mepoisson — Multilevel mixed-effects Poisson regression

Description

`mepoisson` fits mixed-effects models for count responses. The conditional distribution of the response given the random effects is assumed to be Poisson.

`mepoisson` performs optimization with the original metric of variance components. When variance components are near the boundary of the parameter space, you may consider using the `meqrpoisson` command, which provides alternative parameterizations of variance components; see [ME] **meqrpoisson**.

Quick start

Without weights

Two-level Poisson regression of `y` on `x` with random intercepts by `lev2`

```
mepoisson y x || lev2:
```

Add `evar` measuring exposure

```
mepoisson y x, exposure(evar) || lev2:
```

As above, but report incidence-rate ratios

```
mepoisson y x, exposure(evar) || lev2:, irr
```

Add indicators for levels of categorical variable `a` and random coefficients on `x`

```
mepoisson y x i.a || lev2: x, irr
```

Three-level random-intercept model of `y` on `x` with `lev2` nested within `lev3`

```
mepoisson y x || lev3: || lev2:
```

With weights

Two-level Poisson regression of `y` on `x` with random intercepts by `lev2` and observation-level frequency weights `wvar1`

```
mepoisson y x [fweight=wvar1] || lev2:
```

Two-level random-intercept model from a two-stage sampling design with PSUs identified by `psu` using PSU-level and observation-level sampling weights `wvar2` and `wvar1`, respectively

```
mepoisson y x [pweight=wvar1] || psu:, pweight(wvar2)
```

Add secondary sampling stage with units identified by `ssu` having weights `wvar2` and PSU-level weights `wvar3` for a three-level random-intercept model

```
mepoisson y x [pw=wvar1] || psu:, pw(wvar3) || ssu:, pw(wvar2)
```

Same as above, but svyset data first

```
svyset psu [pw=wvar3] || ssu, weight(wvar2) || _n, weight(wvar1)
svy: mepoisson y x || psu: || ssu:
```

Menu

Statistics > Multilevel mixed-effects models > Poisson regression

Syntax

`mepoisson` *depvar fe_equation* [|| *re_equation*] [|| *re_equation* ...] [, *options*]

where the syntax of *fe_equation* is

[*indepvars*] [*if*] [*in*] [*weight*] [, *fe_options*]

and the syntax of *re_equation* is one of the following:

for random coefficients and intercepts

levelvar: [*varlist*] [, *re_options*]

for random effects among the values of a factor variable

levelvar: `R.`*varname*

levelvar is a variable identifying the group structure for the random effects at that level or is `_all` representing one group comprising all observations.

fe_options	Description
Model	
`noconstant`	suppress the constant term from the fixed-effects equation
`exposure(`$varname_e$`)`	include $\ln(varname_e)$ in model with coefficient constrained to 1
`offset(`$varname_o$`)`	include $varname_o$ in model with coefficient constrained to 1

re_options	Description
Model	
`covariance(`*vartype*`)`	variance–covariance structure of the random effects
`noconstant`	suppress constant term from the random-effects equation
`fweight(`*varname*`)`	frequency weights at higher levels
`iweight(`*varname*`)`	importance weights at higher levels
`pweight(`*varname*`)`	sampling weights at higher levels

options	Description
Model	
constraints(*constraints*)	apply specified linear constraints
collinear	keep collinear variables
SE/Robust	
vce(*vcetype*)	*vcetype* may be oim, robust, or cluster *clustvar*
Reporting	
level(#)	set confidence level; default is level(95)
irr	report fixed-effects coefficients as incidence-rate ratios
nocnsreport	do not display constraints
notable	suppress coefficient table
noheader	suppress output header
nogroup	suppress table summarizing groups
nolrtest	do not perform likelihood-ratio test comparing with Poisson regression
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Integration	
intmethod(*intmethod*)	integration method
intpoints(#)	set the number of integration (quadrature) points for all levels; default is intpoints(7)
Maximization	
maximize_options	control the maximization process; seldom used
startvalues(*svmethod*)	method for obtaining starting values
startgrid[(*gridspec*)]	perform a grid search to improve starting values
noestimate	do not fit the model; show starting values instead
dnumerical	use numerical derivative techniques
coeflegend	display legend instead of statistics

