

estat bubbleplot — Bubble plots after meta regress[Description](#)[Options](#)[Also see](#)[Quick start](#)[Remarks and examples](#)[Menu](#)[Methods and formulas](#)[Syntax](#)[References](#)

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
```

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]
```

<i>options</i>	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(<i>line_options</i>)	affect rendition of the plotted regression line
CI plot	
ciopts(<i>ciopts</i>)	affect rendition of the plotted CI band
Add plots	
addplot(<i>plot</i>)	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 [nocl](#).

[ci](#) and [nocl](#) 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

[stata.com](http://www.stata.com)

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/r16/bcgset
(Efficacy of BCG vaccine against tuberculosis; set with -meta esize-)
. meta query, short
-> meta esize npost nnegt nposc 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
Method: REML
Number of obs =      13
Residual heterogeneity:
      tau2 = .07635
      I2 (%) = 68.39
      H2 = 3.16
      R-squared (%) = 75.63
Wald chi2(1) = 16.36
Prob > chi2 = 0.0001
```

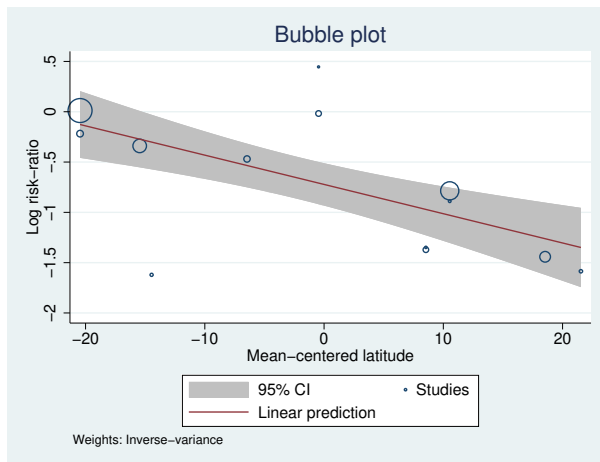
_meta_es	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
latitude_c	-.0291017	.0071953	-4.04	0.000	-.0432043	-.0149991
_cons	-.7223204	.1076535	-6.71	0.000	-.9333174	-.5113234

Test of residual homogeneity: $Q_{res} = \chi^2(11) = 30.73$ 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
```

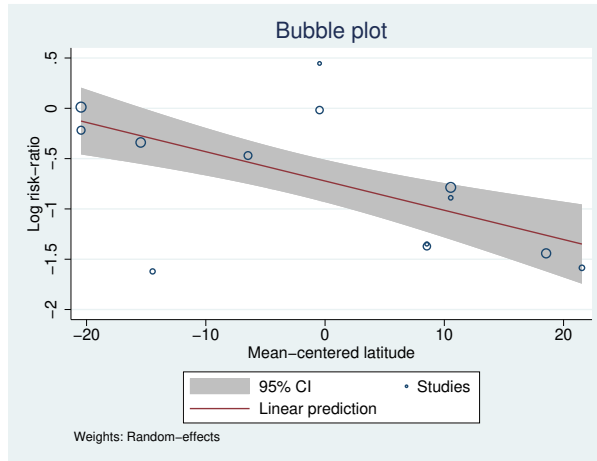


The graph shows the log risk-ratios plotted against the mean-centered latitudes of the studies' locations. By default, the regression line and corresponding confidence intervals are plotted. We could suppress these and plot just the bubbles with options `noregline` and `noci`, respectively. The regression 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
```

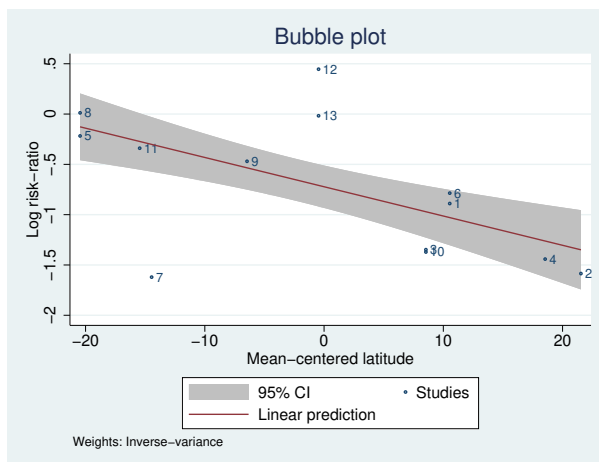


With random-effects weights, the sizes of the bubbles are more uniform across the studies as compared with precision (fixed-effects) weights used in [example 1](#). This will always be true except when $\hat{\tau}^2 = 0$, in which case the bubble sizes will be identical with both types of weights.

► 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)
```



