the heterogeneity between the effect sizes, and therefore there is no difference between the random-effects and fixed-effects models. Values of I^2_{JWR} that are close to 100% mean that considerable residual heterogeneity is still present in the model so that the random-effects model is more appropriate. In our example, for instance, for outcome `y1`, $R_{\text{JWR}} = 2.6$, and the corresponding $I^2_{\text{JWR}} = 85.26\% > 75\%$, which suggests "large heterogeneity" according to Higgins et al. (2003).

Other multivariate extensions of the I^2 heterogeneity statistic have also been used in practice. For example, the White I^2 statistic (White 2011) can be computed by using the `white` option.

```
. estat heterogeneity, white
Method: White
y1:
  I2 (%) = 77.26
y2:
  I2 (%) = 94.32
```

The White I^2 statistic is a direct extension of the univariate I^2 statistic (*Residual heterogeneity measures* in *Methods and formulas* in [META] **meta regress**), except the estimated between-study variance $\hat{\tau}^2$ is replaced by a diagonal of the estimated between-study covariance matrix, $\widehat{\boldsymbol{\Sigma}}$. It has the same interpretation as the univariate I^2 and reduces to it when there is only one dependent variable.

Unlike the Cochran and Jackson–White–Riley statistics that can assess heterogeneity jointly for all outcomes, the White statistic can only quantify heterogeneity separately for each outcome; see table 1 in [META] **estat heterogeneity (mv)**. In our example, continuing with outcome `y1`, we see that $I^2_{\text{W}} = 77.26\% > 75\%$ also reports the presence of a large between-study variability for that outcome even after accounting for `pubyear`.

◁

❑ Technical note

The actual definition for the Jackson–White–Riley R_{JWR} statistic is somewhat technical. It is easier to think about it first in the univariate setting, where it is defined as the ratio of the widths of the CIs of the random-effects estimator for the regression coefficient vector to the corresponding fixed-effects estimator raised to the power of $1/2p$. In the multivariate setting, the widths of confidence intervals become areas or volumes of confidence regions, and the power becomes $1/2pd$.

For example, for outcome `y1`, $d = 1$, $p = 2$, and $\hat{\beta}_{01}$ and $\hat{\beta}_{11}$ are the estimates of the constant and the regression coefficient for `pubyear`. Then, $R_{\text{JWR}} = 2.6$ is the ratio, raised to the power of $1/4$, of the areas of the confidence regions (ellipses) for estimates $\hat{\beta}_{01}$ and $\hat{\beta}_{11}$ under the random-effects and fixed-effects multivariate meta-regressions. This ratio is greater than 1 because the area of the confidence region under the random-effects model is larger.

The $I^2_{\text{JWR}} = 85.26\%$ for outcome `y1` is interpreted as roughly an 85% increase in the area of the confidence regions for the random-effects estimator of β_{01} and β_{11} relative to the fixed-effects estimator. See Jackson, White, and Riley (2012) for more ways of interpreting the I^2_{JWR} statistic in terms of generalized variances and geometric means.

Note that with three- and higher-dimensional models, the areas of confidence regions become volumes, and the shapes of confidence regions become ellipsoids.

❑

Example 5: Jackson–White–Riley random-effects method

Continuing with example 2, we demonstrate the use of an alternative random-effects estimation method, the Jackson–White–Riley method, instead of the default REML method. This method is a multivariate extension of the popular univariate DerSimonian–Laird method.

```
. meta mvregress y1 y2 = pubyear, wcovvariables(v*) random(jwriley)
Multivariate random-effects meta-regression     Number of obs     =         10
Method: Jackson-White-Riley                     Number of studies =          5
                                                Obs per study:
                                                              min =          2
                                                              avg =        2.0
                                                              max =          2
                                                Wald chi2(2)      =       0.30
                                                Prob > chi2       =     0.8621
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
y1						
pubyear	.0046544	.023268	0.20	0.841	-.04095	.0502588
_cons	.358993	.0783252	4.58	0.000	.2054784	.5125075
y2						
pubyear	-.0117463	.0419197	-0.28	0.779	-.0939074	.0704147
_cons	-.335579	.1393286	-2.41	0.016	-.608658	-.0624999

```
Test of homogeneity: Q_M = chi2(6) = 125.76                  Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Unstructured:	
sd(y1)	.1547229
sd(y2)	.2947281
corr(y1,y2)	.6518347

The estimates of the regression coefficients are very similar to those from example 2 using the REML method. For instance, the coefficient of pubyear for outcome y1 is 0.0049 and is similar to the REML estimate of 0.0047. The standard errors and estimates of variance components are larger than those obtained from the REML estimation. This is because REML assumes normality and, when this assumption is satisfied, it is likely to produce more efficient estimates than a method of moments estimator such as the Jackson–White–Riley.

◁

Example 6: Jackson–Riley standard-error adjustment

Jackson and Riley (2014) proposed a multivariate extension of the univariate Knapp and Hartung (2003) standard-error adjustment that provides more accurate inference for the regression coefficients when the number of studies is small (as is the case in our example where $K = 5$).

Continuing with example 2, we compute the Jackson–Riley standard-error adjustment by specifying the `se(jriley)` suboption within `random()`.

```
. meta mvregress y1 y2 = pubyear, wcovvariables(v*) random(reml, se(jriley))
Performing EM optimization ...

Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -3.5544446
Iteration 1:  Log restricted-likelihood = -3.5402086
Iteration 2:  Log restricted-likelihood = -3.5399568
Iteration 3:  Log restricted-likelihood = -3.5399567

Multivariate random-effects meta-regression     Number of obs     =         10
Method: REML                                    Number of studies =          5
SE adjustment: Jackson-Riley                    Obs per study:
                                                              min =          2
                                                              avg =        2.0
                                                              max =          2
                                                F(2,     6.00)    =       0.20
Log restricted-likelihood = -3.5399567          Prob > F         =     0.8249
```

	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
y1						
pubyear	.0048615	.021313	0.23	0.827	-.0472895	.0570124
_cons	.3587569	.0716413	5.01	0.002	.183457	.5340569
y2						
pubyear	-.0115367	.0292256	-0.39	0.707	-.0830492	.0599758
_cons	-.3357368	.0955846	-3.51	0.013	-.569624	-.1018496

```
Test of homogeneity: Q_M = chi2(6) = 125.76                  Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Unstructured:	
sd(y1)	.1429917
sd(y2)	.2021314
corr(y1,y2)	.561385

The regression coefficients and variance components are identical to those in example 2. But the standard errors of the regression coefficients have been adjusted; see *Jackson–Riley standard-error adjustment* in *Methods and formulas* below. The tests of the regression coefficients and the model test now use the Student's t and F distributions, respectively, instead of the default normal and χ^2 distributions.

Another standard error adjustment that is used in practice is the truncated Jackson–Riley adjustment, which may be obtained by specifying the `se(truncjriley)` suboption. The Jackson–Riley standard-error adjustment reduces to the Knapp–Hartung adjustment when there is only one dependent variable. ◁

▷ Example 7: When within-study covariances are not available

Glas et al. (2003, table 3) reported a dataset of 10 studies to investigate the sensitivity and specificity of the tumor marker telomerase to diagnose primary bladder cancer. This dataset was also analyzed by Riley et al. (2007) and White (2016). Let's describe our dataset.

```
. use https://www.stata-press.com/data/r18/telomerase
(Telomerase for diagnosing primary bladder cancer)
. describe
Contains data from https://www.stata-press.com/data/r18/telomerase.dta
 Observations:            10                  Telomerase for diagnosing
                                                primary bladder cancer
    Variables:             8                  4 Feb 2023 04:09
                                              (_dta has notes)
------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
trial           str22   %22s                  Trial label
trialnum        byte    %9.0g                 Trial ID
y1              float   %9.0g                 Logit sensitivity
y2              float   %9.0g                 Logit specificity
s1              float   %9.0g                 Standard error of logit
                                                sensitivity
s2              float   %9.0g                 Standard error of logit
                                                specificity
v1              double  %10.0g                Variance of logit sensitivity
v2              double  %10.0g                Variance of logit specificity
------------------------------------------------------------------------------
Sorted by:
```

Variables `y1` and `y2` are logit-transformed sensitivity and specificity for telomerase, and `s1` and `s2` are the corresponding standard errors.

No within-study covariances are reported for this dataset. When this occurs, one possible approach is to perform a sensitivity analysis (see example 10), where we assess the impact of different magnitudes of correlations on our bivariate meta-analysis results. In our case, sensitivity and specificity are typically measured on independent groups of individuals, so it is reasonable to assume that the within-study correlation is zero between `y1` and `y2`.

We specify the `variance` option to report variances and covariances of the random effects instead of the default standard deviations and correlations to replicate the results of Riley et al. (2007, table 3), who reported variances of the random effects.

```
. meta mvregress y*, wsevariables(s*) wcorrelation(0) variance
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -28.449202  (not concave)
Iteration 1:  Log restricted-likelihood =  -25.18825
Iteration 2:  Log restricted-likelihood = -24.713278
Iteration 3:  Log restricted-likelihood =  -24.60991
Iteration 4:  Log restricted-likelihood = -24.418125
Iteration 5:  Log restricted-likelihood = -24.415969
Iteration 6:  Log restricted-likelihood = -24.415967
Multivariate random-effects meta-analysis       Number of obs     =         20
Method: REML                                    Number of studies =         10
                                                Obs per study:
                                                              min =          2
                                                              avg =        2.0
                                                              max =          2
                                                Wald chi2(0)      =          .
Log restricted-likelihood = -24.415967          Prob > chi2       =          .
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
y1						
_cons	1.166189	.1861349	6.27	0.000	.801371	1.531006
y2						
_cons	2.057721	.5534499	3.72	0.000	.9729789	3.142462

```
Test of homogeneity: Q_M = chi2(18) = 90.87                  Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Unstructured:	
var(y1)	.2022309
var(y2)	2.583339
cov(y1,y2)	-.7227936

Our results match those reported by Riley et al. (2007). The estimated overall sensitivity for `y1` is `invlogit(1.166)` $=$ 76.24 or roughly 76%, and the estimated overall specificity for `y2` is `invlogit(2.058)` $=$ 88.68 or roughly 89%. Glas et al. (2003) noted that the sensitivity of telomerase may not be large enough for clinical use in diagnosing bladder cancer.

Had we not specified the `variance` option and reported the default standard deviations and correlations of the random-effects, we would get `corr(y1,y2)` $= -1$. We can verify this either by typing `meta mvregress` to replace the results or by using the postestimation command `estat sd`. We demonstrate the latter.

```
. estat sd
```

Random-effects parameters	Estimate
Unstructured:	
sd(y1)	.4497009
sd(y2)	1.607277
corr(y1,y2)	-.9999998

Riley et al. (2007) noted that having a between-study correlation of 1 or −1 is common in multivariate meta-analysis when the number of studies is small, especially when the within-study variances are similar to or larger than the corresponding between-study variances. This is the case in our data where, for example, the mean within-study variance for `y1` is 0.18 (for instance, type `summarize v1`), which is comparable with the estimated between-study variance `var(y1)` = 0.20. Other random-effects covariance structures should be explored to address correlations of 1 and −1; see example 1 of [META] **meta mvregress postestimation**.

◁

▷ Example 8: Missing outcome data

Fiore et al. (1996) reported a dataset of 24 studies investigating the impact of 4 intervention types to promote smoking cessation. This dataset was also analyzed by Lu and Ades (2006).

The four intervention types are (a) no contact, (b) self-help, (c) individual counseling, and (d) group counseling. The goal is to compare types (b), (c), and (d) with (a). Variables `yb`, `yc`, and `yd` represent the log odds-ratios for types (b), (c), and (d) relative to group (a). The corresponding within-study variances and covariances are reported by the six variables `vbb`, `vbc`, `vbd`, `vcc`, `vcd`, and `vdd`.

An odds ratio greater than 1 (or, equivalently, positive log odds-ratio) means that the odds of quitting smoking are larger in the corresponding group compared with the odds in type (a). This dataset is an example of multiple-treatment studies.

```
. use https://www.stata-press.com/data/r18/smokecess
(Smoking cessation interventions)
. describe y* v*
```

Variable name	Storage type	Display format	Value label	Variable label
yb	double	%9.0g		Log-odds ratio (b vs a)
yc	double	%9.0g		Log-odds ratio (c vs a)
yd	double	%9.0g		Log-odds ratio (d vs a)
vbb	double	%9.0g		Variance of yb
vbc	double	%9.0g		Covariance of yb and yc
vbd	double	%9.0g		Covariance of yb and yd
vcc	double	%9.0g		Variance of yc
vcd	double	%9.0g		Covariance of yc and yd
vdd	double	%9.0g		Variance of yd

Let's explore the missing-value structure of this dataset.

```
. misstable pattern y*, frequency
  Missing-value patterns
    (1 means complete)

                 |  Pattern
  Frequency      |  1  2  3
-----------------+---------
          1      |  1  1  1
         14      |  1  0  0
          3      |  0  0  1
          3      |  1  1  0
          1      |  0  1  0
          1      |  0  1  1
          1      |  1  0  1
-----------------+---------
         24      |
Variables are  (1) yc  (2) yd  (3) yb
```

There are 24 observations, and only 1 contains values for all 3 variables. There is only one observation when both `yd` and `yb` and both `yc` and `yb` are observed. And variables `yd` and `yb` have only six nonmissing values. So, among all variables, there are a total of $72 = 3 \times 24$ values, and only $31 = 72 - (14 \times 2 + 3 \times 2 + 3 + 2 + 1 + 1)$ of them are not missing. Given how small and sparse these data are, we can anticipate that the joint estimation of these variables will be challenging without additional, potentially strong, assumptions about the data.

In fact, if we try to run the following model, where for demonstration we use the ML method,

```
. meta mvregress yb yc yd, wcovvariables(vbb vbc vbd vcc vcd vdd) random(mle)
(output omitted)
```

we will obtain a correlation between the random effects associated with outcomes `yb` and `yd`, `corr(yb,yd)`, close to 1. This is because only 2 out of the 24 studies have observations on both of the outcomes (type `misstable pattern yb yd, frequency`), which makes the estimation of `corr(yb,yd)` unstable and inaccurate. Also, the between-study covariance structure may be overparameterized given how sparse the data are.

Note that `meta mvregress` uses all available data (all 31 nonmissing values in our example) and not just complete observations. It produces valid results under the assumption that the missing observations are missing at random.

The first model we ran assumed an unrestricted (`unstructured`) between-study covariance for `yb`, `yc`, and `yd`. Let's simplify this assumption and assume an independent covariance structure to reduce the number of estimated variance components. Also, whenever a large portion of the observations is missing, as in our example, parameter estimates tend to be less accurate. We thus specify the `cformat(%9.3f)` option to display results up to three decimal points.

```
. meta mvregress y*, wcovvariables(v*) random(mle, covariance(independent))
> cformat(%9.3f)
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log likelihood = -71.117927  (not concave)
Iteration 1:  Log likelihood =  -57.19315  (not concave)
Iteration 2:  Log likelihood = -53.591507
Iteration 3:  Log likelihood = -52.323506
Iteration 4:  Log likelihood = -52.108529
Iteration 5:  Log likelihood = -52.106793
Iteration 6:  Log likelihood = -52.106792
Multivariate random-effects meta-analysis       Number of obs     =         31
Method: ML                                      Number of studies =         24
                                                Obs per study:
                                                              min =          1
                                                              avg =        1.3
                                                              max =          3
                                                Wald chi2(0)      =          .
Log likelihood = -52.106792                     Prob > chi2       =          .
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
yb						
_cons	0.147	0.135	1.09	0.274	-0.116	0.411
yc						
_cons	0.649	0.193	3.36	0.001	0.270	1.027
yd						
_cons	0.663	0.243	2.72	0.006	0.186	1.140

```
Test of homogeneity: Q_M = chi2(28) = 204.22              Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Independent:	
sd(yb)	0.000
sd(yc)	0.694
sd(yd)	0.092

All the regression coefficient estimates are positive, which means that all interventions are better than intervention (a), although without statistical significance for outcome `yb`. Parameter `sd(yb)` is close to 0, which means that the between-study covariance may still be overparameterized. In example 9 below, we will demonstrate alternative random-effects covariance structures that further restrict the between-study covariance structure.

◁

▷ Example 9: Between-study covariance structures

Continuing with example 8, we further reduce the number of variance components to be estimated by specifying a more restrictive between-study covariance structure than `covariance(independent)`. One such structure is `identity`, where we assume that all random effects are uncorrelated and have one common variance, which is to be estimated.

```
. meta mvregress y*, wcovvariables(v*) random(mle, covariance(identity))
> cformat(%9.3f)

Performing EM optimization ...