vartype	Description
independent	one unique variance parameter per random effect, all covariances 0; the default unless the R. notation is used
exchangeable	equal variances for random effects, and one common pairwise covariance
identity	equal variances for random effects, all covariances 0; the default if the R. notation is used
unstructured	all variances and covariances to be distinctly estimated
fixed(*matname*)	user-selected variances and covariances constrained to specified values; the remaining variances and covariances unrestricted
pattern(*matname*)	user-selected variances and covariances constrained to be equal; the remaining variances and covariances unrestricted

intmethod	Description
mvaghermite	mean–variance adaptive Gauss–Hermite quadrature; the default unless a crossed random-effects model is fit
mcaghermite	mode-curvature adaptive Gauss–Hermite quadrature
ghermite	nonadaptive Gauss–Hermite quadrature
laplace	Laplacian approximation; the default for crossed random-effects models

indepvars may contain factor variables; see **[U] 11.4.3 Factor variables**.

depvar, *indepvars*, and *varlist* may contain time-series operators; see **[U] 11.4.4 Time-series varlists**.

by and svy are allowed; see **[U] 11.1.10 Prefix commands**.

vce() and weights are not allowed with the svy prefix; see **[SVY] svy**.

fweights, iweights, and pweights are allowed; see **[U] 11.1.6 weight**. Only one type of weight may be specified. Weights are not supported under the Laplacian approximation or for crossed models.

startvalues(), startgrid, noestimate, dnumerical, 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 and may be specified for the fixed-effects equation and for any of or all the random-effects equations.

exposure(*varname*$_e$) specifies a variable that reflects the amount of exposure over which the *depvar* events were observed for each observation; ln(*varname*$_e$) is included in the fixed-effects portion of the model with the coefficient constrained to be 1.

offset(*varname*$_o$) specifies that *varname*$_o$ be included in the fixed-effects portion of the model with the coefficient constrained to be 1.

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, fixed(*matname*), or pattern(*matname*).

covariance(independent) covariance structure allows for a distinct variance for each random effect within a random-effects equation and assumes that all covariances are 0. The default is covariance(independent) unless a crossed random-effects model is fit, in which case the default is covariance(identity).

covariance(exchangeable) structure specifies one common variance for all random effects and one common pairwise covariance.

covariance(identity) is short for "multiple of the identity"; that is, all variances are equal and all covariances are 0.

covariance(unstructured) allows for all variances and covariances to be distinct. If an equation consists of p random-effects terms, the unstructured covariance matrix will have $p(p+1)/2$ unique parameters.

covariance(fixed(*matname*)) and covariance(pattern(*matname*)) covariance structures provide a convenient way to impose constraints on variances and covariances of random effects. Each specification requires a *matname* that defines the restrictions placed on variances and covariances. Only elements in the lower triangle of *matname* are used, and row and column names of *matname* are ignored. A missing value in *matname* means that a given element is unrestricted. In a `fixed(`*matname*`)` covariance structure, (co)variance (i, j) is constrained to equal the value specified in the i, jth entry of *matname*. In a `pattern(`*matname*`)` covariance structure, (co)variances (i, j) and (k, l) are constrained to be equal if $matname[i, j] = matname[k, l]$.

`fweight(`*varname*`)` specifies frequency weights at higher levels in a multilevel model, whereas frequency weights at the first level (the observation level) are specified in the usual manner, for example, `[fw=`*fwtvar1*`]`. *varname* can be any valid Stata variable name, and you can specify `fweight()` at levels two and higher of a multilevel model. For example, in the two-level model

```
. mecmd fixed_portion [fw = wt1] || school: ... , fweight(wt2) ...
```

the variable `wt1` would hold the first-level (the observation-level) frequency weights, and `wt2` would hold the second-level (the school-level) frequency weights.