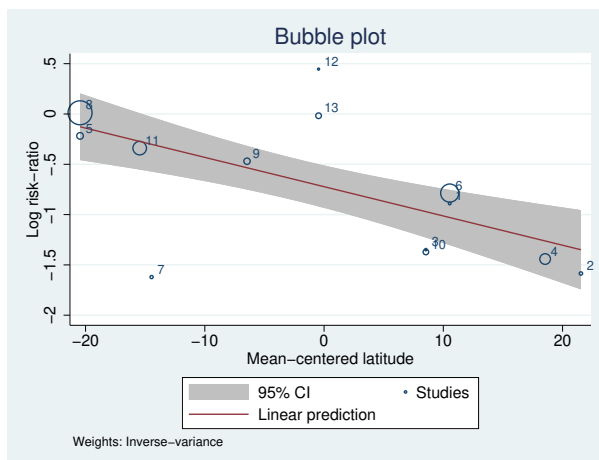
Specifying the `mlabel()` option causes all the markers to have the same size. One way to get around this is by using the `addplot()` option.

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(navy) 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(navy)` option controls the color of the study labels (navy here). 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.

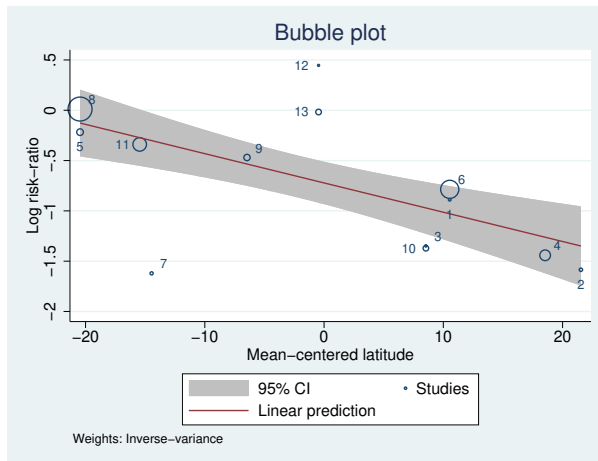
Let's generate our position variable.

```
. 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.

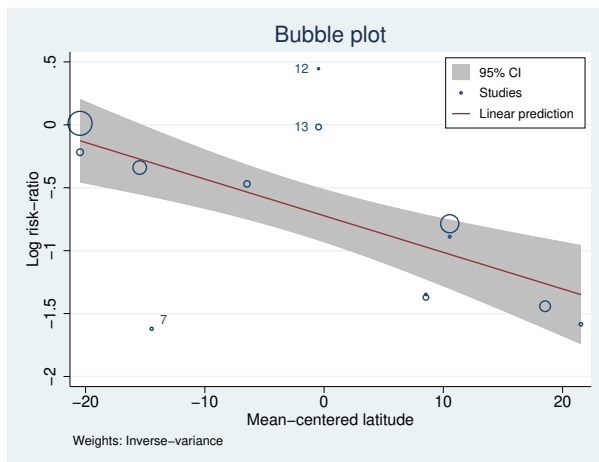
We now use a similar specification of `addplot()` from [example 3](#), but here we add `mlabvpos(pos)` and `mlabgap(*2)` to double the gap size between the marker labels and the markers so that the trial labels do not touch the hollow circles; see trials 6 and 8.

```
. 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 `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) cols(1) size(small) symxsize(*0.3))
. 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. `cols(1)` specifies that all legend keys be displayed in one column. `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. ◀

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 $\hat{\beta}_0 + \hat{\beta}_1 x_1 = \mathbf{x}_j \hat{\beta}$. The CIs are computed as

$$\mathbf{x}_j \hat{\beta} \pm z_{1-\alpha/2} \widehat{SE}(\mathbf{x}_j \hat{\beta})$$

where the computation of $\widehat{SE}(\mathbf{x}_j \hat{\beta})$ 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.
- Thompson, S. G., and J. P. T. Higgins. 2002. How should meta-regression analyses be undertaken and interpreted? *Statistics in Medicine* 21: 1559–1573.
- Thompson, S. G., and S. J. Sharp. 1999. Explaining heterogeneity in meta-analysis: A comparison of methods. *Statistics in Medicine* 18: 2693–2708.

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