Performing gradient-based optimization:
Iteration 0:  Log likelihood = -62.707676  (not concave)
Iteration 1:  Log likelihood = -54.538092
Iteration 2:  Log likelihood = -54.501914
Iteration 3:  Log likelihood = -54.501897
Iteration 4:  Log likelihood = -54.501897

Multivariate random-effects meta-analysis       Number of obs     =         31
Method: ML                                      Number of studies =         24
                                                Obs per study:
                                                              min =          1
                                                              avg =        1.3
                                                              max =          3
                                                Wald chi2(0)      =          .
Log likelihood = -54.501897                     Prob > chi2       =          .
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
yb						
_cons	0.367	0.317	1.16	0.247	-0.254	0.988
yc						
_cons	0.674	0.176	3.83	0.000	0.329	1.019
yd						
_cons	0.864	0.396	2.18	0.029	0.087	1.641

```
Test of homogeneity: Q_M = chi2(28) = 204.22              Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Identity:	
sd(yb yc yd)	0.580

The random-effects (or between-study) covariance structure is now labeled `Identity:`, and the common standard deviation is labeled as `sd(yb yc yd)` and is equal to 0.580. Notice how sensitive the regression coefficient estimates are to the choice of the between-study covariance structure. This phenomenon is a consequence of many missing values in the data. In this case, it is important to also explore univariate results by performing meta-analysis separately for each outcome.

We can also assume that all random effects have the same correlation and the same variance by specifying the `exchangeable` covariance structure.

```
. meta mvregress y*, wcovvariables(v*) random(mle, covariance(exchangeable))
> cformat(%9.3f)

Performing EM optimization ...

Performing gradient-based optimization:
Iteration 0:  Log likelihood = -65.135877  (not concave)
Iteration 1:  Log likelihood = -54.442271  (not concave)
Iteration 2:  Log likelihood = -53.488789
Iteration 3:  Log likelihood = -53.376427
Iteration 4:  Log likelihood =  -53.35636
Iteration 5:  Log likelihood = -53.356319
Iteration 6:  Log likelihood = -53.356319

Multivariate random-effects meta-analysis       Number of obs     =         31
Method: ML                                      Number of studies =         24
                                                Obs per study:
                                                              min =          1
                                                              avg =        1.3
                                                              max =          3
                                                Wald chi2(0)      =          .
Log likelihood = -53.356319                     Prob > chi2       =          .
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
yb						
_cons	0.413	0.296	1.40	0.162	-0.166	0.992
yc						
_cons	0.705	0.193	3.66	0.000	0.327	1.082
yd						
_cons	0.837	0.308	2.71	0.007	0.232	1.441

```
Test of homogeneity: Q_M = chi2(28) = 204.22                 Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Exchangeable:	
sd(yb yc yd)	0.672
corr(yb yc yd)	0.817

The common correlation is labeled as `corr(yb yc yd)` with an estimated value of 0.817, and the common standard deviation, `sd(yb yc yd)`, is estimated to be 0.672.

`meta mvregress` lists only the estimated variance components. If you would like to see the full between-study covariance matrix, you can use the `estat recovariance` command.

```
. estat recovariance

Between-study covariance matrix
```

	yb	yc	yd
yb	.451656		
yc	.3690338	.451656	
yd	.3690338	.3690338	.451656

To see the corresponding correlation matrix, you can specify the `correlation` option.

Example 10: Sensitivity meta-analysis

It is quite common in multivariate meta-regression to produce unstable estimates, especially when the number of observations is small relative to the number of parameters to be estimated or when a relatively large portion of the observations is missing. In this case, our goal may shift toward assessing the impact of different magnitudes of between-study variances and covariances on the estimates of regression coefficients.

Continuing with the dataset in example 8, we can investigate the effect of no correlation, moderate correlation (0.4), and high correlation (0.8) between the random-effects associated with variables yb and yc on the regression coefficients estimates. For simplicity, we will assume that the random effect associated with yd is uncorrelated with the random-effects of yb and yc and that all random-effects have unit variance (so covariances and correlations are identical). Thus, our fixed between-study covariance matrices for the three scenarios are

```
. matrix Sigma1 = (1,0,0\0,1,0\0,0,1)
. matrix Sigma2 = (1,0.4,0\0.4,1,0\0,0,1)
. matrix Sigma3 = (1,0.8,0\0.8,1,0\0,0,1)
```

We fit the first model using the correlations of 0 and store the estimation results as corr0.

```
. meta mvregress y*, wcovvariables(v*) random(mle, covariance(fixed(Sigma1)))
Multivariate random-effects meta-analysis       Number of obs     =         31
Method: User-specified Sigma = Sigma1           Number of studies =         24
                                                Obs per study:
                                                              min =          1
                                                              avg =        1.3
                                                              max =          3
                                                Wald chi2(0)      =          .
                                                Prob > chi2       =          .
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
yb						
_cons	.4293913	.502528	0.85	0.393	-.5555455	1.414328
yc						
_cons	.7629462	.2739889	2.78	0.005	.2259379	1.299955
yd						
_cons	1.028532	.5979445	1.72	0.085	-.1434175	2.200482

```
Test of homogeneity: Q_M = chi2(28) = 204.22              Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
User-specified Sigma1:	
sd(yb)	1
sd(yc)	1
sd(yd)	1
corr(yb,yc)	0
corr(yb,yd)	0
corr(yc,yd)	0

```
. estimates store corr0
```

Next, we fit the model with correlations of 0.4 and store results as corr4 and the model with correlations of 0.8 and store results as corr8. For brevity, we suppress the output from both commands.

```
. quietly meta mvregress y*, wcovvariables(v*) random(mle, covariance(fixed(Sigma2)))
. estimates store corr4
. quietly meta mvregress y*, wcovvariables(v*) random(mle, covariance(fixed(Sigma3)))
. estimates store corr8
```

We compare the estimates side by side by using `estimates table`:

```
. estimates table corr0 corr4 corr8,
> keep(yb:_cons yc:_cons yd:_cons) b(%8.3f) se(%8.3f)
```

Variable	corr0	corr4	corr8
yb			
_cons	0.429	0.472	0.566
	0.503	0.478	0.418
yc			
_cons	0.763	0.752	0.730
	0.274	0.271	0.266
yd			
_cons	1.029	1.039	1.057
	0.598	0.603	0.607

Legend: b/se

As the correlation between the random effects associated with `yb` and `yc` increases, the coefficient estimate for `yb` increases, whereas that for `yc` decreases. Also, the two estimates become more precise (have smaller standard errors) as the correlation increases. This is expected because estimation borrows information from one outcome to estimate the coefficient of the other correlated outcome. This phenomenon is referred to as "strength borrowing" in the multivariate meta-analysis literature. Notice also how the various magnitudes of correlations had little to no impact on the estimation of `yd` because of the assumption of zero correlation between the random effect of `yd` and those of `yb` and of `yc`.

◁

▷ Example 11: Fixed-effects multivariate meta-regression

Gleser and Olkin (2009) reported six studies that compare the effects of five types of exercise with a control group (no exercise) on systolic blood pressure. This dataset was also analyzed by Hartung, Knapp, and Sinha (2008). Variables `y1` to `y5` are standard mean differences between each type of exercise and the control group. Ten variables, `v11`, `v12`, ..., `v55`, define the corresponding within-study variances and covariances.

The goal of this example is to demonstrate a potential problem that you may encounter in practice when there are missing observations in the data. And we also demonstrate how to perform a fixed-effects multivariate meta-analysis.

If we run the default random-effects model, we will get the following error message:

```
. use https://www.stata-press.com/data/r18/systolicbp
(Effect of exercise on systolic blood pressure)
. meta mvregress y*, wcovvariables(v*)
cannot estimate unstructured between-study covariance
    Variables y1 and y4 have 1 jointly observed value. With recov
    unstructured, at least 2 jointly observed values are required to estimate
    the between-study covariance. You may try specifying a different recov in
    option random(), such as random(, covariance(independent)).
r(459);
```

We list the observations on variables y1 and y4:

```
. list y1 y4, sep(0) noobs
```

y1	y4
.808	.
.	1.962
.	2.568
.	.
1.171	3.159
.681	.

As the error message suggests, the estimation of the between-study covariance matrix, especially the element `cov(y1,y4)`, is not possible, because there is only one joint observation $(1.171, 3.159)$ on variables y1 and y4.

We may try a different random-effects covariance structure (see example 9 and example 10). Alternatively, we will follow Gleser and Olkin (2009) and perform a fixed-effects multivariate meta-analysis by specifying the `fixed` option.

```
. meta mvregress y*, wcovvariables(v*) fixed
Multivariate fixed-effects meta-analysis        Number of obs     =         15
                                                Number of studies =          6
                                                Obs per study:
                                                              min =          1
                                                              avg =        2.5
                                                              max =          4
                                                Wald chi2(0)      =          .
                                                Prob > chi2       =          .
```

		Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
y1							
	_cons	.7560005	.1144556	6.61	0.000	.5316716	.9803294
y2							
	_cons	1.398708	.1265397	11.05	0.000	1.150695	1.646722
y3							
	_cons	1.745014	.1646159	10.60	0.000	1.422373	2.067655
y4							
	_cons	2.146055	.1823172	11.77	0.000	1.78872	2.50339
y5							
	_cons	2.141486	.2338656	9.16	0.000	1.683118	2.599854