`iweight(`*varname*`)` specifies importance weights at higher levels in a multilevel model, whereas importance weights at the first level (the observation level) are specified in the usual manner, for example, `[iw=`*iwtvar1*`]`. *varname* can be any valid Stata variable name, and you can specify `iweight()` at levels two and higher of a multilevel model. For example, in the two-level model

```
. mecmd fixed_portion [iw = wt1] || school: ... , iweight(wt2) ...
```

the variable `wt1` would hold the first-level (the observation-level) importance weights, and `wt2` would hold the second-level (the school-level) importance weights.

`pweight(`*varname*`)` specifies sampling weights at higher levels in a multilevel model, whereas sampling weights at the first level (the observation level) are specified in the usual manner, for example, `[pw=`*pwtvar1*`]`. *varname* can be any valid Stata variable name, and you can specify `pweight()` at levels two and higher of a multilevel model. For example, in the two-level model

```
. mecmd fixed_portion [pw = wt1] || school: ... , pweight(wt2) ...
```

variable `wt1` would hold the first-level (the observation-level) sampling weights, and `wt2` would hold the second-level (the school-level) sampling weights.

`constraints(`*constraints*`)`, `collinear`; see [R] **estimation options**.

SE/Robust

`vce(`*vcetype*`)` specifies the type of standard error reported, which includes types that are derived from asymptotic theory (`oim`), that are robust to some kinds of misspecification (`robust`), and that allow for intragroup correlation (`cluster` *clustvar*); see [R] ***vce_option***. If `vce(robust)` is specified, robust variances are clustered at the highest level in the multilevel model.

Reporting

`level(`*#*`)`; see [R] **estimation options**.

`irr` reports estimated fixed-effects coefficients transformed to incidence-rate ratios, that is, $\exp(\beta)$ rather than β. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. `irr` may be specified either at estimation or upon replay.

`nocnsreport`; see [R] **estimation options**.

`notable` suppresses the estimation table, either at estimation or upon replay.

`noheader` suppresses the output header, either at estimation or upon replay.

`nogroup` suppresses the display of group summary information (number of groups, average group size, minimum, and maximum) from the output header.

nolrtest prevents mepoisson from performing a likelihood-ratio test that compares the mixed-effects Poisson model with standard (marginal) Poisson regression. This option may also be specified upon replay to suppress this test from the output.

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**.

Integration

intmethod(*intmethod*) specifies the integration method to be used for the random-effects model. mvaghermite performs mean–variance adaptive Gauss–Hermite quadrature; mcaghermite performs mode-curvature adaptive Gauss–Hermite quadrature; ghermite performs nonadaptive Gauss–Hermite quadrature; and laplace performs the Laplacian approximation, equivalent to mode-curvature adaptive Gaussian quadrature with one integration point.

The default integration method is mvaghermite unless a crossed random-effects model is fit, in which case the default integration method is laplace. The Laplacian approximation has been known to produce biased parameter estimates; however, the bias tends to be more prominent in the estimates of the variance components rather than in the estimates of the fixed effects.

For crossed random-effects models, estimation with more than one quadrature point may be prohibitively intensive even for a small number of levels. For this reason, the integration method defaults to the Laplacian approximation. You may override this behavior by specifying a different integration method.

intpoints(#) sets the number of integration points for quadrature. The default is intpoints(7), which means that seven quadrature points are used for each level of random effects. This option is not allowed with intmethod(laplace).

The more integration points, the more accurate the approximation to the log likelihood. However, computation time increases as a function of the number of quadrature points raised to a power equaling the dimension of the random-effects specification. In crossed random-effects models and in models with many levels or many random coefficients, this increase can be substantial.

Maximization

maximize_options: difficult, technique(*algorithm_spec*), iterate(#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(#), ltolerance(#), nrtolerance(#), nonrtolerance, and from(*init_specs*); see [R] **maximize**. Those that require special mention for mepoisson are listed below.

from() accepts a properly labeled vector of initial values or a list of coefficient names with values. A list of values is not allowed.

The following options are available with mepoisson but are not shown in the dialog box:

startvalues(*svmethod*), startgrid[(*gridspec*)], noestimate, and dnumerical; see [ME] **meglm**.

coeflegend; see [R] **estimation options**.

Remarks and examples

stata.com

For a general introduction to me commands, see [ME] **me**.

Remarks are presented under the following headings:

Introduction

Mixed-effects Poisson regression is Poisson regression containing both fixed effects and random effects. In longitudinal data and panel data, random effects are useful for modeling intracluster correlation; that is, observations in the same cluster are correlated because they share common cluster-level random effects.

Comprehensive treatments of mixed models are provided by, for example, Searle, Casella, and McCulloch (1992); Verbeke and Molenberghs (2000); Raudenbush and Bryk (2002); Demidenko (2004); Hedeker and Gibbons (2006); McCulloch, Searle, and Neuhaus (2008); and Rabe-Hesketh and Skrondal (2012). Rabe-Hesketh and Skrondal (2012, chap. 13) is a good introductory read on applied multilevel modeling of count data.

mepoisson allows for not just one, but many levels of nested clusters. For example, in a three-level model you can specify random effects for schools and then random effects for classes nested within schools. In this model, the observations (presumably, the students) comprise the first level, the classes comprise the second level, and the schools comprise the third level.

However, for simplicity, for now we consider the two-level model, where for a series of M independent clusters, and conditional on a set of random effects $\mathbf{u}_j$,

$$\Pr(y_{ij} = y|\mathbf{x}_{ij}, \mathbf{u}_j) = \exp\left(-\mu_{ij}\right) \mu_{ij}^{y}/y! \qquad (1)$$

for $\mu_{ij} = \exp(\mathbf{x}_{ij}\boldsymbol{\beta} + \mathbf{z}_{ij}\mathbf{u}_j)$, $j = 1,\dots,M$ clusters, with cluster j consisting of $i = 1,\dots,n_j$ observations. The responses are counts y_{ij}. The $1 \times p$ row vector $\mathbf{x}_{ij}$ are the covariates for the fixed effects, analogous to the covariates you would find in a standard Poisson regression model, with regression coefficients (fixed effects) $\boldsymbol{\beta}$. For notational convenience here and throughout this manual entry, we suppress the dependence of y_{ij} on $\mathbf{x}_{ij}$.

The $1 \times q$ vector $\mathbf{z}_{ij}$ are the covariates corresponding to the random effects and can be used to represent both random intercepts and random coefficients. For example, in a random-intercept model, $\mathbf{z}_{ij}$ is simply the scalar 1. The random effects $\mathbf{u}_j$ are M realizations from a multivariate normal distribution with mean $\mathbf{0}$ and $q \times q$ variance matrix $\boldsymbol{\Sigma}$. The random effects are not directly estimated as model parameters but are instead summarized according to the unique elements of $\boldsymbol{\Sigma}$, known as variance components. One special case of (1) places $\mathbf{z}_{ij} = \mathbf{x}_{ij}$ so that all covariate effects are essentially random and distributed as multivariate normal with mean $\boldsymbol{\beta}$ and variance $\boldsymbol{\Sigma}$.

As noted in chapter 13.7 of Rabe-Hesketh and Skrondal (2012), the inclusion of a random intercept causes the marginal variance of y_{ij} to be greater than the marginal mean, provided the variance of the random intercept is not 0. Thus the random intercept in a mixed-effects Poisson model produces overdispersion, a measure of variability above and beyond that allowed by a Poisson process; see [R] **nbreg** and [ME] **menbreg**.