```
Test of homogeneity: Q_M = chi2(10) = 10.10                  Prob > Q_M = 0.4318
```

The homogeneity test based on the statistic $Q_{\mathrm{M}} = 10.1$ favors the fixed-effects model ($p = 0.4318$). However, we should be careful not to rely solely on this test because it is known to have low power when the number of studies is small (Hedges and Pigott 2001).

◁

Stored results

meta mvregress stores the following in e():

Scalars

e(N)	total number of observations on *depvars*
e(k)	number of parameters
e(k_eq)	number of dependent variables
e(k_f)	number of fixed-effects parameters
e(k_r)	number of random-effects parameters
e(k_rs)	number of variances
e(k_rc)	number of covariances
e(seadj)	standard error adjustment (se() only)
e(ll)	log (restricted) likelihood (mle and reml only)
e(rank)	rank of e(V)
e(ic)	number of iterations (mle and reml only)
e(df_m)	model degrees of freedom
e(chi2)	model χ^2 Wald test statistic
e(df_r)	model denominator degrees of freedom (tdistribution() only)
e(F)	model F statistic (tdistribution() only)
e(p)	p-value for model test
e(Q_M)	multivariate Cochran Q residual homogeneity test statistic
e(df_Q_M)	degrees of freedom for residual homogeneity test
e(p_Q_M)	p-value for residual homogeneity test
e(converged)	1 if converged, 0 otherwise (mle and reml only)
e(s_max)	maximum number of observations per study
e(s_avg)	average number of observations per study
e(s_min)	minimum number of observations per study
e(N_s)	number of studies

Macros

e(cmd)	meta mvregress
e(cmdline)	command as typed
e(model)	multivariate meta-analysis model
e(method)	multivariate meta-analysis estimation method
e(title)	title in estimation output
e(chi2type)	Wald; type of model χ^2 test
e(depvars)	names of dependent variables
e(indepvars)	names of independent variables (moderators)
e(wcovvariables)	variables defining within-study covariance matrix
e(wsevariables)	standard error variables from wsevariables()
e(wcorrelations)	values of the assumed within-study correlations from wcorrelations()
e(redim)	random-effects dimensions
e(vartypes)	variance-structure types
e(seadjtype)	type of standard error adjustment (se() only)
e(technique)	maximization technique (mle and reml only)
e(ml_method)	type of ml method
e(opt)	type of optimization (mle and reml only)
e(optmetric)	matsqrt or matlog; random-effects matrix parameterization (mle and reml only)
e(properties)	b V
e(predict)	program used to implement predict
e(estat_cmd)	program used to implement estat
e(marginsok)	predictions allowed by margins
e(marginsnotok)	predictions disallowed by margins
e(marginsdefault)	default predict() specification for margins
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices
`e(b)` coefficient vector
`e(V)` variance–covariance matrix of the estimators
Functions
`e(sample)` marks estimation sample

In addition to the above, the following is stored in `r()`:

Matrices
`r(table)` matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in `r()` are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Methods and formulas

Methods and formulas are presented under the following headings:

Fixed-effects multivariate meta-regression
Random-effects multivariate meta-regression
Iterative methods for computing $\boldsymbol{\Sigma}$
Noniterative method for computing $\boldsymbol{\Sigma}$
Random-effects covariance structures
Jackson–Riley standard-error adjustment
Multivariate meta-analysis
Residual homogeneity test

For an overview of estimation methods used by multivariate meta-regression, see van Houwelingen, Arends, and Stijnen (2002), Jackson, Riley, and White (2011), White (2011), and Sera et al. (2019).

Consider data from K independent studies and d outcomes (effect sizes). Let $\hat{\theta}_{ij}$ be the estimated effect size reported by study j for outcome i, and let the $d \times 1$ vector $\hat{\boldsymbol{\theta}}_j = (\hat{\theta}_{1j}, \hat{\theta}_{2j}, \ldots, \hat{\theta}_{dj})'$ be an estimate of the true population effect size $\boldsymbol{\theta}_j$ for study j.

Fixed-effects multivariate meta-regression

A model for the fixed-effects multivariate meta-regression (Raudenbush, Becker, and Kalaian 1988) can be expressed as

$$\hat{\theta}_{ij} = \beta_{i0} + \beta_{i1}x_{1j} + \cdots + \beta_{i,p-1}x_{p-1,j} + \epsilon_{ij} = \mathbf{x}_j\boldsymbol{\beta}_i + \epsilon_{ij}$$

for outcome $i = 1, \ldots, d$ and study $j = 1, \ldots, K$. Here $\mathbf{x}_j = (1, x_{1j}, \ldots, x_{p-1,j})$ is a $1 \times p$ vector of categorical and continuous moderators (covariates), $\boldsymbol{\beta}_i$ is an outcome-specific $p \times 1$ vector of unknown regression coefficients, and $\boldsymbol{\epsilon}_j = (\epsilon_{1j}, \epsilon_{2j}, \ldots, \epsilon_{dj})'$ is a $d \times 1$ vector of within-study errors that have a d-variate normal distribution with zero mean vector and a $d \times d$ covariance matrix $\mathrm{Var}(\boldsymbol{\epsilon}_j) = \boldsymbol{\Lambda}_j$. The within-study covariance matrices $\boldsymbol{\Lambda}_j$'s are treated as known and do not require estimation. $\boldsymbol{\Lambda}_j$'s reduce to $\hat{\sigma}_j^2$ in the case of univariate meta-analysis; see *Methods and formulas* of [META] **meta summarize**.

In matrix notation, the above fixed-effects model can be defined as

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \boldsymbol{\epsilon}_j, \quad \boldsymbol{\epsilon}_j \sim N_d\left(\mathbf{0},\, \boldsymbol{\Lambda}_j\right)$$

where $\mathbf{X}_j = \mathbf{x}_j \otimes I_d$ ($\otimes$ is the Kronecker product) is a $d \times dp$ matrix and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \ldots, \boldsymbol{\beta}_d')'$ is a $dp \times 1$ vector of all unknown regression coefficients.

Let $\mathbf{W}_j = \mathbf{\Lambda}_j^{-1}$, a $d \times d$ matrix. Then the fixed-effects estimator for the regression coefficients is

$$\widehat{\boldsymbol{\beta}} = \left(\sum_{j=1}^{K} \mathbf{X}_j' \mathbf{W}_j \mathbf{X}_j \right)^{-1} \sum_{j=1}^{K} \mathbf{X}_j' \mathbf{W}_j \hat{\boldsymbol{\theta}}_j$$

and the corresponding covariance matrix is

$$\mathrm{Var}(\widehat{\boldsymbol{\beta}}) = \left(\sum_{j=1}^{K} \mathbf{X}_j' \mathbf{W}_j \mathbf{X}_j \right)^{-1} \tag{1}$$

The above fixed-effects regression does not account for residual heterogeneity. This can lead to standard errors of regression coefficients that are too small. Next we present a random-effects multivariate meta-regression model that incorporates residual heterogeneity by including an additive between-study covariance component $\mathbf{\Sigma}$.

Random-effects multivariate meta-regression

Consider the following extension of a fixed-effects multivariate meta-regression model (Berkey et al. 1998):

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j \boldsymbol{\beta} + \boldsymbol{\epsilon}_j^*, \quad \text{where } \boldsymbol{\epsilon}_j^* \sim N_d \left(\mathbf{0}, \mathbf{\Lambda}_j + \mathbf{\Sigma} \right)$$

Alternatively, the above model can be written as

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j \boldsymbol{\beta} + \mathbf{u}_j + \boldsymbol{\epsilon}_j, \quad \boldsymbol{\epsilon}_j \sim N_d \left(\mathbf{0}, \mathbf{\Lambda}_j \right)$$

where random effects $\mathbf{u}_j = (u_{1j}, u_{2j}, \ldots, u_{dj})' \sim N_d(\mathbf{0}, \mathbf{\Sigma})$ $(j = 1, \ldots, K)$ account for the additional variation that is not explained by moderators $\mathbf{X}_j$.

The models above define a random-effects multivariate meta-regression.

Let $\widehat{\mathbf{\Sigma}}$ be an estimate of the between-study covariance matrix $\mathbf{\Sigma}$ (to be discussed later), and let $\mathbf{W}_j^* = (\widehat{\mathbf{\Sigma}} + \mathbf{\Lambda}_j)^{-1}$. The random-effects estimator for the regression coefficients is

$$\widehat{\boldsymbol{\beta}}^* = \left(\sum_{j=1}^{K} \mathbf{X}_j' \mathbf{W}_j^* \mathbf{X}_j \right)^{-1} \sum_{j=1}^{K} \mathbf{X}_j' \mathbf{W}_j^* \hat{\boldsymbol{\theta}}_j$$

The corresponding covariance matrix is given by

$$\mathrm{Var}(\widehat{\boldsymbol{\beta}}^*) = \left(\sum_{j=1}^{K} \mathbf{X}_j' \mathbf{W}_j^* \mathbf{X}_j \right)^{-1} \tag{2}$$

In the following section, we outline the estimation of the between-study covariance matrix $\mathbf{\Sigma}$ for the ML and REML iterative methods. For the noniterative Jackson–White–Riley of estimating $\mathbf{\Sigma}$, see *Noniterative method for computing* $\mathbf{\Sigma}$.

Iterative methods for computing $\mathbf{\Sigma}$

The two estimators described below do not have a closed-form solution, and an iterative algorithm is needed to estimate $\mathbf{\Sigma}$.

The joint log-likelihood function of $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}$ for a random-effects multivariate meta-regression can be expressed as

$$\ln L_{\text{ML}}\left(\boldsymbol{\beta},\, \boldsymbol{\Sigma}\right) = -\frac{1}{2}\left\{ n\ln(2\pi) + \sum_{j=1}^{K} \ln\left|\mathbf{V}_j\right| + \sum_{j=1}^{K} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\boldsymbol{\beta}\right)' \mathbf{V}_j^{-1} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\boldsymbol{\beta}\right) \right\}$$

where $\mathbf{V}_j = \boldsymbol{\Sigma} + \boldsymbol{\Lambda}_j$, $|\mathbf{V}_j|$ is the determinant of $\mathbf{V}_j$, and n is the total number of observations $\hat{\theta}_{ij}$ ($n = Kd$ when there are no missing data).

The between-study covariance $\boldsymbol{\Sigma}$ is estimated by maximizing the profile log-likelihood function obtained by treating $\boldsymbol{\beta}$ as known and plugging $\widehat{\boldsymbol{\beta}}^*$ into $\ln L_{\text{ML}}(\boldsymbol{\beta},\, \boldsymbol{\Sigma})$ in place of $\boldsymbol{\beta}$ (Pinheiro and Bates [2000, ch. 2]):

$$\ln L_{\text{ML}}\left(\boldsymbol{\Sigma}\right) = -\frac{1}{2}\left\{ n\ln(2\pi) + \sum_{j=1}^{K} \ln\left|\mathbf{V}_j\right| + \sum_{j=1}^{K} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}^*\right)' \mathbf{V}_j^{-1} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}^*\right) \right\}$$

The MLE of $\boldsymbol{\Sigma}$ does not incorporate the uncertainty about the unknown regression coefficients $\boldsymbol{\beta}$ and thus can be negatively biased.

The REML estimator of $\boldsymbol{\Sigma}$ maximizes the restricted log-likelihood function

$$\ln L_{\text{REML}}\left(\boldsymbol{\Sigma}\right) = \ln L_{\text{ML}}\left(\boldsymbol{\Sigma}\right) - \frac{1}{2}\ln\left|\sum_{j=1}^{K} \mathbf{X}_j'\mathbf{V}_j^{-1}\mathbf{X}_j\right| + \frac{dp}{2}\ln(2\pi)$$

The REML method estimates $\boldsymbol{\Sigma}$ by accounting for the uncertainty in the estimation of $\boldsymbol{\beta}$, which leads to a nearly unbiased estimate of $\boldsymbol{\Sigma}$. The optimization of the above log-likelihood functions can be done using the machinery of the mixed-effects models to obtain the estimates $\widehat{\boldsymbol{\beta}}^*$ and $\widehat{\boldsymbol{\Sigma}}$. For details, see Pinheiro and Bates (2000) and *Methods and formulas* of [ME] **mixed**. When $d = 1$, that is, in the context of univariate meta-analysis, the above ML and REML estimators reduce to their univariate counterparts as reported by `meta regress`.

Noniterative method for computing $\boldsymbol{\Sigma}$

This section describes a noniterative method to estimate the between-study covariance matrix $\boldsymbol{\Sigma}$, which has a closed-form expression. The formulas in this section are based on Jackson, White, and Riley (2013).

Using the notation for a fixed-effects multivariate meta-regression, define a $d \times d$ matrix

$$\mathbf{Q}_{\text{JWR}} = \sum_{j=1}^{K} \mathbf{W}_j \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}\right) \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}\right)' \mathbf{R}_j$$

where $\mathbf{R}_j$ is a $d \times d$ diagonal matrix with the ith diagonal element equal to 1 if $\hat{\theta}_{ij}$ is observed and 0 if it is missing.

The role of $\mathbf{R}_j$ is to ensure that missing outcomes do not contribute to the computation of $\mathbf{Q}_{\text{JWR}}$. Let $\mathbf{R} = \oplus_{j=1}^{K}\mathbf{R}_j$ and $\mathbf{W} = \oplus_{j=1}^{K}\mathbf{W}_j$ be $Kd \times Kd$ block-diagonal matrices formed by submatrices $\mathbf{R}_j$ and $\mathbf{W}_j$, respectively; $\oplus$ is the Kronecker sum. In the presence of missing outcome values, the matrix $\mathbf{W}_j = \boldsymbol{\Lambda}_j^{-1}$ is obtained by inverting the submatrix of $\boldsymbol{\Lambda}_j$ corresponding to the observed outcome values and by replacing the remaining elements with zeros.

Let $\mathbf{X}$ denote a $Kd \times p$ matrix constructed by vertically stacking the $d \times p$ matrices $\mathbf{X}_j$, that is, $\mathbf{X} = (\mathbf{X}_1', \mathbf{X}_2', \ldots, \mathbf{X}_K')'$. Define

$$\begin{aligned} \mathbf{P}_{\mathrm{M}} &= (\mathbf{I}_{Kd} - \mathbf{H})' \mathbf{W} \\ \mathbf{B} &= (\mathbf{I}_{Kd} - \mathbf{H})' \mathbf{R} \end{aligned} \tag{3}$$

where $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}$ and $\mathbf{I}_{Kd}$ is the $Kd \times Kd$ identity matrix. The subscript M in $\mathbf{P}_{\mathrm{M}}$ is used to emphasize that the $Kd \times Kd$ matrix $\mathbf{P}_{\mathrm{M}}$ generalizes the $K \times K$ matrix $\mathbf{P}$, defined by (1) in *Methods and formulas* of [META] **meta regress**, to the multivariate meta-regression setting.

Partition the $Kd \times Kd$ matrices $\mathbf{P}_{\mathrm{M}}$ and $\mathbf{B}$ into K^2 blocks of $d \times d$ matrices, and denote the jth by lth submatrix of $\mathbf{P}_{\mathrm{M}}$ by $(\mathbf{P}_{\mathrm{M}})_{jl}$ and of $\mathbf{B}$ by $(\mathbf{B})_{jl}$, respectively. The method of moments estimator proposed by Jackson, White, and Riley (2013) solves the system of d^2 estimating equations

$$\mathrm{vec}\,(\mathbf{Q}_{\mathrm{JWR}}) = \mathrm{vec}\left\{\sum_{j=1}^{K} (\mathbf{B})_{jj}\right\} + \left\{\sum_{l=1}^{K}\sum_{j=1}^{K} (\mathbf{B})_{jl}' \otimes (\mathbf{P}_{\mathrm{M}})_{lj}\right\} \mathrm{vec}(\widetilde{\boldsymbol{\Sigma}})$$

where $\mathrm{vec}(\mathbf{A})$ vectorizes $\mathbf{A}$ column by column and $\otimes$ is the Kronecker product. Solving for $\mathrm{vec}(\widetilde{\boldsymbol{\Sigma}})$ and hence $\widetilde{\boldsymbol{\Sigma}}$, we obtain the JWR estimator of the between-study covariance matrix,

$$\widehat{\boldsymbol{\Sigma}}_{\mathrm{JWR}} = \frac{\widetilde{\boldsymbol{\Sigma}} + \widetilde{\boldsymbol{\Sigma}}'}{2}$$

The estimator $\widehat{\boldsymbol{\Sigma}}_{\mathrm{JWR}}$ is symmetric but not necessarily positive semidefinite. We can obtain a positive semidefinite estimator, $\widehat{\boldsymbol{\Sigma}}^{+}_{\mathrm{JWR}}$, based on spectral decomposition $\widehat{\boldsymbol{\Sigma}}_{\mathrm{JWR}} = \sum_{i=1}^{d} \lambda_i \mathbf{e}_i \mathbf{e}_i'$ as follows,

$$\widehat{\boldsymbol{\Sigma}}^{+}_{\mathrm{JWR}} = \sum_{i=1}^{d} \max(0, \lambda_i)\, \mathbf{e}_i \mathbf{e}_i'$$

where λ_is are the eigenvalues of $\widehat{\boldsymbol{\Sigma}}_{\mathrm{JWR}}$ and $\mathbf{e}_i$s are the corresponding orthonormal eigenvectors. $\widehat{\boldsymbol{\Sigma}}^{+}_{\mathrm{JWR}}$ has the same eigenvectors as $\widehat{\boldsymbol{\Sigma}}_{\mathrm{JWR}}$ but with negative eigenvalues truncated at 0.

The JWR estimator can be viewed as an extension of the DerSimonian–Laird estimator from the random-effects meta-regression to multivariate meta-regression. For univariate meta-analysis ($d = 1$), the JWR estimator reduces to the DerSimonian–Laird estimator from `meta regress`. The truncation of $\widehat{\boldsymbol{\Sigma}}_{\mathrm{JWR}}$ to obtain $\widehat{\boldsymbol{\Sigma}}^{+}_{\mathrm{JWR}}$ is equivalent to truncating $\hat{\tau}^2_{\mathrm{DL}}$ at 0 in univariate meta-regression whenever the estimate is negative.

Random-effects covariance structures

Several covariance structures may be assumed for the between-study covariance matrix $\boldsymbol{\Sigma}$. The default covariance structure is `unstructured`, which is the most general structure in which all elements or, more precisely, $d(d+1)/2$ variance components are estimated. Other covariance structures are `independent`, `exchangeable`, `identity`, and `fixed(`*matname*`)`. These structures may be useful to provide more stable estimates by reducing the complexity of the model, especially when the number of observations, n, is relatively small.

For example, when $d = 3$, the covariance structures are

$$\texttt{unstructured} \quad \boldsymbol{\Sigma} = \begin{bmatrix} \sigma_{11} & & \\ \sigma_{21} & \sigma_{22} & \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix}$$

$$\texttt{independent} \quad \boldsymbol{\Sigma} = \begin{bmatrix} \sigma_{11} & & \\ 0 & \sigma_{22} & \\ 0 & 0 & \sigma_{33} \end{bmatrix}$$

$$\texttt{exchangeable} \quad \boldsymbol{\Sigma} = \begin{bmatrix} \sigma_{11} & & \\ \sigma_{21} & \sigma_{11} & \\ \sigma_{21} & \sigma_{21} & \sigma_{11} \end{bmatrix}$$

$$\texttt{identity} \quad \boldsymbol{\Sigma} = \begin{bmatrix} \sigma_{11} & & \\ 0 & \sigma_{11} & \\ 0 & 0 & \sigma_{11} \end{bmatrix}$$

Any of the above covariance structures may be specified with the ML and REML methods. Only the `unstructured` covariance structure is allowed with the JWR method. When covariance structure `fixed(`*matname*`)` is specified, *matname* is assumed to be the known between-study covariance, and thus no iteration is needed.

Jackson–Riley standard-error adjustment

By default, the inference about the regression coefficients and their confidence intervals from meta-regression is based on a normal distribution. The test of the significance of all regression coefficients is based on a χ^2 distribution with $d(p-1)$ degrees of freedom.

Jackson and Riley (2014) proposed an adjustment to the standard errors of the estimated regression coefficients to account for the uncertainty in the estimation of $\boldsymbol{\Sigma}$. They showed that the corresponding tests of individual regression coefficients and their confidence intervals are based on the Student's t distribution with $n - dp$ degrees of freedom and that the overall test of significance is based on an F distribution with $d(p-1)$ numerator and $n - dp$ denominator degrees of freedom.

The Jackson–Riley adjustment first calculates the quadratic form,

$$q_{\text{JR}} = \frac{1}{n - dp} \sum_{j=1}^{K} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}\right)' \mathbf{W}_j^* \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}\right)$$

It then multiplies the regular expressions of the variances of regression coefficients by q_{JR} or, in the case of the truncated Jackson–Riley adjustment, by $\max(1, q_{\text{JR}})$. When $d = 1$, the Jackson–Riley adjustment, q_{JR}, reduces to the Knapp–Hartung adjustment, q_{KH}, from *Knapp–Hartung standard-error adjustment* in *Methods and formulas* in [META] **meta regress**.

Multivariate meta-analysis

The formulas presented so far are derived for the general case of multivariate meta-regression. Methods and formulas for the special case of multivariate meta-analysis (when no moderators are included) can be obtained by taking $\mathbf{x}_j = 1$ and $p = 1$. When $d = 1$, the REML, ML, and JWR estimators reduce to the univariate REML, ML, and DL estimators described in [META] **meta summarize** for constant-only models and in [META] **meta regress** for regression models.

Residual homogeneity test

Consider a test of residual homogeneity, which mathematically translates to H_0: $\boldsymbol{\Sigma} = \mathbf{0}_{d\times d}$ for the random-effects multivariate meta-regression. This test is based on the multivariate residual weighted sum of squares, Q_{M},

$$Q_{\mathrm{M}} = \sum_{j=1}^{K} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}\right)' \mathbf{W}_j \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}\right)$$

where $\widehat{\boldsymbol{\beta}}$ is a fixed-effects estimator of regression coefficients defined for a fixed-effects multivariate meta-regression.

Under the null hypothesis of residual homogeneity, Q_{M} follows a χ^2 distribution with $n - dp$ degrees of freedom (Seber and Lee 2003, sec. 2.4). The Q_{M} statistic reduces to the univariate residual homogeneity test statistic, Q_{res}, when $d = 1$ (see *Residual homogeneity test* in *Methods and formulas* in [META] **meta regress**). It also reduces to the univariate homogeneity statistic Q when no moderators are included (see *Homogeneity test* in *Methods and formulas* in [META] **meta summarize**).

References

Antczak-Bouckoms, A., K. Joshipura, E. Burdick, and J. F. Camilla Tulloch. 1993. Meta-analysis of surgical versus non-surgical methods of treatment for periodontal disease. *Journal of Clinical Periodontology* 20: 259–268. https://doi.org/10.1111/j.1600-051X.1993.tb00355.x.

Berkey, C. S., D. C. Hoaglin, F. Mosteller, and G. A. Colditz. 1998. Meta-analysis of multiple outcomes by regression with random effects. *Statistics in Medicine* 17: 2537–2550. https://doi.org/10.1002/(SICI)1097-0258(19981130)17:22%3C2537::AID-SIM953%3E3.0.CO;2-C.

Fiore, M. C., W. C. Bailey, S. J. Cohen, S. F. Dorfman, M. G. Goldstein, E. R. Gritz, R. B. Heyman, J. Holbrook, C. R. Jaen, T. E. Kottke, H. A. Lando, R. Mecklenburg, P. D. Mullen, L. M. Nett, L. Robinson, M. L. Stitzer, A. C. Tommasello, L. Villejo, and M. E. Wewers. 1996. "Smoking Cessation". In *Clinical Practice Guideline No. 18*. Rockville, MD: U.S. Department of Health and Human Services, Public Health Service, Agency for Health Care Policy and Research. AHCPR Publication No. 96-0692.

Glas, A. S., D. Roos, M. Deutekom, A. H. Zwinderman, P. M. M. Bossuyt, and K. H. Kurth. 2003. Tumor markers in the diagnosis of primary bladder cancer. A systematic review. *Journal of Urology* 169: 1975–1982. https://doi.org/10.1097/01.ju.0000067461.30468.6d.

Gleser, L. J., and I. Olkin. 2009. "Stochastically dependent effect sizes". In *The Handbook of Research Synthesis and Meta-Analysis*, edited by H. Cooper, L. V. Hedges, and J. C. Valentine, 357–376. 2nd ed. New York: Russell Sage Foundation.

Harbord, R. M., and J. P. T. Higgins. 2016. "Meta-regression in Stata". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 85–111. 2nd ed. College Station, TX: Stata Press.

Hartung, J., G. Knapp, and B. K. Sinha. 2008. *Statistical Meta-Analysis with Applications*. Hoboken, NJ: Wiley. https://doi.org/10.1002/9780470386347.

Hedges, L. V., and T. D. Pigott. 2001. The power of statistical tests in meta-analysis. *Psychological Methods* 6: 203–217. https://doi.org/10.1037/1082-989X.6.3.203.

Higgins, J. P. T., and S. G. Thompson. 2004. Controlling the risk of spurious findings from meta-regression. *Statistics in Medicine* 23: 1663–1682. https://doi.org/10.1002/sim.1752.

Higgins, J. P. T., S. G. Thompson, J. J. Deeks, and D. G. Altman. 2003. Measuring inconsistency in meta-analyses. *BMJ* 327: 557–560. https://doi.org/10.1136/bmj.327.7414.557.

Jackson, D., and R. D. Riley. 2014. A refined method for multivariate meta-analysis and meta-regression. *Statistics in Medicine* 33: 541–554. https://doi.org/10.1002/sim.5957.

Jackson, D., R. D. Riley, and I. R. White. 2011. Multivariate meta-analysis: Potential and promise. *Statistics in Medicine* 30: 2481–2498. https://doi.org/10.1002/sim.4172.

Jackson, D., I. R. White, and R. D. Riley. 2012. Quantifying the impact of between-study heterogeneity in multivariate meta-analyses. *Statistics in Medicine* 31: 3805–3820. https://doi.org/10.1002/sim.5453.

———. 2013. A matrix-based method of moments for fitting the multivariate random effects model for meta-analysis and meta-regression. *Biometrical Journal* 55: 231–245. https://doi.org/10.1002/bimj.201200152.

Knapp, G., and J. Hartung. 2003. Improved tests for a random effects meta-regression with a single covariate. *Statistics in Medicine* 22: 2693–2710. https://doi.org/10.1002/sim.1482.

Lu, G., and A. E. Ades. 2006. Assessing evidence inconsistency in mixed treatment comparisons. *Journal of the American Statistical Association* 101: 447–459. https://doi.org/10.1198/016214505000001302.

Orsini, N. 2021. Weighted mixed-effects dose–response models for tables of correlated contrasts. *Stata Journal* 21: 320–347.

Pinheiro, J. C., and D. M. Bates. 2000. *Mixed-Effects Models in S and S-PLUS*. New York: Springer. https://doi.org/10.1007/b98882.

Raudenbush, S. W., B. J. Becker, and H. Kalaian. 1988. Modeling multivariate effect sizes. *Psychological Bulletin* 103: 111–120. https://doi.org/10.1037/0033-2909.103.1.111.

Riley, R. D., K. R. Abrams, A. J. Sutton, P. C. Lambert, and J. R. Thompson. 2007. Bivariate random-effects meta-analysis and the estimation of between-study correlation. *BMC Medical Research Methodology* 7: art. 3. https://doi.org/10.1186/1471-2288-7-3.

Rose, C. J. 2024. Multivariate random-effects meta-analysis for sparse data using smvmeta. *Stata Journal* 24: 301–328.

Rosenthal, R., and D. B. Rubin. 1986. Meta-analytic procedures for combining studies with multiple effect sizes. *Psychological Bulletin* 99: 400–406. https://doi.org/10.1037/0033-2909.99.3.400.

Seber, G. A. F., and A. J. Lee. 2003. *Linear Regression Analysis*. 2nd ed. Hoboken, NJ: Wiley. https://doi.org/10.1002/9780471722199.

Sera, F., B. Armstrong, M. Blangiardo, and A. Gasparrini. 2019. An extended mixed-effects framework for meta-analysis. *Statistics in Medicine* 38: 5429–5444. https://doi.org/10.1002/sim.8362.

Thompson, S. G., and S. J. Sharp. 1999. Explaining heterogeneity in meta-analysis: A comparison of methods. *Statistics in Medicine* 18: 2693–2708. https://doi.org/10.1002/(sici)1097-0258(19991030)18:20<2693::aid-sim235>3.0.co;2-v.

van Houwelingen, H. C., L. R. Arends, and T. Stijnen. 2002. Advanced methods in meta-analysis: Multivariate approach and meta-regression. *Statistics in Medicine* 21: 589–624. https://doi.org/10.1002/sim.1040.

White, I. R. 2011. Multivariate random-effects meta-regression: Updates to mvmeta. *Stata Journal* 11: 240–254.

———. 2016. "Multivariate random-effects meta-analysis". In *Meta-Analysis in Stata: An Updated Collection from the Stata Journal*, edited by T. M. Palmer and J. A. C. Sterne, 232–248. 2nd ed. College Station, TX: Stata Press.

Also see

[META] **meta mvregress postestimation** — Postestimation tools for meta mvregress

[META] **meta regress** — Meta-analysis regression

[META] **meta summarize** — Summarize meta-analysis data+

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

[U] 20 Estimation and postestimation commands

meta mvregress postestimation — Postestimation tools for meta mvregress

Postestimation commands | predict | margins
Remarks and examples | Methods and formulas | References
Also see

Postestimation commands

The following postestimation commands are of special interest after `meta mvregress`:

Command	Description
`estat heterogeneity`	compute multivariate heterogeneity statistics
`estat recovariance`	display the estimated random-effects covariance matrix
`estat sd`	display variance components as standard deviations and correlations

The following standard postestimation commands are also available:

Command	Description
`contrast`	contrasts and ANOVA-style joint tests of parameters
`estat ic`	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC, respectively)
`estat summarize`	summary statistics for the estimation sample
`estat vce`	variance–covariance matrix of the estimators (VCE)
`estimates`	cataloging estimation results
`etable`	table of estimation results
`lincom`	point estimates, standard errors, testing, and inference for linear combinations of parameters
`margins`	marginal means, predictive margins, marginal effects, and average marginal effects
`marginsplot`	graph the results from margins (profile plots, interaction plots, etc.)
`nlcom`	point estimates, standard errors, testing, and inference for nonlinear combinations of parameters
`predict`	predictions and their SEs, leverage statistics, etc.
`predictnl`	point estimates, standard errors, testing, and inference for generalized predictions
`pwcompare`	pairwise comparisons of parameters
`test`	Wald tests of simple and composite linear hypotheses
`testnl`	Wald tests of nonlinear hypotheses

predict

Description for predict

`predict` creates a new variable containing predictions such as linear predictions, standard errors, fitted values, residuals, and standardized residuals. After random-effects multivariate meta-regression, you can also obtain estimates of random effects and their standard errors or variances and covariances.

Menu for predict

Statistics > Postestimation

Syntax for predict

Syntax for obtaining predictions other than best linear unbiased predictions (BLUPs) of random effects