Model (1) is a member of the class of generalized linear mixed models (GLMMs), which generalize the linear mixed-effects (LME) model to non-Gaussian responses. You can fit LMEs in Stata by using mixed and fit GLMMs by using meglm. Because of the relationship between LMEs and GLMMs, there is insight to be gained through examination of the linear mixed model. This is especially true for Stata users because the terminology, syntax, options, and output for fitting these types of models are nearly identical. See [ME] **mixed** and the references therein, particularly in the *Introduction*, for more information.

Log-likelihood calculations for fitting any generalized mixed-effects model require integrating out the random effects. One widely used modern method is to directly estimate the integral required to calculate the log likelihood by Gauss–Hermite quadrature or some variation thereof. Because the log likelihood itself is estimated, this method has the advantage of permitting likelihood-ratio tests for comparing nested models. Also, if done correctly, quadrature approximations can be quite accurate, thus minimizing bias.

`mepoisson` supports three types of Gauss–Hermite quadrature and the Laplacian approximation method; see *Methods and formulas* of [ME] **meglm** for details.

Below we present two short examples of mixed-effects Poisson regression; refer to [ME] **me** and [ME] **meglm** for additional examples including crossed random-effects models.

A two-level model

We begin with a simple application of (1) as a two-level model, because a one-level model, in our terminology, is just standard Poisson regression; see [R] **poisson**.

▷ Example 1

Breslow and Clayton (1993) fit a mixed-effects Poisson model to data from a randomized trial of the drug progabide for the treatment of epilepsy.

```
. use http://www.stata-press.com/data/r14/epilepsy
(Epilepsy data; progabide drug treatment)
. describe
Contains data from http://www.stata-press.com/data/r14/epilepsy.dta
  obs:           236                        Epilepsy data; progabide drug
                                              treatment
 vars:             8                        31 May 2014 14:09
 size:         4,956                        (_dta has notes)
-------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
-------------------------------------------------------------------------------
subject         byte    %9.0g                 Subject ID: 1-59
seizures        int     %9.0g                 No. of seizures
treat           byte    %9.0g                 1: progabide; 0: placebo
visit           float   %9.0g                 Dr. visit; coded as (-.3, -.1,
                                                .1, .3)
lage            float   %9.0g                 log(age), mean-centered
lbas            float   %9.0g                 log(0.25*baseline seizures),
                                                mean-centered
lbas_trt        float   %9.0g                 lbas/treat interaction
v4              byte    %8.0g                 Fourth visit indicator
-------------------------------------------------------------------------------
Sorted by: subject
```

Originally from Thall and Vail (1990), data were collected on 59 subjects (31 progabide, 28 placebo). The number of epileptic seizures (`seizures`) was recorded during the two weeks prior to each of four doctor visits (`visit`). The treatment group is identified by the indicator variable `treat`. Data were also collected on the logarithm of age (`lage`) and the logarithm of one-quarter the number of seizures during the eight weeks prior to the study (`lbas`). The variable `lbas_trt` represents the interaction between lbas and treatment. lage, lbas, and lbas_trt are mean centered. Because the study originally noted a substantial decrease in seizures prior to the fourth doctor visit, an indicator v4 for the fourth visit was also recorded.

Breslow and Clayton (1993) fit a random-effects Poisson model for the number of observed seizures,

$$\log(\mu_{ij}) = \beta_0 + \beta_1 \texttt{treat}_{ij} + \beta_2 \texttt{lbas}_{ij} + \beta_3 \texttt{lbas\_trt}_{ij} + \beta_4 \texttt{lage}_{ij} + \beta_5 \texttt{v4}_{ij} + u_j$$