```
predict [type] newvar [if] [in] [ , statistic fixedonly depvar(depname | ##) ]
```

Syntax for obtaining BLUPs of random effects and the BLUPs' standard errors

```
predict [type] { stub* | newvarlist } [if] [in], reffects [reses(resesspec)
  revce(stub* | newvarlist) ]
```

statistic	Description
Main	
`xb`	linear prediction for the fixed portion of the model only; the default
`stdp`	standard error of the fixed-portion linear prediction
`fitted`	fitted values, fixed-portion linear prediction plus contributions based on predicted random effects
`residuals`	residuals, response minus fitted values
`rstandard`	standardized residuals

These statistics are available both in and out of sample; type `predict ... if e(sample) ...` if wanted only for the estimation sample.

Options for predict

Main

`xb`, the default, calculates the linear predictor $\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i$ for the specified dependent variable. For the random-effects multivariate meta-regression, this corresponds to the fixed portion of the linear predictor based on the estimated regression coefficients. That is, this is equivalent to fixing all random effects in the model to a theoretical mean value of 0.

`stdp` calculates the standard error of the linear predictor for the specified dependent variable.

`fitted` calculates the fitted values for the specified dependent variable. With fixed-effects multivariate meta-regression or with random-effects multivariate meta-regression when option `fixedonly` is also specified, this option is equivalent to `xb`. For random-effects multivariate meta-regression without `fixedonly`, it calculates $\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i + \widehat{u}_{ij}$, which is equal to the fixed portion of the linear predictor plus predicted random effects for the specified dependent variable.

`residuals` calculates the residuals, which are equal to the responses minus the fitted values for the specified dependent variable. With fixed-effects multivariate meta-regression or with random-effects multivariate meta-regression when option `fixedonly` is also specified, it calculates $\widehat{\theta}_{ij} - \mathbf{x}_j\widehat{\boldsymbol{\beta}}_i$. The former is known as marginal residuals in the context of the random-effects model. For random-effects multivariate meta-regression without `fixedonly`, this option calculates $\widehat{\theta}_{ij} - (\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i + \widehat{u}_{ij})$, which are known as conditional residuals.

`rstandard` calculates the standardized residuals that, for the specified dependent variable #i, are equal to the ith component of the residuals multiplied by the inverse square root of the within-study covariance matrices, $\boldsymbol{\Lambda}_j^{-1/2}\widehat{\boldsymbol{\epsilon}}_j$. With fixed-effects multivariate meta-regression or with random-effects multivariate meta-regression with `fixedonly`, $\widehat{\epsilon}_{ij} = \widehat{\theta}_{ij} - \mathbf{x}_j\widehat{\boldsymbol{\beta}}_i$. For random-effects multivariate meta-regression without `fixedonly`, $\widehat{\epsilon}_{ij} = \widehat{\theta}_{ij} - (\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i + \widehat{u}_{ij})$.

`fixedonly` specifies that all random effects be set to zero, equivalent to using only the fixed portion of the model. This option is allowed only with options `fitted`, `residuals`, and `rstandard`.

`depvar(`*depname* | `##`*)* specifies the dependent (outcome) variable of interest.

`depvar()` is filled in with one *depname* or `##` for the `xb`, `stdp`, `fitted`, `residuals`, and `rstandard` options. `depvar(#1)` would mean the calculation is to be made for the first outcome, `depvar(#2)` would mean the second, and so on. You could also refer to the outcomes by their variable names. `depvar(sensitivity)` would refer to the dependent variable named `sensitivity` and `depvar(specificity)` to the dependent variable named `specificity`.

If you do not specify `depvar()`, results are the same as if you specified `depvar(#1)`.

`reffects` calculates BLUPs of the random effects. You must specify d new variables, where d is the number of random-effects terms in the model, which is equal to the number of *depvars*. However, it is much easier to just specify *stub*`*` and let Stata name the variables *stub*`1`, *stub*`2`, …, *stub*`d` for you.

`reses(`*resesspec*`)` calculates the standard errors of the random effects; see the `reffects` option. This option may not be combined with option `revce()`. The syntax for *resesspec* is

stub`*` | *newvarlist*`[ , comparative | diagnostic ]`

`comparative`, the default, computes comparative random-effects standard errors. For linear models, these correspond to posterior standard deviations of random effects and to standard errors of marginal prediction errors $\hat{\mathbf{u}}_j - \mathbf{u}_j$. These standard errors are used for inference about the random effects.

`diagnostic` computes diagnostic random-effects standard errors. These correspond to marginal standard errors of BLUPs, $\mathrm{SE}(\hat{\mathbf{u}}_j)$. These standard errors are used for model diagnostics.

You must specify d new variables, where d is the number of random-effects terms in the model. However, it is much easier to just specify *stub*`*` and let Stata name the variables *stub*`1`, *stub*`2`, …, *stub*`d` for you. The new variables will have the same storage type as the corresponding random-effects variables.

`revce(`*stub*`*` | *newvarlist*`)` calculates the variances and covariances of the BLUPs of the random effects; see option `reffects`. This option may not be combined with option `reses()`.

You must specify $q = d(d+1)/2$ new variables, where d is the number of random-effects terms in the model. However, it is much easier to just specify *stub*`*` and let Stata name the variables *stub*`1`, *stub*`2`, …, *stub*`q` for you. The new variables will have the same storage type as the corresponding random-effects variables.

The reffects and reses() options generate d new variables at once. The random effects (or standard errors) contained in the generated variables correspond to the order in which the dependent variables, *depvars*, are specified with meta mvregress. Option revce() generates $d(d+1)/2$ variables at once. The generated variables correspond to the same order in which you specify variables in option wcovvariables() with meta mvregress. Still, examining the variable labels of the generated variables (with the describe command, for instance) can be useful in deciphering which variables correspond to which terms in the model.

margins

Description for margins

margins estimates margins of response for linear predictions.

Menu for margins

Statistics > Postestimation

Syntax for margins

margins [*marginlist*] [, *options*]

margins [*marginlist*], predict(*statistic* ...) [*options*]

statistic	Description
xb	linear predictor for the fixed portion of the model only; the default
fitted	fitted values; implies fixedonly
stdp	not allowed with margins
residuals	not allowed with margins
standard	not allowed with margins
reffects	not allowed with margins

xb and fitted default to the first outcome.

Statistics not allowed with margins are functions of stochastic quantities other than e(b).

For the full syntax, see [R] **margins**.

Remarks and examples

Various predictions and statistics are available after fitting a multivariate meta-regression using meta mvregress. For a random-effects multivariate meta-regression, calculation centers around obtaining BLUPs of the random effects. Random effects are not estimated when the model is fit but rather must be predicted after the estimation of the model parameters. The estimates of the random effects are in turn used to obtain predicted values and residuals for each dependent variable. These are useful for checking model assumptions and may be used in general as model-building tools.

▷ Example 1: Obtaining predictions of random effects

In example 7 of [META] **meta mvregress**, we conducted a multivariate meta-analysis of the logit-transformed sensitivity and specificity for the tumor marker telomerase. We refit that model, but here we assume an independent random-effects covariance structure.

```
. use https://www.stata-press.com/data/r18/telomerase
(Telomerase for diagnosing primary bladder cancer)
. meta mvregress y*, wsevariables(s*) wcorrelation(0)
> random(reml, covariance(independent))
Performing EM optimization ...
Performing gradient-based optimization:
Iteration 0:  Log restricted-likelihood = -30.469414
Iteration 1:  Log restricted-likelihood = -27.928351
Iteration 2:  Log restricted-likelihood = -27.586358
Iteration 3:  Log restricted-likelihood = -27.456381
Iteration 4:  Log restricted-likelihood = -27.456281
Iteration 5:  Log restricted-likelihood = -27.456281
Multivariate random-effects meta-analysis      Number of obs     =         20
Method: REML                                   Number of studies =         10
                                               Obs per study:
                                                             min =          2
                                                             avg =        2.0
                                                             max =          2
                                               Wald chi2(0)      =          .
Log restricted-likelihood = -27.456281         Prob > chi2       =          .
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
y1						
_cons	1.154606	.1855479	6.22	0.000	.7909387	1.518273
y2						
_cons	1.963801	.5413727	3.63	0.000	.9027297	3.024872

```
Test of homogeneity: Q_M = chi2(18) = 90.87                  Prob > Q_M = 0.0000
```

Random-effects parameters	Estimate
Independent:	
sd(y1)	.4310376
sd(y2)	1.544806

Below, we predict the random effects using `predict, reffects` and obtain their diagnostic standard errors by specifying the `reses(, diagnostic)` option. Because we have two random-effects in our model (one for each outcome), we need to specify two new variable names with `predict` and two new variable names within `reses()`. Alternatively, it is much easier to specify a stubname, say, `u*`, and `predict` will construct variables `u1` and `u2` for you. This way you do not have to worry about specifying the correct number of variables with `predict`. We will also specify a stubname, `u_se*`, within `reses()`. And we will use suboption `diagnostic` of the `reses()` option to request the diagnostic standard errors instead of the default comparative standard errors. The diagnostic standard errors are used for model diagnostics (Goldstein 2011; Skrondal and Rabe-Hesketh 2009).

```
. predict double u*, reffects reses(se_u*, diagnostic)
. list trialnum u* se_u*, mean(u1 u2) labvar(trialnum)
```

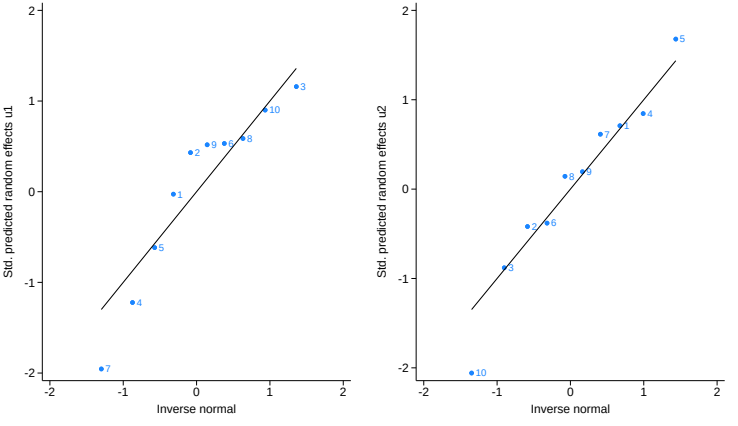
	trialnum	u1	u2	se_u1	se_u2
1.	1	-.00803546	.87413065	.29790195	1.2328506
2.	2	.10980179	-.56421535	.25481757	1.3471946
3.	3	.39364529	-1.2526626	.3395802	1.4227301
4.	4	-.36519382	1.1525847	.2988524	1.3641041
5.	5	-.20599987	2.0787496	.33418853	1.2382177
6.	6	.16425798	-.53113834	.30890465	1.3953169
7.	7	-.64318066	.67059071	.32901024	1.0915151
8.	8	.16670084	.18934479	.28423823	1.3202025
9.	9	.12138806	.26416706	.23461556	1.3593008
10.	10	.26661585	-2.8815512	.29607045	1.4000379
	Mean	-7.772e-17	-2.220e-16		

We listed the random-effects variables `u1` and `u2` with their corresponding standard error variables `se_u1` and `se_u2`. The random effects are study-specific deviations from the overall mean effect size. For example, for study 2 and outcome `y1`, the predicted logit-sensitivity is 0.1098 higher than the overall logit-sensitivity $\widehat{\theta}_1 = 1.155$. We also show the mean (average) of variables `u1` and `u2` at the bottom. Note that the means of these study-specific deviations are close to 0, which is expected because $E(u_{ij}) = 0$.