for $j = 1, \dots, 59$ subjects and $i = 1, \dots, 4$ visits. The random effects u_j are assumed to be normally distributed with mean 0 and variance σ_u^2.

```
. mepoisson seizures treat lbas lbas_trt lage v4 || subject:
Fitting fixed-effects model:
Iteration 0:   log likelihood = -1016.4106
Iteration 1:   log likelihood = -819.20112
Iteration 2:   log likelihood = -817.66006
Iteration 3:   log likelihood = -817.65925
Iteration 4:   log likelihood = -817.65925
Refining starting values:
Grid node 0:   log likelihood = -680.40523
Fitting full model:
Iteration 0:   log likelihood = -680.40523  (not concave)
Iteration 1:   log likelihood = -672.95766  (not concave)
Iteration 2:   log likelihood = -667.14039
Iteration 3:   log likelihood = -665.51823
Iteration 4:   log likelihood = -665.29165
Iteration 5:   log likelihood = -665.29067
Iteration 6:   log likelihood = -665.29067
Mixed-effects Poisson regression                Number of obs     =        236
Group variable:         subject                 Number of groups  =         59
                                                Obs per group:
                                                              min =          4
                                                              avg =        4.0
                                                              max =          4
Integration method: mvaghermite                 Integration pts.  =          7
                                                Wald chi2(5)      =     121.70
Log likelihood = -665.29067                     Prob > chi2       =     0.0000
```

seizures	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
treat	-.9330306	.4007512	-2.33	0.020	-1.718489	-.1475727
lbas	.8844225	.1312033	6.74	0.000	.6272689	1.141576
lbas_trt	.3382561	.2033021	1.66	0.096	-.0602087	.736721
lage	.4842226	.3471905	1.39	0.163	-.1962582	1.164703
v4	-.1610871	.0545758	-2.95	0.003	-.2680536	-.0541206
_cons	2.154578	.2199928	9.79	0.000	1.7234	2.585756
subject						
var(_cons)	.2528664	.0589844			.1600801	.399434

```
LR test vs. Poisson model: chibar2(01) = 304.74       Prob >= chibar2 = 0.0000
```

The number of seizures before the fourth visit does exhibit a significant drop, and the patients on progabide demonstrate a decrease in frequency of seizures compared with the placebo group. The subject-specific random effects also appear significant: $\widehat{\sigma}_u^2 = 0.25$ with standard error 0.06.

Because this is a simple random-intercept model, you can obtain equivalent results by using xtpoisson with the re and normal options.

◁

A three-level model

mepoisson can also fit higher-level models with multiple levels of nested random effects.

▷ Example 2

Rabe-Hesketh and Skrondal (2012, exercise 13.7) describe data from the *Atlas of Cancer Mortality in the European Economic Community* (EEC) (Smans, Mair, and Boyle 1993). The data were analyzed in Langford, Bentham, and McDonald (1998) and record the number of deaths among males due to malignant melanoma during 1971–1980.

```
. use http://www.stata-press.com/data/r14/melanoma
(Skin cancer (melanoma) data)

. describe

Contains data from http://www.stata-press.com/data/r14/melanoma.dta
  obs:           354                          Skin cancer (melanoma) data
 vars:             6                          30 May 2014 17:10
 size:         4,956                          (_dta has notes)
--------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
--------------------------------------------------------------------------------
nation          byte    %11.0g     n          Nation ID
region          byte    %9.0g                 Region ID: EEC level-I areas
county          int     %9.0g                 County ID: EEC level-II/level-III
                                                areas
deaths          int     %9.0g                 No. deaths during 1971-1980
expected        float   %9.0g                 No. expected deaths
uv              float   %9.0g                 UV dose, mean-centered
--------------------------------------------------------------------------------
Sorted by:
```

Nine European nations (variable nation) are represented, and data were collected over geographical regions defined by EEC statistical services as level I areas (variable region), with deaths being recorded for each of 354 counties, which are level II or level III EEC-defined areas (variable county, which identifies the observations). Counties are nested within regions, and regions are nested within nations.

The variable deaths records the number of deaths for each county, and expected records the expected number of deaths (the exposure) on the basis of crude rates for the combined countries. Finally, the variable uv is a measure of exposure to ultraviolet (UV) radiation.