Instead of `reses()`, you may specify the `revce()` option to obtain the full variance–covariance matrix of the predicted random effects instead of only the standard errors.

Let's use the predicted random effects and their diagnostic standard errors to compute standardized random-effects variables, `ustan1` and `ustan2`, to check for outliers. We will use the `qnorm` command (see [R] **Diagnostic plots**) to obtain the normal quantile plot.

```
. generate double ustan1 = u1/se_u1
. generate double ustan2 = u2/se_u2
. label variable ustan1 "Std. predicted random effects u1"
. label variable ustan2 "Std. predicted random effects u2"
. qnorm ustan1, mlabel(trialnum) name(gr_u1) xtitle("Inverse normal")
. qnorm ustan2, mlabel(trialnum) name(gr_u2) xtitle("Inverse normal")
. graph combine gr_u1 gr_u2
```

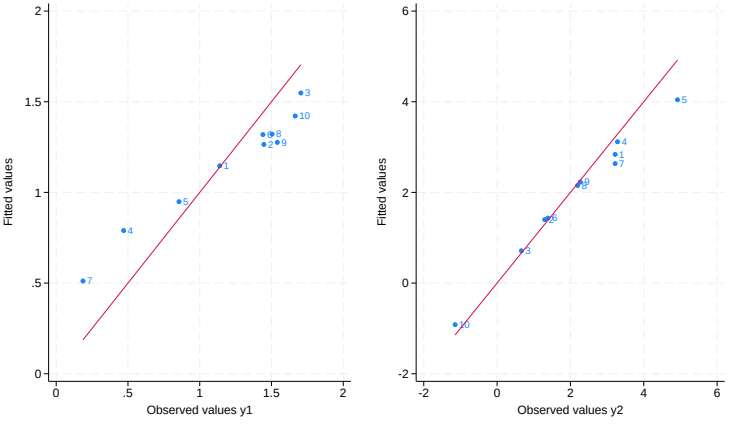


From the plots, study 7 for outcome `y1` and study 10 for outcome `y2` appear to be outliers. Study 7 has a sensitivity value of `invlogit(y1[7])` = `invlogit(.1866)` ≈ 54.65%, which is substantively lower than sensitivities of the other studies. Similarly, study 10 has a specificity value of `invlogit(y2[10])` = `invlogit(-1.145)` ≈ 24.14%.

▷ Example 2: Checking model fit

Continuing with example 1, we specify the `fitted` option to obtain the fitted values and plot them against the observed values of `y1` and `y2`. By default, for outcome i and study j, the mean effect size conditional on the estimated random effects, $\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i + \hat{u}_{ij}$, is computed for multivariate meta-regression. In our case of multivariate meta-analysis, the fitted values are $\hat{\theta}_i + \hat{u}_{ij}$. Predicted values based on only the fixed portion of the mean effect size, $\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i$ (or $\hat{\theta}_i$ in our example), may be computed by specifying the `fixedonly` option.

```
. predict double fit1, fitted depvar(#1)
. predict double fit2, fitted depvar(#2)
. twoway (scatter fit1 y1, mlabel(trialnum))
>        (function y = x, range(y1)),
> name(graph1) legend(off) xtitle(Observed values y1) ytitle(Fitted values)
. twoway (scatter fit2 y2, mlabel(trialnum))
>        (function y = x, range(y2)),
> name(graph2) legend(off) xtitle(Observed values y2) ytitle(Fitted values)
. graph combine graph1 graph2
```



In the above code, we computed the fitted values for each dependent variable using `predict, fitted depvar()`. We then produced scatterplots of the fitted values versus the observed values of `y1` and `y2`, respectively. We added a reference line $y = x$ to assess goodness of fit. Studies that are close to the reference line have their fitted values close to the observed values. Overall, it seems that goodness of fit is better for outcome `y2`.

You could also use the `rstandard` option with `predict` to compute standardized residuals. In theory, the standardized residuals are useful for checking the normality assumption. But because the number of studies is often small (for instance, only 10 in our example), standardized residuals are typically of limited use in practice with multivariate meta-regression.

◁

Methods and formulas

Methods and formulas are presented under the following headings:

Random-effects multivariate meta-regression
Fixed-effects multivariate meta-regression

The following formulas are used by `predict`. The notation is based on *Methods and formulas* of [META] **meta mvregress**.

Random-effects multivariate meta-regression

Consider a random-effects multivariate meta-regression

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \mathbf{u}_j + \boldsymbol{\epsilon}_j, \quad \boldsymbol{\epsilon}_j \sim N_d\left(\mathbf{0}, \boldsymbol{\Lambda}_j\right)$$

where $\mathbf{u}_j = (u_{1j}, u_{2j}, \ldots, u_{dj})' \sim N_d(\mathbf{0}, \boldsymbol{\Sigma})$ $(j = 1, \ldots, K)$ define random effects, $\mathbf{x}_j = (1, x_{1j}, \ldots, x_{p-1,j})$ is a $1 \times p$ vector of moderators (covariates), $\mathbf{X}_j = \mathbf{x}_j \otimes I_d$ is a $d \times dp$ design matrix, and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \ldots, \boldsymbol{\beta}_d')'$ is a $dp \times 1$ vector of unknown regression coefficients.

The random-effects estimator for regression coefficients is

$$\widehat{\boldsymbol{\beta}}^* = \left(\sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j^*\mathbf{X}_j\right)^{-1} \sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j^*\hat{\boldsymbol{\theta}}_j$$

where $\mathbf{W}_j^* = (\widehat{\boldsymbol{\Sigma}} + \boldsymbol{\Lambda}_j)^{-1}$.

The formulas below apply to outcome i, as specified in the `depvar()` option, for $i = 1, \ldots, d$ and study j for $j = 1, \ldots, K$.

The fixed portion of the linear predictor (option `xb`) is $\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i^*$.

The estimated standard error of the fixed portion of the linear predictor (option `stdp`) is

$$\widehat{\mathrm{SE}}\left(\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i^*\right) = \sqrt{\left\{\mathbf{X}_j\left(\sum_{l=1}^{K}\mathbf{X}_l'\mathbf{W}_l^*\mathbf{X}_l\right)^{-1}\mathbf{X}_j'\right\}_{ii}}$$

where $\mathbf{A}_{ii}$ denotes the ith diagonal of matrix $\mathbf{A}$.

The BLUP of a $d \times 1$ random-effects vector $\mathbf{u}_j$ (option `reffects`) is

$$\hat{\mathbf{u}}_j = \widehat{\boldsymbol{\Sigma}}\mathbf{W}_j^*\left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}^*\right)$$

When the `reses()` option is specified with `reffects`, the estimated covariance matrix of $\hat{\mathbf{u}}_j - \mathbf{u}_j$ is computed:

$$\widehat{\mathrm{Var}}\left(\hat{\mathbf{u}}_j - \mathbf{u}_j\right) = \widehat{\boldsymbol{\Sigma}} - \widehat{\boldsymbol{\Sigma}}\mathbf{W}_j^*\left\{\left(\mathbf{W}_j^*\right)^{-1} - \mathbf{X}_j\mathrm{Var}(\widehat{\boldsymbol{\beta}}^*)\mathbf{X}_j'\right\}\mathbf{W}_j^*\widehat{\boldsymbol{\Sigma}}$$

The comparative standard errors of the random effects can be obtained by taking the square root of the diagonal elements of $\widehat{\mathrm{Var}}(\hat{\mathbf{u}}_j - \mathbf{u}_j)$.

If the `diagnostic` suboption is specified within `reses()`, then the estimated covariance matrix of $\hat{\mathbf{u}}_j$ is computed:

$$\widehat{\mathrm{Var}}\left(\hat{\mathbf{u}}_j\right) = \widehat{\boldsymbol{\Sigma}}\mathbf{W}_j^*\left\{\left(\mathbf{W}_j^*\right)^{-1} - \mathbf{X}_j\mathrm{Var}(\widehat{\boldsymbol{\beta}}^*)\mathbf{X}_j'\right\}\mathbf{W}_j^*\widehat{\boldsymbol{\Sigma}}$$

The diagnostic standard errors of the random effects can be obtained by taking the square root of the diagonal elements of $\widehat{\mathrm{Var}}\left(\hat{\mathbf{u}}_j\right)$.

See Goldstein (2011), Skrondal and Rabe-Hesketh (2009), and Rabe-Hesketh and Skrondal (2022) for more details.

The `revce()` option generates variables corresponding to the variances and covariances in the $\widehat{\mathrm{Var}}(\hat{\mathbf{u}}_j)$ matrix.

The fitted value (option `fitted`) is

$$\tilde{\theta}_{ij} = \mathbf{x}_j\widehat{\boldsymbol{\beta}}_i^* + \hat{u}_{ij}$$

The residual (option `residuals`) is

$$\hat{\epsilon}_{ij} = \hat{\theta}_{ij} - \tilde{\theta}_{ij}$$

The standardized residual (option `rstandard`) is the ith element of the $d \times 1$ vector $\boldsymbol{\Lambda}_j^{-1/2}\,\hat{\boldsymbol{\epsilon}}_j$,

$$\tilde{\epsilon}_{ij} = \left(\boldsymbol{\Lambda}_j^{-1/2}\,\hat{\boldsymbol{\epsilon}}_j\right)_i$$

where $\hat{\boldsymbol{\epsilon}}_j = (\hat{\epsilon}_{1j}, \hat{\epsilon}_{2j}, \ldots, \hat{\epsilon}_{dj})'$.

When the `fixedonly` option is specified, the formulas for the fitted values, residuals, and standardized residuals are adjusted by replacing the value of $\hat{u}_{ij}$ with 0.

Fixed-effects multivariate meta-regression

Consider a fixed-effects multivariate meta-regression

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \boldsymbol{\epsilon}_j, \quad \boldsymbol{\epsilon}_j \sim N_d(\mathbf{0},\, \boldsymbol{\Lambda}_j)$$

with a $d \times dp$ design matrix $\mathbf{X}_j = \mathbf{x}_j \otimes I_d$ and a $dp \times 1$ vector $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \ldots, \boldsymbol{\beta}_d')'$ of all unknown regression coefficients.

Let $\mathbf{W}_j = \boldsymbol{\Lambda}_j^{-1}$. The fixed-effects estimator for regression coefficients is

$$\widehat{\boldsymbol{\beta}} = \left(\sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j\mathbf{X}_j\right)^{-1} \sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j\hat{\boldsymbol{\theta}}_j$$

The linear predictor (option `xb`) is $\mathbf{x}_j\widehat{\boldsymbol{\beta}}$.

The estimated standard error of the linear predictor (option `stdp`) is

$$\widehat{\mathrm{SE}}\left(\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i\right) = \sqrt{\mathbf{X}_j\left(\sum_{l=1}^{K} \mathbf{X}_l'\mathbf{W}_l\mathbf{X}_l\right)^{-1}\mathbf{X}_j'}$$

The fitted value (option `fitted`) is the same as the linear predictor:

$$\tilde{\theta}_{ij} = \mathbf{x}_j\widehat{\boldsymbol{\beta}}_i$$

The residual (option `residuals`) is

$$\hat{\epsilon}_{ij} = \hat{\theta}_{ij} - \tilde{\theta}_{ij}$$

The standardized residual (option `rstandard`) is the ith element of a $d \times 1$ vector $\boldsymbol{\Lambda}_j^{-1/2}\,\hat{\boldsymbol{\epsilon}}_j$,

$$\tilde{\epsilon}_{ij} = \left(\boldsymbol{\Lambda}_j^{-1/2}\,\hat{\boldsymbol{\epsilon}}_j\right)_i$$

References

Goldstein, H. 2011. *Multilevel Statistical Models*. 4th ed. Chichester, UK: Wiley. https://doi.org/10.1002/9780470973394.

Rabe-Hesketh, S., and A. Skrondal. 2022. *Multilevel and Longitudinal Modeling Using Stata*. 4th ed. College Station, TX: Stata Press.

Skrondal, A., and S. Rabe-Hesketh. 2009. Prediction in multilevel generalized linear models. *Journal of the Royal Statistical Society*, A ser., 172: 659–687. https://doi.org/10.1111/j.1467-985X.2009.00587.x.

Also see

[META] **meta mvregress** — Multivariate meta-regression

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

[U] 20 Estimation and postestimation commands

estat group — Summarize the composition of the nested groups

Description Menu for estat Syntax Remarks and examples
Also see

Description

`estat group` reports the number of groups and minimum, average, and maximum group sizes for each level of the model. Model levels are identified by the corresponding group variable in the data. Because groups are treated as nested, the information in this summary may differ from what you would get if you used the `tabulate` command on each group variable individually. `estat group` is available only after commands `meta meregress` and `meta multilevel`.

Menu for estat

Statistics > Postestimation

Syntax

```
estat group
```

Remarks and examples

See example 1 of [META] **meta me postestimation**.

Also see

[META] **meta me postestimation** — Postestimation tools for multilevel mixed-effects meta-analysis

[META] **meta meregress** — Multilevel mixed-effects meta-regression

[META] **meta multilevel** — Multilevel random-intercepts meta-regression

[U] 20 Estimation and postestimation commands

estat heterogeneity (me) — Compute multilevel heterogeneity statistics

Description | Menu for estat | Syntax | Option
Remarks and examples | Stored results | Methods and formulas | References
Also see

Description

`estat heterogeneity` computes multilevel heterogeneity statistics after multilevel meta-regression fit by `meta multilevel` or `meta meregress`.

Menu for estat

Statistics > Postestimation

Syntax

`estat heterogeneity [ , relevel(`*levelvar*`) ]`

Option

Main

`relevel(`*levelvar*`)` displays the amount of the total heterogeneity that can be attributed to a specific level of hierarchy defined by *levelvar*. This option affects the multilevel Higgins–Thompson I^2 statistic.

Remarks and examples

For multilevel meta-regression models, `estat heterogeneity` reports Cochran's statistic and the Higgins–Thompson I^2 statistic. Cochran's I^2_{Q} is available for all models fit by `meta meregress` or `meta multilevel`, whereas the Higgins–Thompson I^2_{HT} and $I^2_{\text{HT},\,l}$ statistics are available after `meta multilevel` and random-intercepts multilevel models (models with no random slopes) fit by `meta meregress`.

The Cochran I^2_{Q} is based on Q_{M}, a multilevel extension of the Cochran statistic Q_{res} defined in (3) in *Methods and formulas* of [META] **meta regress**. Cochran's I^2_{Q} does not depend on the random-effects component of the multilevel meta-regression model; therefore, its value does not change for models with the same fixed-effects component. Because I^2_{Q} does not depend on the random-effects component of the model, it may be computed for all multilevel meta-regression models (with or without random slopes). It quantifies the heterogeneity among the effect sizes jointly for all levels of hierarchy and has a similar interpretation to the I^2 statistic reported in standard meta-analysis or standard meta-regression. This statistic reduces to the I^2_{res} [see (4) in *Methods and formulas* of [META] **meta regress**] based on the DerSimonian–Laird random-effects method in the standard meta-analysis setting.

The multilevel Higgins–Thompson I^2_{HT} and $I^2_{\text{HT},\,l}$ (Nakagawa and Santos 2012; Cheung 2014) statistics are defined for multilevel models with random intercepts only (without random slopes). These statistics can be computed separately for each level of hierarchy and jointly for all levels of hierarchy in the model. Therefore, they quantify the contribution of each level of hierarchy to the total heterogeneity, in addition to their joint contribution.

See example 3 of [META] **meta meregress** for assessing heterogeneity in multilevel meta-analysis models.

Stored results

`estat heterogeneity` stores the following in `r()`:

Scalars

`r(I2_Q)`	Cochran I^2_{Q} heterogeneity statistic

Matrices

`r(I2_HT)`	Higgins–Thompson I^2_{HT} and $I^2_{\text{HT},\,l}$ heterogeneity statistics

Methods and formulas

Methods and formulas are presented under the following headings:

Multilevel heterogeneity statistics
Cochran heterogeneity statistic
Higgins–Thompson heterogeneity statistics

Multilevel heterogeneity statistics

See *Methods and formulas* of [META] **meta meregress**. We will discuss two types of multilevel I^2 statistics: Cochran's I^2_{Q} and the Higgins–Thompson $I^2_{\text{HT},\,l}$ and I^2_{HT}. Cochran's I^2_{Q} does not depend on the random-effects component of the multilevel meta-regression model, whereas the Higgins–Thompson $I^2_{\text{HT},\,l}$ and I^2_{HT} are defined for multilevel models with random intercepts only (with no random slopes). Therefore, we will use the random-intercepts three-level meta-regression to illustrate their computations.