In modeling the number of deaths, one possibility is to include dummy variables for the nine nations as fixed effects. Another is to treat these as random effects and fit the three-level random-intercept Poisson model,

$$\log(\mu_{ijk}) = \log(\texttt{expected}_{ijk}) + \beta_0 + \beta_1 \texttt{uv}_{ijk} + u_k + v_{jk}$$

for nation k, region j, and county i. The model includes an exposure term for expected deaths.

```
. mepoisson deaths c.uv##c.uv, exposure(expected) || nation: || region:
Fitting fixed-effects model:
Iteration 0:   log likelihood = -2136.0274
Iteration 1:   log likelihood =  -1723.127
Iteration 2:   log likelihood = -1722.9762
Iteration 3:   log likelihood = -1722.9762
Refining starting values:
Grid node 0:   log likelihood = -1166.9773
Fitting full model:
Iteration 0:   log likelihood = -1166.9773  (not concave)
Iteration 1:   log likelihood = -1152.6069  (not concave)
Iteration 2:   log likelihood =  -1151.902  (not concave)
Iteration 3:   log likelihood =  -1127.412  (not concave)
Iteration 4:   log likelihood = -1101.9248
Iteration 5:   log likelihood = -1094.1984
Iteration 6:   log likelihood =   -1088.05
Iteration 7:   log likelihood = -1086.9097
Iteration 8:   log likelihood = -1086.8995
Iteration 9:   log likelihood = -1086.8994
Mixed-effects Poisson regression                Number of obs     =        354

-------------------------------------------------------------
                |   No. of       Observations per Group
 Group Variable |   Groups    Minimum    Average    Maximum
----------------+--------------------------------------------
         nation |        9          3       39.3         95
         region |       78          1        4.5         13
-------------------------------------------------------------

Integration method: mvaghermite                 Integration pts.  =          7

                                                Wald chi2(2)      =      25.70
Log likelihood = -1086.8994                     Prob > chi2       =     0.0000
------------------------------------------------------------------------------
      deaths |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          uv |   .0057002   .0137919     0.41   0.679    -.0213315    .0327318
             |
  c.uv#c.uv |  -.0058377   .0013879    -4.21   0.000     -.008558   -.0031174
             |
       _cons |   .1289989   .1581224     0.82   0.415    -.1809154    .4389132
ln(expected) |          1  (exposure)
-------------+----------------------------------------------------------------
nation       |
   var(_cons)|   .1841878   .0945722                      .0673298    .5038655
-------------+----------------------------------------------------------------
nation>region|
   var(_cons)|   .0382645   .0087757                      .0244105    .0599811
------------------------------------------------------------------------------
LR test vs. Poisson model: chi2(2) = 1272.15              Prob > chi2 = 0.0000
Note: LR test is conservative and provided only for reference.
```

By including an exposure variable that is an expected rate, we are in effect specifying a linear model for the log of the standardized mortality ratio, the ratio of observed deaths to expected deaths that is based on a reference population, the reference population being all nine nations.

Looking at the estimated variance components, we can see that there is more unobserved variability between nations than between regions within each nation. This may be due to, for example, country-specific informational campaigns on the risks of sun exposure.

◁

Stored results

mepoisson stores the following in e():

Scalars	
e(N)	number of observations
e(k)	number of parameters
e(k_dv)	number of dependent variables
e(k_eq)	number of equations in e(b)
e(k_eq_model)	number of equations in overall model test
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(df_m)	model degrees of freedom
e(ll)	log likelihood
e(N_clust)	number of clusters
e(chi2)	χ^2
e(p)	significance
e(ll_c)	log likelihood, comparison model
e(chi2_c)	χ^2, comparison model
e(df_c)	degrees of freedom, comparison model
e(p_c)	significance, comparison model
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise
Macros	
e(cmd)	meglm
e(cmd2)	mepoisson
e(cmdline)	command as typed
e(depvar)	name of dependent variable
e(wtype)	weight type
e(wexp)	weight expression (first-level weights)
e(fweight*k*)	fweight variable for *k*th highest level, if specified
e(iweight*k*)	iweight variable for *k*th highest level, if specified
e(pweight*k*)	pweight variable for *k*th highest level, if specified
e(covariates)	list of covariates
e(ivars)	grouping variables
e(model)	poisson
e(title)	title in estimation output
e(link)	log
e(family)	poisson
e(clustvar)	name of cluster variable
e(offset)	offset
e(exposure)	exposure variable
e(intmethod)	integration method
e(n_quad)	number of integration points
e(chi2type)	Wald; type of model χ^2
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. Err.
e(opt)	type of optimization
e(which)	max or min; whether optimizer is to perform maximization or minimization
e(ml_method)	type of ml method