Consider the three-level random-intercepts meta-regression model

$$\hat{\theta}_{jkr} = \mathbf{x}_{jkr}\boldsymbol{\beta} + u^{(3)}_{j} + u^{(2)}_{jk} + \epsilon_{jkr}$$

where $u^{(3)}_{j} \sim N(0,\, \tau^2_3)$ and $u^{(2)}_{jk} \sim N(0,\, \tau^2_2)$. Let $\mathbf{X}_j$, $\hat{\boldsymbol{\theta}}_j$, and $\boldsymbol{\epsilon}_j$ be defined as in *Methods and formulas* of [META] **meta meregress**, and then the model can be written in matrix form as

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \dot{\mathbf{Z}}_j\dot{\mathbf{u}}_j + \boldsymbol{\epsilon}_j, \quad j = 1, 2, \ldots, M$$

where $m_{j.} \times (m_j + 1)$ matrix $\dot{\mathbf{Z}}_j = (\mathbf{1}_{m_{j.}}, \oplus_{k=1}^{m_j}\mathbf{1}_{m_{jk}})$ and $(m_j + 1) \times 1$ vector of random effects $\dot{\mathbf{u}}_j = (u^{(3)}_{j}, u^{(2)}_{j1}, u^{(2)}_{j2}, \ldots, u^{(2)}_{jm_j})'$.

Cochran heterogeneity statistic

The multilevel Q_{M} is defined as

$$Q_{\text{M}} = \sum_{j=1}^{M} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}_f\right)' \boldsymbol{\Lambda}_j^{-1} \left(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}}_f\right)$$

where $\widehat{\boldsymbol{\beta}}_f$ is a fixed-effects estimator obtained by fitting a standard fixed-effects meta-regression (see [META] **meta regress**) of the $\hat{\theta}_{jk}$'s on the moderators defining the $\mathbf{X}_j$ matrix. Cochran's I^2_{Q} is defined as

$$I^2_{\text{Q}} = 100 \times \max\left(0,\, \frac{Q_{\text{M}} - n + p}{Q_{\text{M}}}\right)$$

One drawback of the Cochran statistic is that it assesses only the impact of heterogeneity jointly for all levels of hierarchy. It may be of interest to separately investigate the extent of each level of hierarchy's contribution to the total variability. This will be possible with the heterogeneity statistics discussed below.

Higgins–Thompson heterogeneity statistics

For a level l ($l = 2$ and $l = 3$ here), the Higgins–Thompson I^2_{HT} statistics are defined as

$$I^2_{\text{HT},\,l} = \frac{\hat{\tau}^2_l}{\hat{\tau}^2_2 + \hat{\tau}^2_3 + s^2_{\text{HT}}}, \quad l = 2, 3$$

$$I^2_{\text{HT}} = \frac{\hat{\tau}^2_2 + \hat{\tau}^2_3}{\hat{\tau}^2_2 + \hat{\tau}^2_3 + s^2_{\text{HT}}}$$

where $\hat{\tau}^2_l$ is the estimated variance of the random intercepts at level $l = 2, 3$ and s^2_{HT} is defined below. When option `relevel(`*levelvar*`)` is specified, then only $I^2_{\text{HT},\,l}$ is reported by `estat heterogeneity`, where l corresponds to the level identified by variable *levelvar*.

Let $X = (\mathbf{X}'_1, \mathbf{X}'_2, \ldots, \mathbf{X}_M)'$ and $\mathbf{\Lambda} = \oplus_{j=1}^{M} \mathbf{\Lambda}_j$. The level-1 variance s^2_{HT} is computed similarly to s^2 in *Residual heterogeneity measures* in *Methods and formulas* in [META] **meta regress** and is given by

$$s^2_{\text{HT}} = \frac{n - p}{\text{tr}(\mathbf{P})}$$

where $n = \sum_{j=1}^{M} m_j$ is the total number of observations and
$\mathbf{P} = \mathbf{\Lambda}^{-1} - \mathbf{\Lambda}^{-1}\mathbf{X}\left(\mathbf{X}'\mathbf{\Lambda}^{-1}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{\Lambda}^{-1}$.

The $I^2_{\text{HT},\,l}$ and I^2_{HT} statistics reduce to I^2_{res} reported by (4) in *Methods and formulas* of [META] **meta regress** when there are two levels of hierarchy in the model.

References

Cheung, M. W.-L. 2014. Modeling dependent effect sizes with three-level meta-analyses: A structural equation modeling approach. *Psychological Methods* 19: 211–229. https://doi.org/10.1037/a0032968.

Nakagawa, S., and E. S. A. Santos. 2012. Methodological issues and advances in biological meta-analysis. *Evolutionary Ecology* 26: 1253–1274. https://doi.org/10.1007/s10682-012-9555-5.

Also see

[META] **meta me postestimation** — Postestimation tools for multilevel mixed-effects meta-analysis

[META] **meta meregress** — Multilevel mixed-effects meta-regression

[META] **meta multilevel** — Multilevel random-intercepts meta-regression

[U] **20 Estimation and postestimation commands**

estat heterogeneity (mv) — Compute multivariate heterogeneity statistics

Description | Menu for estat | Syntax | Options
Remarks and examples | Stored results | Methods and formulas | References
Also see

Description

`estat heterogeneity` computes multivariate heterogeneity statistics after a random-effects multivariate meta-regression fit by `meta mvregress`. By default, the multivariate Cochran and Jackson–White–Riley heterogeneity statistics are computed, but the White statistic is also available.

Menu for estat

Statistics > Postestimation

Syntax

`estat het`erogeneity [, *statistics*]

statistics	Description
Main	
`coch`ran	Cochran statistics; the default
`jwri`ley[(*depnames*)]	Jackson–White–Riley statistics; the default for all dependent variables
`white`	White statistics
`all`	all heterogeneity statistics

You may specify one or multiple statistics simultaneously.

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

Options

Main

`cochran`, the default, specifies that the multivariate Cochran I^2_Q and H^2_Q heterogeneity statistics be computed. These statistics are computed jointly for all dependent variables and thus are not available for each dependent variable separately. These statistics are displayed by default.

`jwriley`, the default, and `jwriley(`*depnames*`)` specify that Jackson–White–Riley heterogeneity statistics I^2_{JWR} and R_{JWR} be computed. These statistics are available jointly for all dependent variables, separately for each dependent variable, or jointly for any subset of dependent variables. The statistics for all dependent variables, separately and jointly, are displayed by default.

`jwriley` computes the Jackson–White–Riley heterogeneity statistics jointly for all dependent variables and separately for each dependent variable. For example, if you have three dependent variables `y1`, `y2`, and `y3`, then the Jackson–White–Riley heterogeneity statistics are computed separately for each variable and jointly for all variables, resulting in four I^2_{JWR} statistics and four R_{JWR} statistics.

`jwriley(`*depnames*`)` computes the Jackson–White–Riley statistics jointly for the specified dependent variables *depnames*. For example, if you have three dependent variables `y1`, `y2`, and `y3`, you may specify `jwriley(y1 y2)` to compute heterogeneity statistics based on dependent variables `y1` and `y2` jointly.

`white` specifies that White I^2 statistics be computed. These statistics are available only separately for each dependent variable. With one dependent variable, the White I^2 statistic reduces to the univariate I^2 statistic as reported by [META] **meta summarize** or [META] **meta regress**.

`all` specifies that all heterogeneity statistics be reported. This option implies `cochran`, `jwriley`, and `white`.

Remarks and examples

▷ Example 1: Jackson–White–Riley heterogeneity statistics

Consider the dataset from example 8 of [META] **meta mvregress**. We fit a random-effects multivariate meta-regression using maximum likelihood estimation and an `exchangeable` covariance structure for the between-study covariance matrix:

```
. use https://www.stata-press.com/data/r18/smokecess
(Smoking cessation interventions)
. quietly meta mvregress y*, wcovvariables(v*)
> random(mle, covariance(exchangeable))
```

We can then compute the multivariate Jackson–White–Riley heterogeneity statistics by typing

```
. estat heterogeneity, jwriley
Method: Jackson-White-Riley
yb:
  I2 (%) = 81.83
       R =  2.35
yc:
  I2 (%) = 90.60
       R =  3.26
yd:
  I2 (%) = 62.84
       R =  1.64
Joint:
  I2 (%) = 78.61
       R =  2.16
```

The `jwriley` option displays the Jackson–White–Riley heterogeneity statistics separately for each dependent variable and jointly for all dependent variables. See example 4 of [META] **meta mvregress** for the interpretation of these statistics.

One feature that is unique to the Jackson–White–Riley heterogeneity statistics is the possibility of assessing heterogeneity jointly for a subset of dependent variables. Below, we specify the `jwriley(yb yc)` option to assess heterogeneity jointly for these two dependent variables.

```
. estat heterogeneity, jwriley(yb yc)
Method: Jackson-White-Riley
yb yc:
  I2 (%) = 86.02
       R =  2.67
```

The value of I^2 is 86.02%, which suggests that there is considerable heterogeneity among the effect sizes corresponding to these variables; see example 4 of [META] **meta mvregress** for details about the interpretation of I^2_{JWR} and R_{JWR}.

◁

Stored results

`estat heterogeneity` stores the following in `r()`:

Scalars

`r(I2_Q)`	Cochran I^2 heterogeneity statistic (option `cochran`)
`r(H2)`	H^2 heterogeneity statistic (option `cochran`)

Matrices

`r(R)`	Jackson–White–Riley R heterogeneity statistics (option `jwriley`)
`r(I2_JWR)`	Jackson–White–Riley I^2 heterogeneity statistics (option `jwriley`)
`r(I2_W)`	White I^2 heterogeneity statistics (option `white`)

Methods and formulas

Methods and formulas are presented under the following headings:

Brief overview of heterogeneity statistics
Cochran heterogeneity statistics
Jackson–White–Riley heterogeneity statistics
White heterogeneity statistics

See *Methods and formulas* of [META] **meta mvregress**.

Consider a random-effects multivariate meta-regression

$$\hat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \mathbf{u}_j + \boldsymbol{\epsilon}_j, \quad \boldsymbol{\epsilon}_j \sim N_d\left(\mathbf{0}, \boldsymbol{\Lambda}_j\right)$$

where $\mathbf{u}_j = (u_{1j}, u_{2j}, \ldots, u_{dj})' \sim N_d(\mathbf{0}, \boldsymbol{\Sigma})$ $(j = 1, \ldots, K)$ define random effects, $\mathbf{x}_j = (1, x_{1j}, \ldots, x_{p-1,j})$ is a $1 \times p$ vector of moderators (covariates), $\mathbf{X}_j = \mathbf{x}_j \otimes I_d$ is a $d \times dp$ design matrix, and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \ldots, \boldsymbol{\beta}_d')'$ is a $dp \times 1$ vector of unknown regression coefficients.

The random-effects estimator for regression coefficients is

$$\widehat{\boldsymbol{\beta}}^* = \left(\sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j^*\mathbf{X}_j\right)^{-1} \sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j^*\hat{\boldsymbol{\theta}}_j$$

where $\mathbf{W}_j^* = (\widehat{\boldsymbol{\Sigma}} + \boldsymbol{\Lambda}_j)^{-1}$. The corresponding covariance matrix is given by

$$\text{Var}(\widehat{\boldsymbol{\beta}}^*) = \left(\sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j^*\mathbf{X}_j\right)^{-1} \tag{1}$$

Let $\mathbf{W}_j = \mathbf{\Lambda}_j^{-1}$. The fixed-effects estimator for regression coefficients is

$$\widehat{\boldsymbol{\beta}} = \left(\sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j\mathbf{X}_j\right)^{-1} \sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j\hat{\boldsymbol{\theta}}_j$$

and the corresponding covariance matrix is

$$\mathrm{Var}(\widehat{\boldsymbol{\beta}}) = \left(\sum_{j=1}^{K} \mathbf{X}_j'\mathbf{W}_j\mathbf{X}_j\right)^{-1} \tag{2}$$

Brief overview of heterogeneity statistics

`estat heterogeneity` supports the following heterogeneity statistics: Cochran I^2_{Q} and H^2_{Q}, Jackson–White–Riley R_{JWR} and I^2_{JWR}, and White I^2_{W}. Some statistics can be computed separately for each outcome, some can be computed jointly for all outcomes, and some can be computed both jointly and separately. We summarize how each statistic can quantify heterogeneity in table 1, which is a modified version of Jackson, White, and Riley (2012, table V).

Table 1. Quantifying heterogeneity for various heterogeneity statistics

	Quantifies heterogeneity for		
Statistic	each outcome	all outcomes jointly	subsets of outcomes
I^2_{Q}, H^2_{Q}	No	Yes	No
I^2_{JWR}, R_{JWR}	Yes	Yes	Yes
I^2_{W}	Yes	No	No

Cochran heterogeneity statistics

The Cochran I^2_{Q} and H^2_{Q} are based on Q_{M}, a multivariate extension of the Cochran statistic Q_{res} defined in (3) in *Methods and formulas* of [META] **meta regress**. These statistics reduce to the I^2_{res} and H^2_{res} [see (4) and (5) in *Methods and formulas* of [META] **meta regress**] based on the DerSimonian–Laird random-effects method in the univariate case.

$$I^2_{\mathrm{Q}} = 100 \times \max\left(0,\ \frac{Q_{\mathrm{M}} - N + dp}{Q_{\mathrm{M}}}\right)$$

$$H^2_{\mathrm{Q}} = \max\left(1,\ \frac{Q_{\mathrm{M}}}{N - dp}\right)$$

where $Q_{\mathrm{M}} = \sum_{j=1}^{K} (\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}})'\mathbf{W}_j(\hat{\boldsymbol{\theta}}_j - \mathbf{X}_j\widehat{\boldsymbol{\beta}})$.

One drawback of the Cochran statistics is that they assess only the impact of heterogeneity jointly for all outcomes. It may be of interest to investigate separately the extent of each outcome's contribution to the total variability. This will be possible with all heterogeneity statistics discussed below.

Jackson–White–Riley heterogeneity statistics

Jackson, White, and Riley (2012) proposed to extend the heterogeneity statistic R of Higgins and Thompson (2002) (not to be confused with the R^2 statistic reported after [META] **meta regress**) to the multivariate setting. They then used the R statistic to construct an I^2 statistic that can quantify heterogeneity more conveniently. These I^2 and R statistics have an advantage over other versions of heterogeneity statistics because they may be computed for any subset of outcome variables.

For all outcomes jointly, the statistic R is defined as the ratio of the volumes of the confidence regions for the regression coefficients under the random-effects and fixed-effects models raised to the power $1/dp$,

$$R_{\text{JWR}} = \left(\frac{\text{Vol}_{\text{RE}}}{\text{Vol}_{\text{FE}}}\right)^{1/dp}$$

where Vol_{RE} and Vol_{FE} are the volumes of the confidence regions under the random-effects and fixed-effects models, respectively. It can be shown (see Jackson, White, and Riley [2012, appendix]) that the volumes of these regions are proportional to the square root of the determinants of the covariance matrices defined in (1) and (2),

$$\text{Vol}_{\text{RE}} \propto \left|\text{Var}(\widehat{\boldsymbol{\beta}}^{*})\right|^{1/2} \text{ and } \text{Vol}_{\text{FE}} \propto \left|\text{Var}(\widehat{\boldsymbol{\beta}})\right|^{1/2}$$

In what follows, we will define the JWR R and I^2 statistics in terms of these determinants.

For all outcome jointly, the JWR R and I^2 statistics are given by

$$R_{\text{JWR}} = \left\{\frac{\left|\text{Var}(\widehat{\boldsymbol{\beta}}^{*})\right|}{\left|\text{Var}(\widehat{\boldsymbol{\beta}})\right|}\right\}^{1/2dp} = \left\{\left|\text{Var}(\widehat{\boldsymbol{\beta}}^{*})\right| \left|\text{Var}(\widehat{\boldsymbol{\beta}})\right|^{-1}\right\}^{1/2dp}$$

$$I^2_{\text{JWR}} = 100 \times \max\left(0, \frac{\left|\text{Var}(\widehat{\boldsymbol{\beta}}^{*})\right|^{1/dp} - \left|\text{Var}(\widehat{\boldsymbol{\beta}})\right|^{1/dp}}{\left|\text{Var}(\widehat{\boldsymbol{\beta}}^{*})\right|^{1/dp}}\right) = 100 \times \max\left(0, \frac{R^2_{\text{JWR}} - 1}{R^2_{\text{JWR}}}\right)$$

where $\text{Var}(\widehat{\boldsymbol{\beta}}^{*})$ and $\text{Var}(\widehat{\boldsymbol{\beta}})$ are the estimated covariance matrices of the regression coefficients under the random-effects and fixed-effects models, defined in (1) and (2), respectively.

For each outcome i, the expressions for the JWR statistics are

$$R_{\text{JWR},\,i} = \left\{\left|\text{Var}(\widehat{\boldsymbol{\beta}}^{*}_{i})\right| \left|\text{Var}(\widehat{\boldsymbol{\beta}}_{i})\right|^{-1}\right\}^{1/2p}$$

$$I^2_{\text{JWR},\,i} = 100 \times \max\left(0, \frac{R^2_{\text{JWR},\,i} - 1}{R^2_{\text{JWR},\,i}}\right)$$

where $\text{Var}(\widehat{\boldsymbol{\beta}}^{*}_{i})$ and $\text{Var}(\widehat{\boldsymbol{\beta}}_{i})$ are the $p \times p$ submatrices of $\text{Var}(\widehat{\boldsymbol{\beta}}^{*})$ and $\text{Var}(\widehat{\boldsymbol{\beta}})$ corresponding to the regression coefficient parameters of the ith outcome.

When the `jwriley(`*depnames*`)` option is specified, it is possible to compute the JWR statistics for any subset of outcomes defined by *depnames*. For a subset $\{s\}$ of outcome variables,

$$R_{\text{JWR},\,\{s\}} = \left\{\left|\text{Var}(\widehat{\boldsymbol{\beta}}^{*}_{\{s\}})\right| \left|\text{Var}(\widehat{\boldsymbol{\beta}}_{\{s\}})\right|^{-1}\right\}^{1/2\nu}$$

$$I^2_{\text{JWR},\,\{s\}} = 100 \times \max\left(0, \frac{R^2_{\text{JWR},\,\{s\}} - 1}{R^2_{\text{JWR},\,\{s\}}}\right)$$

where $\widehat{\boldsymbol{\beta}}^{*}_{\{s\}}$ and $\widehat{\boldsymbol{\beta}}_{\{s\}}$ represent the set of regression coefficients corresponding to the subset $\{s\}$ and ν is the number of the corresponding estimated regression coefficients.

White heterogeneity statistics

White (2011) (option `white`) suggested to compute the I^2 statistic separately for each outcome but did not provide a method to quantify heterogeneity jointly based on all outcomes. For each outcome i, the White I^2 statistic is defined as

$$I^2_{\mathrm{W},\,i} = \frac{\widehat{\boldsymbol{\Sigma}}_{ii}}{\widehat{\boldsymbol{\Sigma}}_{ii} + s^2_{\mathrm{W},\,i}}$$

where $\widehat{\boldsymbol{\Sigma}}_{ii}$ is the ith diagonal element of the estimated between-study covariance matrix $\widehat{\boldsymbol{\Sigma}}$ described in *Iterative methods for computing* $\boldsymbol{\Sigma}$ and *Noniterative method for computing* $\boldsymbol{\Sigma}$ of [META] **meta mvregress**.

The typical within-study variance $s^2_{\mathrm{W},\,i}$ is computed from a univariate meta-regression of the ith outcome on the moderators. Therefore, $s^2_{\mathrm{W},\,i}$ is computed in the same way as s^2 in *Residual heterogeneity measures* in *Methods and formulas* in [META] **meta regress** and is given by

$$s^2_{\mathrm{W},\,i} = \frac{n_i - p}{\mathrm{tr}(\mathbf{P})}$$

where n_i is the number of observations on outcome i, which equals K if there are no missing values, and $\mathbf{P}$ is given by (1) in *Methods and formulas* of [META] **meta regress**, with $\mathbf{A} = \mathbf{W}$ being the $n_i \times n_i$ diagonal matrix containing inverse-variance weights for outcome i.

The $I^2_{\mathrm{W},\,i}$ statistic reduces to I^2_{res} reported by (4) in *Methods and formulas* of [META] **meta regress** when $d = 1$.

References

Higgins, J. P. T., and S. G. Thompson. 2002. Quantifying heterogeneity in a meta-analysis. *Statistics in Medicine* 21: 1539–1558. https://doi.org/10.1002/sim.1186.

Jackson, D., I. R. White, and R. D. Riley. 2012. Quantifying the impact of between-study heterogeneity in multivariate meta-analyses. *Statistics in Medicine* 31: 3805–3820. https://doi.org/10.1002/sim.5453.

White, I. R. 2011. Multivariate random-effects meta-regression: Updates to mvmeta. *Stata Journal* 11: 240–254.

Also see

[META] **meta mvregress** — Multivariate meta-regression

[META] **meta mvregress postestimation** — Postestimation tools for meta mvregress

[U] **20 Estimation and postestimation commands**

estat recovariance — Display estimated random-effects covariance matrices

Description Menu for estat Syntax Options
Remarks and examples Stored results Also see

Description

`estat recovariance` is for use after a random-effects multivariate meta-regression fit by `meta mvregress` or a multilevel meta-regression model fit by `meta meregress` or `meta multilevel`. It displays the estimated variance–covariance matrix of the random effects.

Menu for estat

Statistics > Postestimation

Syntax

`estat recovariance` [`, relevel(`*levelvar*`) correlation` *matlist_options*]

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

Options

`relevel(`*levelvar*`)` specifies the level in the model for which the random-effects covariance matrix is to be displayed. By default, the covariance matrices for all levels in the model are displayed. *levelvar* is the name of the variable describing the grouping at that level. This option is not supported after `meta mvregress`.

`correlation` displays the covariance matrix as a correlation matrix.

matlist_options are style and formatting options that control how the matrix (or matrices) is displayed; see [P] **matlist** for a list of options that are available.

Remarks and examples

See example 9 of [META] **meta mvregress** and example 6 of [META] **meta meregress**.

Stored results

`estat recovariance` stores the following in `r()`:

Scalars

`r(relevels)`	number of levels

Matrices

`r(Cov#)`	level-# random-effects covariance matrix
`r(cov)`	random-effects covariance matrix (after `meta mvregress`)
`r(Corr#)`	level-# random-effects correlation matrix (if option `correlation` was specified)
`r(corr)`	random-effects correlation matrix (after `meta mvregress` and if option `correlation` was specified)

For a G-level nested model, # can be any integer between 2 and G.

Also see

[META] **meta me postestimation** — Postestimation tools for multilevel mixed-effects meta-analysis

[META] **meta mvregress postestimation** — Postestimation tools for meta mvregress

[META] **meta meregress** — Multilevel mixed-effects meta-regression

[META] **meta multilevel** — Multilevel random-intercepts meta-regression

[META] **meta mvregress** — Multivariate meta-regression

[U] **20 Estimation and postestimation commands**

estat sd — Display variance components as standard deviations and correlations

Description Menu for estat Syntax Options
Remarks and examples Stored results Also see

Description

`estat sd` displays the random-effects estimates as standard deviations and correlations. `estat sd` is available only after a random-effects multivariate meta-regression fit by `meta mvregress` or a multilevel meta-regression model fit by `meta meregress` or `meta multilevel`.

Menu for estat

Statistics > Postestimation

Syntax

`estat sd` [, `variance verbose post coeflegend`]

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

Options

`variance` specifies that `estat sd` display the random-effects parameter estimates as variances and covariances. If the `post` option is specified, the estimated variances and covariances are posted to `e()`.

`verbose` specifies that the full estimation table be displayed. By default, only the random-effects parameters are displayed. This option is implied when `post` is specified.

`post` causes `estat sd` to behave like a Stata estimation (e-class) command. `estat sd` posts the vector of calculated standard deviation and correlation parameters to `e()`, so that you can treat the estimated parameters just as you would results from any other estimation command.

The following option is not shown in the dialog box:

`coeflegend` specifies that the legend of the coefficients and how to specify them in an expression be displayed rather than displaying the statistics for the coefficients. This option is allowed only if `post` is also specified.

Remarks and examples

See example 7 of [META] **meta mvregress** and example 6 of [META] **meta meregress**.

Stored results

estat sd stores the following in r():

Matrices

r(b)	coefficient vector
r(V)	variance–covariance matrix of the estimators
r(table)	table of results

Note: After meta mvregress, either the verbose or the post option must be specified for r(table) to be stored.

If post is specified, estat sd stores the following in e():

Macros

e(cmd)	estat sd
e(properties)	b V

Matrices

e(b)	coefficient vector
e(V)	variance–covariance matrix of the estimators

Also see

[META] **meta me postestimation** — Postestimation tools for multilevel mixed-effects meta-analysis

[META] **meta mvregress postestimation** — Postestimation tools for meta mvregress

[META] **meta meregress** — Multilevel mixed-effects meta-regression

[META] **meta multilevel** — Multilevel random-intercepts meta-regression

[META] **meta mvregress** — Multivariate meta-regression

[U] 20 Estimation and postestimation commands

Glossary

Begg test, Begg and Mazumdar test. A nonparametric rank correlation test for funnel-plot asymmetry of Begg and Mazumdar (1994). It tests whether Kendall's rank correlation between the effect sizes and their variances equals zero. The regression-based tests such as the tend to perform better in terms of type I error than the rank correlation test. This test is no longer recommended in the literature and provided mainly for completeness. See [META] **meta bias**.

between-study covariance matrix. In the context of multivariate meta-regression, the between-study covariance matrix, $\mathbf{\Sigma}$, is the covariance matrix of the random effects. It models heterogeneity between studies. By default, no structure is assumed when estimating $\mathbf{\Sigma}$, but several covariance structures may be considered; see *Random-effects covariance structures* in *Methods and formulas* in [META] **meta mvregress**.

between-study sample size. The number of studies in a meta-analysis.

between-study variability. Also known as between-study heterogeneity; see *heterogeneity*.

BLUPs. BLUPs are best linear unbiased predictions of either random effects or linear combinations of random effects. In linear models containing random effects, these effects are not estimated directly but instead are integrated out of the estimation. Once the fixed effects and variance components have been estimated, you can use these estimates to predict group-specific random effects. These predictions are called BLUPs because they are unbiased and have minimal mean squared errors among all linear functions of the response.

bubble plot. A scatterplot of effect size against a continuous covariate (moderator) in the meta-regression. The size of points representing the studies is proportional to study weights from a fixed-effects or, optionally, random-effects meta-analysis.

clinical heterogeneity. According to Deeks, Higgins, and Altman (2017), it is "variability in the participants, interventions and outcomes studied". Clinical variation will lead to heterogeneity if the effect size is affected by any of these varying factors.

Cochran's Q statistic. See *Q statistic*.

Cohen's d. An effect-size measure introduced by Cohen (1988) for a two-group comparison of continuous outcomes. It is a standardized mean difference where the difference between the two group means is usually divided by the standard deviation pooled across both groups. See *Standardized mean difference* of *Methods and formulas* in [META] **meta esize**.

combined effect size. See *overall effect size*.

common-effect meta-analysis model. A meta-analysis model that assumes that a single (common) true effect size underlies all the primary study results. See *Common-effect ("fixed-effect") model* in [META] **Intro**.

correlation data. Meta-analysis of correlation data deals with aggregating evidence about the correlation between two variables of interest. Each study must report the correlation coefficient and the study sample size to estimate the overall correlation.

cumulative meta-analysis. Cumulative meta-analysis performs multiple meta-analyses by accumulating studies one at a time. The studies are first ordered with respect to the variable of interest, the ordering variable. Meta-analysis summaries are then computed for the first study, for the first two studies, for the first three studies, and so on. The last meta-analysis will correspond to the standard meta-analysis using all studies. See [META] **meta summarize**.

cumulative overall effect sizes. In the context of cumulative meta-analysis, cumulative (overall) effect sizes refer to the overall effect sizes computed by accumulating one study at a time. That is, the first overall effect size is simply the individual effect size of the first study. The second overall effect size is the overall effect size computed based on the first two studies. The third overall effect size is the overall effect size computed based on the first three studies. And so on. The last effect size in a cumulative meta-analysis corresponds to the overall effect size computed using all studies in a standard meta-analysis.

DerSimonian–Laird's method. A noniterative, random-effects estimator of the between-study variance parameter that does not make any assumptions about the distribution of random effects. This method was introduced in DerSimonian and Laird (1986). Historically, random-effects meta-analysis has been based solely on this method. See *Noniterative methods* of *Methods and formulas* in [META] **meta summarize**.

effect size. A numerical summary of the group differences or of association between factors. For example, effect sizes for two-group comparisons include standardized and unstandardized mean differences, odds ratio, risk ratio, hazard ratio, and correlation coefficient. See [META] **meta esize**.

effect-size standard errors. See *sampling standard errors*.

effect-size variances. See *sampling variances*.

Egger test. A regression-based test for funnel-plot asymmetry of (Egger et al. 1997). This is the test of a slope coefficient in a weighted regression of the effect sizes on their standard errors. See [META] **meta bias**.

Fisher's z transformation. A transformation introduced by Fisher (1921). In the context of meta-analysis, it is applied to correlations to stabilize their variances—the variance of a transformed correlation does not depend on the sample correlation. This transformation also leads to a CI between -1 and 1 for the correlation in each study.

fixed-effects meta-analysis model. A meta-analysis model that assumes effect sizes are different across the studies and estimates a weighted average of their true values. This model is not valid for making inferences about studies beyond those included in the meta-analysis. See *Fixed-effects model* in [META] **Intro**.

fixed-effects meta-regression. Meta-regression that assumes a fixed-effects meta-analysis model. This regression model does not account for residual heterogeneity. See *Introduction* in [META] **meta regress**.

forest plot. A forest plot is a graphical representation of the results of a meta-analysis. In addition to meta-analytic summary such as overall effect size and its confidence interval and heterogeneity statistics and tests, it includes study-specific effect sizes and confidence intervals. See [META] **meta forestplot**.

Freeman–Tukey transformation. A transformation introduced by Freeman and Tukey (1950). In the context of meta-analysis, it is applied to proportions to stabilize their variances—the variance of a transformed proportion does not depend on the number of events. This transformation also leads to a CI between 0 and 1 for the proportion in each study. And it does not require a continuity correction when a study has zero events (successes) or failures.

funnel plot. The funnel plot is a scatterplot of the study-specific effect sizes against measures of study precision. This plot is commonly used to explore small-study effects or publication bias. In the absence of small-study effects, the shape of the scatterplot should resemble a symmetric inverted funnel. See [META] **meta funnelplot**.

Galbraith plot. The Galbraith plot is a scatterplot of the standardized effect sizes (z scores) against precision (inverse standard errors). It is commonly used to assess heterogeneity and for detecting potential outliers. When the number of studies is so large that it becomes inconvenient to present the results on a forest plot, the Galbraith plot provides a good alternative to report the meta-analysis results.

Glass's Δ. An effect-size measure introduced by Smith and Glass (1977) for a two-group comparison of continuous outcomes. It is a standardized mean difference where the difference between the two group means is divided by the sample standard deviation of the control group. Another variation of this statistic uses the sample standard deviation of the treatment group for the standardization. See *Standardized mean difference* of *Methods and formulas* in [META] **meta esize**.

grey literature. In the context of meta-analysis, grey literature refers to the literature that is difficult to obtain; it is thus rarely included in a meta-analysis.

H^2 statistic. A statistic for assessing heterogeneity. A value of $H^2 = 1$ indicates perfect homogeneity among the studies. See *Heterogeneity measures* of *Methods and formulas* in [META] **meta summarize**.

Hedges's g. An effect-size measure introduced by Hedges (1981) for a two-group comparison of continuous outcomes. It is a Cohen's d statistic adjusted for bias. See *Standardized mean difference* of *Methods and formulas* in [META] **meta esize**.

heterogeneity. In a meta-analysis, statistical heterogeneity, or simply heterogeneity, refers to the variability between the study-specific effect sizes that cannot be explained by a random variation. See *Heterogeneity* in [META] **Intro**.

heterogeneity parameter. In a random-effects meta-analysis, the variance of the random effects, τ^2, is used to account for the between-study heterogeneity. It is often referred to as the "heterogeneity parameter".

homogeneity. The opposite of heterogeneity.

homogeneity test. A test based on Cochran's Q statistic for assessing whether effect sizes from studies in a meta-analysis are homogeneous. See *Homogeneity test* of *Methods and formulas* in [META] **meta summarize**.

I^2 statistic. A statistic for assessing heterogeneity. It estimates the proportion of variation between the effect sizes due to heterogeneity relative to the pure sampling variation. $I^2 > 50$ indicates substantial heterogeneity. See *Heterogeneity measures* of *Methods and formulas* in [META] **meta summarize** for a standard meta-analysis. Also see *Multilevel heterogeneity statistics* in [META] **estat heterogeneity (me)** and [META] **estat heterogeneity (mv)** for multilevel and multivariate meta-analysis models.

intervention effects. See *effect size*.

inverse-variance method. A method of estimating the overall effect size as a weighted average of the study-specific effect sizes by using the weights that are inversely related to the variance (Whitehead and Whitehead 1991). This method is applicable to all meta-analysis models and all types of effect sizes.

Jackson–White–Riley method. In the context of multivariate meta-regression, the Jackson–White–Riley method provides a noniterative random-effects estimator of the between-study covariance matrix $\boldsymbol{\Sigma}$. This method was introduced by Jackson, White, and Riley (2013) and can be thought of as an extension of the univariate DerSimonian–Laird method to the multivariate setting.

L'Abbé plot. A scatterplot of the summary outcome measure such as log odds in the control group on the x axis and of that in the treatment group on the y axis. It is used with a two-group comparison of binary outcomes to inspect the range of group-level summary outcome measures among the studies to identify excessive heterogeneity. See [META] **meta labbeplot**.

large-strata limiting model. A model assumption for two-sample binary data in which the number of studies remains fixed but similar cell sizes in the 2×2 tables increase. See Robins, Breslow, and Greenland (1986).

leave-one-out meta-analysis. The leave-one-out meta-analysis performs multiple meta-analyses, where each analysis is produced by excluding a single study. It is a useful tool to assess the influence of a single study on the meta-analysis results and for identifying potential outliers.

Mantel–Haenszel method. In the context of meta-analysis, the Mantel–Haenszel method combines odds ratios, risk ratios, and risk differences. This method performs well in the presence of sparse data. For nonsparse data, its results are similar to those of the inverse-variance method. It was introduced by Mantel and Haenszel (1959) for odds ratios and extended to risk ratios and risk differences by Greenland and Robins (1985). See *Mantel–Haenszel method for two-group comparison of binary outcomes* of *Methods and formulas* in [META] **meta summarize**.

meta data. `meta` data are the data that were `meta` set (or declared) by either `meta set` or `meta esize`. `meta` data store key variables and characteristics about your meta-analysis specifications, which will be used by all `meta` commands during your meta-analysis session. Thus, declaration of your data as `meta` data is the first step of your meta-analysis in Stata. This step helps minimize mistakes and saves you time—you need to specify the necessary information only once. Also see [META] **meta data**.

meta settings. Meta settings refers to the meta-analysis information specified during the declaration of the `meta` data via `meta set` or `meta esize`. This includes the declared effect size, meta-analysis model, estimation method, confidence level, and more. See *Declaring meta-analysis information* in [META] **meta data** for details.

meta-analysis. The statistical analysis that combines quantitative results from multiple individual studies into a single result. It is often performed as part of a systematic review. See *Brief overview of meta-analysis* in [META] **Intro**.

meta-regression. A weighted regression of study effect sizes on study-level covariates or moderators. You can think of it as an extension of standard meta-analysis to incorporate the moderators to account for between-study heterogeneity. See [META] **meta regress**.

methodological heterogeneity. Variability in study design and conduct (Deeks, Higgins, and Altman 2017). See *Heterogeneity* in [META] **Intro**.

mixed-treatment studies. See *multiple-treatment studies*.

moderator. A moderator is a study-level covariate that may help explain between-study heterogeneity. If the moderator is categorical, its effect may be investigated by a subgroup analysis (see [META] **meta summarize**); if the moderator is continuous, its effect may be investigated by a meta-regression. See [META] **meta regress**.

multilevel meta-analysis. An extension of (univariate) meta-analysis to the analysis of potentially dependent effect sizes reported by studies that exhibit a hierarchical structure. For example, effect sizes can be nested within studies that are themselves nested within higher groups such as school districts. Like univariate meta-analysis, the goals of multilevel meta-analysis are to obtain an estimate of the overall effect size when it is sensible and to assess the heterogeneity across the different levels of the hierarchy. See [META] **meta meregress** and [META] **meta multilevel**.

multilevel meta-regression. A multilevel mixed-effects regression of study effect sizes on study-level covariates or moderators. You can think of it as an extension of multilevel meta-analysis to incorporate moderators to account for the heterogeneity among the effect sizes. You may also view it as a generalization of a (univariate) meta-regression to the setting where studies exhibit a hierarchical structure. See [META] **meta meregress** and [META] **meta multilevel**.

multiple subgroup analyses. Subgroup analysis performed separately for each of multiple categorical variables. See [META] **meta summarize**.

multiple-endpoint studies. Studies that compare a treatment group with a control group just like in standard meta-analysis, but more than one outcome (endpoint) is usually of interest. In the context of multivariate meta-regression, the effect sizes that compare these endpoints across the two groups are usually correlated because they were computed on the same set of subjects for each endpoint.

multiple-treatment studies. Studies that compare multiple (more than two) treatment groups. In the context of multivariate meta-regression, effect sizes that compare these groups are usually correlated because they share a common group. For example, an odds ratio that compares group A with group B is correlated with an odds ratio that compares group B with group C because they were computed based on the common group B.

multiplicative dispersion parameter. In a fixed-effects meta-regression, the multiplicative dispersion parameter is a multiplicative factor applied to the variance of each effect size to account for residual heterogeneity. See *Introduction* of [META] **meta regress**.

multiplicative meta-regression. A fixed-effects meta-regression that accounts for residual heterogeneity through a dispersion parameter ϕ applied (multiplicatively) to each effect-size variance. See *Introduction* of [META] **meta regress**.

multivariate meta-analysis. An extension of (univariate) meta-analysis to the analysis of multiple, usually dependent, effect sizes reported by each study. Like univariate meta-analysis, the goal of multivariate meta-analysis is to obtain an estimate of the multivariate overall effect size when it is sensible. See [META] **meta mvregress**.

multivariate meta-regression. A multivariate regression of study effect sizes on study-level covariates or moderators. You can think of it as an extension of multivariate meta-analysis to incorporate moderators to account for between-study heterogeneity. You may also view it as a generalization of a (univariate) meta-regression to multiple outcomes. See [META] **meta mvregress**.

narrative review. In a narrative review, the conclusion about the findings from multiple studies is given by a person, an expert in a particular field, based on his or her research of the studies. This approach is typically subjective and does not allow to account for certain aspects of the studies such as study heterogeneity and publication bias.

odds ratio. A ratio of the odds of a success in one group (treatment group) to those of another group (control group). It is often used as an effect size for comparing binary outcomes of two groups. See [META] **meta esize**.

one-sample binary data. Data collected on a binary outcome to estimate a population proportion or compare it with a reference value. Also see *prevalence data* and *two-sample binary data*.

overall effect size. The main target of interest in meta-analysis. Its interpretation depends on the assumed meta-analysis model. In a common-effect model, it is the common effect size of the studies. In a fixed-effects model, it is a weighted average of the true study-specific effect sizes. In a random-effects model, it is the mean of the distribution of the effect sizes. The overall effect size is usually denoted by `theta` in the output. Also see *Meta-analysis models* in [META] **Intro**.

Peto's method. A method for combining odds ratios that is often used with sparse 2×2 tables. This method does not require a zero-cell adjustment. See *Peto's method for odds ratios* of *Methods and formulas* in [META] **meta summarize**.

pooled effect size. See *overall effect size*.

prediction interval. In a random-effects meta-analysis, a $100(1-\alpha)\%$ prediction interval indicates that the true effect sizes in $100(1-\alpha)\%$ of new studies will lie within the interval. See *Prediction intervals* of *Methods and formulas* in [META] **meta summarize**.

prevalence. The proportion of subjects in a population that experiences a certain event of interest (success) at a specific period of time.

prevalence data. Meta-analysis of prevalence data deals with aggregating evidence about the prevalence of a certain event of interest. It is also known as meta-analysis of proportions. Each study must report the number of events and the study sample size to estimate the overall prevalence. Also see *one-sample binary data*.

primary study. The original study in which data are collected. An observation in a meta-analysis represents a primary study.

pseudo confidence interval. Pseudo confidence intervals refer to the confidence intervals as constructed by the standard funnel plot. See [META] **meta funnelplot**.

publication bias. Publication bias is known in the meta-analysis literature as an association between the likelihood of a publication and the statistical significance of a study result. See *Publication bias* in [META] **Intro**.

Q **statistic**. The test statistic of the homogeneity test. See *Homogeneity test* of *Methods and formulas* in [META] **meta summarize**.

random-effects meta-analysis model. A meta-analysis model that assumes that the study effects are random; that is, the studies used in the meta-analysis represent a random sample from a larger population of similar studies. See *Random-effects model* in [META] **Intro**.

random-effects meta-regression. Meta-regression that assumes a random-effects meta-analysis model. This regression model accounts for residual heterogeneity via an additive error term. See *Introduction* in [META] **meta regress**.

random-intercepts multilevel meta-regression. A special type of multilevel meta-analysis where random intercepts are the only type of random effects present in the model. In other words, the model does not include any random slopes. See *Introduction* in [META] **meta multilevel**.

randomized controlled trial. A randomized controlled trial is an experiment in which participants are randomly assigned to two or more different treatment groups. Randomized controlled trials are commonly used in clinical research to determine the effectiveness of new treatments. By design, they avoid bias in the treatment estimates.

rate ratio. See *risk ratio*.

relative risk. See *risk ratio*.

reporting bias. Systematic difference between the studies selected in a meta-analysis and all the studies relevant to the research question of interest. Also see *publication bias*.

residual heterogeneity. In the meta-regression context, this is the remaining variability between the studies not accounted for by the moderators. It is usually captured by the heterogeneity parameter in a random-effects meta-regression or by a multiplicative dispersion parameter in a fixed-effects meta-regression.

risk ratio. A ratio of the success probability in one group (treatment) to that of another group (control). It is often used as an effect size for comparing binary outcomes of two groups. See [META] **meta esize**.

sampling standard errors. Standard errors of the effect sizes, also known as within-study standard errors in the context of multivariate meta-analysis and standard meta-analysis. See *Introduction* in [META] **meta meregress**.

sampling variances. Variances of the effect sizes, also known as within-study variances in the context of multivariate meta-analysis and standard meta-analysis. See *Introduction* in [META] **meta meregress**.

sensitivity analysis. In the context of meta-analysis, sensitivity analyses are used to assess how robust the meta-analysis results are to assumptions made about the data and meta-analysis models. See [META] **meta summarize**, [META] **meta regress**, and [META] **meta mvregress**.

significance contours. In the context of a funnel plot ([META] **meta funnelplot**), significance contours (or contour lines of statistical significance) are the contour lines corresponding to the tests of significance of individual effect sizes for a given significance level $\alpha = c/100$. In other words, if a study falls in the shaded area of a c-level contour, it is considered not statistically significant at the α level based on a test of significance of the study effect size.

single subgroup analysis. Subgroup analysis performed for one categorical variable. See [META] **meta summarize**.

small-study effects. Small-study effects arise when the results of smaller studies differ systematically from the results of larger studies. See *Introduction* of [META] **meta funnelplot**.

sparse data. For two-sample binary data, a 2×2 table is considered sparse if any of the cell counts are small.

sparse data limiting model. A model assumption for two-sample binary data in which the number of 2×2 tables (studies) increases but the cell sizes remain fixed. See Robins, Breslow, and Greenland (1986).

standard meta-analysis. The classical meta-analysis setting, where each study reports a single effect size and all effect sizes are assumed to be independent.

statistical heterogeneity. See *heterogeneity*.

study precision. Study precision is a function of a study sample size or study variability. Typically, study precision is measured by the inverse of the effect-sizes standard errors, $1/\hat{\sigma}_j$, but other measures are also used. For instance, in a funnel pot, multiple precision metrics such as variances and sample sizes are considered. More precise studies (with larger sample sizes and smaller variances) are assigned larger weights in a meta-analysis.

subgroup analysis. A subgroup analysis divides the studies into groups and then estimates the overall effect size for each of the groups. The goal of subgroup analysis is to compare the overall effect sizes and explore heterogeneity between the subgroups. See [META] **meta summarize** and [META] **meta forestplot**.

subgroup heterogeneity. In the context of meta-analysis, subgroup heterogeneity is between-study heterogeneity induced by the differences between effect sizes of groups defined by one or more categorical variables. See [META] **meta** and [META] **meta summarize**.

summary data. In the context of meta-analysis, we use the term summary data to mean summary statistics that are used to compute the effect sizes and their standard errors for each study in the meta-analysis. For example, for a two-group comparison of continuous outcomes, the summary data contain the number of observations, means, and standard deviations in each group for each study. For a two-group comparison of binary outcomes, the summary data contain the 2×2 tables for each study. When we estimate a single proportion, the summary data contain the numbers of successes and the study sample sizes. For correlation data, the summary data consist of the correlation coefficients and the study sample sizes. See [META] **meta esize**.

summary effect. See *overall effect size*.

systematic review. A procedure that uses systematic and well-defined methods to find, select, and evaluate relevant research studies to answer a specific research question. It typically involves collecting and analyzing summary data of the selected studies. Meta-analysis is the statistical analysis used as part of a systematic review.

trim-and-fill method. A method of testing and adjusting for publication bias in meta-analysis; see [META] **meta trimfill**.

typical within-study variance, **typical sampling variance**. A term coined by Higgins and Thompson (2002) to describe a summary or an "average" of the within-study variances. The value of the typical within-study variance is used in the computation of the heterogeneity statistic I^2.

two-sample binary data. Data collected on binary outcomes to compare proportions in two groups. Also see *one-sample binary data*.

two-sample continuous data. Data collected on continuous outcomes to compare means in two groups.

within-study covariance matrix. In the context of multivariate meta-regression, the within-study covariance matrix, $\mathbf{\Lambda}_j$, is the covariance matrix that models the dependence among the effect sizes within each study. The matrix is assumed to be known and does not require estimation.

zero-cell adjustment. Adjustment made to cells of 2×2 tables containing zero cells for a two-group comparison of binary outcomes and to cells of studies with zero or all successes when estimating a single proportion (or prevalence). In the meta-analysis of binary data, zero-cell counts pose difficulties when computing odds ratios, risk ratios, and proportions. Therefore, it is common to make zero-cell adjustments, such as adding a small number to all cells containing zeros. See *Zero-cells adjustments for two-sample case* and *Zero-cells adjustments for one-sample case* in *Methods and formulas* of [META] **meta esize**.

References

Begg, C. B., and M. Mazumdar. 1994. Operating characteristics of a rank correlation test for publication bias. *Biometrics* 50: 1088–1101. https://doi.org/10.2307/2533446.

Cohen, J. 1988. *Statistical Power Analysis for the Behavioral Sciences*. 2nd ed. Hillsdale, NJ: Erlbaum.

Deeks, J. J., J. P. T. Higgins, and D. G. Altman. 2017. "Analysing data and undertaking meta-analyses". In *Cochrane Handbook for Systematic Reviews of Interventions Version 5.2.0*, edited by J. P. T. Higgins and S. Green. London: The Cochrane Collaboration. https://training.cochrane.org/handbook.

DerSimonian, R., and N. M. Laird. 1986. Meta-analysis in clinical trials. *Controlled Clinical Trials* 7: 177–188. https://doi.org/10.1016/0197-2456(86)90046-2.

Egger, M., G. Davey Smith, M. Schneider, and C. Minder. 1997. Bias in meta-analysis detected by a simple, graphical test. *BMJ* 315: 629–634. https://doi.org/10.1136/bmj.315.7109.629.

Fisher, R. A. 1921. On the "probable error" of a coefficient of correlation deduced from a small sample. *Metron* 1: 3–32.

Freeman, M. F., and J. W. Tukey. 1950. Transformations related to the angular and the square root. *Annals of Mathematical Statistics* 21: 607–611. https://doi.org/10.1214/aoms/1177729756.

Greenland, S., and J. M. Robins. 1985. Estimation of a common effect parameter from sparse follow-up data. *Biometrics* 41: 55–68. https://doi.org/10.2307/2530643.

Hedges, L. V. 1981. Distribution theory for Glass's estimator of effect size and related estimators. *Journal of Educational Statistics* 6: 107–128. https://doi.org/10.2307/1164588.

Higgins, J. P. T., and S. G. Thompson. 2002. Quantifying heterogeneity in a meta-analysis. *Statistics in Medicine* 21: 1539–1558. https://doi.org/10.1002/sim.1186.

Jackson, D., I. R. White, and R. D. Riley. 2013. A matrix-based method of moments for fitting the multivariate random effects model for meta-analysis and meta-regression. *Biometrical Journal* 55: 231–245. https://doi.org/10.1002/bimj.201200152.

Mantel, N., and W. Haenszel. 1959. Statistical aspects of the analysis of data from retrospective studies of disease. *Journal of the National Cancer Institute* 22: 719–748. Reprinted in *Evolution of Epidemiologic Ideas: Annotated Readings on Concepts and Methods*, ed. S. Greenland, pp. 112–141. Newton Lower Falls, MA: Epidemiology Resources.

Robins, J. M., N. E. Breslow, and S. Greenland. 1986. Estimators of the Mantel–Haenszel variance consistent in both sparse data and large-strata limiting models. *Biometrics* 42: 311–323. https://doi.org/10.2307/2531052.

Smith, M. L., and G. V. Glass. 1977. Meta-analysis of psychotherapy outcome studies. *American Psychologist* 32: 752–760. https://doi.org/10.1037/0003-066X.32.9.752.

Whitehead, A., and J. Whitehead. 1991. A general parametric approach to the meta-analysis of randomized clinical trials. *Statistics in Medicine* 10: 1665–1677. https://doi.org/10.1002/sim.4780101105.

Subject and author index

See the combined subject index and the combined author index in the *Stata Index*.