`e(user)`	name of likelihood-evaluator program
`e(technique)`	maximization technique
`e(datasignature)`	the checksum
`e(datasignaturevars)`	variables used in calculation of checksum
`e(properties)`	`b V`
`e(estat_cmd)`	program used to implement `estat`
`e(predict)`	program used to implement `predict`
`e(marginsnotok)`	predictions disallowed by `margins`
`e(marginswtype)`	weight type for `margins`
`e(marginswexp)`	weight expression for `margins`
`e(asbalanced)`	factor variables `fvset` as `asbalanced`
`e(asobserved)`	factor variables `fvset` as `asobserved`

Matrices

`e(b)`	coefficient vector
`e(Cns)`	constraints matrix
`e(ilog)`	iteration log (up to 20 iterations)
`e(gradient)`	gradient vector
`e(N_g)`	group counts
`e(g_min)`	group-size minimums
`e(g_avg)`	group-size averages
`e(g_max)`	group-size maximums
`e(V)`	variance–covariance matrix of the estimators
`e(V_modelbased)`	model-based variance

Functions

`e(sample)`	marks estimation sample

Methods and formulas

In a two-level Poisson model, for cluster j, $j = 1, \dots, M$, the conditional distribution of $\mathbf{y}_j = (y_{j1}, \dots, y_{jn_j})'$, given a set of cluster-level random effects $\mathbf{u}_j$, is

$$f(\mathbf{y}_j|\mathbf{u}_j) = \prod_{i=1}^{n_j} \left[\{\exp\left(\mathbf{x}_{ij}\boldsymbol{\beta} + \mathbf{z}_{ij}\mathbf{u}_j\right)\}^{y_{ij}} \exp\left\{-\exp\left(\mathbf{x}_{ij}\boldsymbol{\beta} + \mathbf{z}_{ij}\mathbf{u}_j\right)\right\}/y_{ij}!\right]$$
$$= \exp\left[\sum_{i=1}^{n_j} \left\{y_{ij}\left(\mathbf{x}_{ij}\boldsymbol{\beta} + \mathbf{z}_{ij}\mathbf{u}_j\right) - \exp\left(\mathbf{x}_{ij}\boldsymbol{\beta} + \mathbf{z}_{ij}\mathbf{u}_j\right) - \log(y_{ij}!)\right\}\right]$$

Defining $c(\mathbf{y}_j) = \sum_{i=1}^{n_j} \log(y_{ij}!)$, where $c(\mathbf{y}_j)$ does not depend on the model parameters, we can express the above compactly in matrix notation,

$$f(\mathbf{y}_j|\mathbf{u}_j) = \exp\left\{\mathbf{y}_j'\left(\mathbf{X}_j\boldsymbol{\beta} + \mathbf{Z}_j\mathbf{u}_j\right) - \mathbf{1}'\exp\left(\mathbf{X}_j\boldsymbol{\beta} + \mathbf{Z}_j\mathbf{u}_j\right) - c(\mathbf{y}_j)\right\}$$

where $\mathbf{X}_j$ is formed by stacking the row vectors $\mathbf{x}_{ij}$ and $\mathbf{Z}_j$ is formed by stacking the row vectors $\mathbf{z}_{ij}$. We extend the definition of $\exp(\cdot)$ to be a vector function where necessary.

Because the prior distribution of $\mathbf{u}_j$ is multivariate normal with mean $\mathbf{0}$ and $q \times q$ variance matrix $\boldsymbol{\Sigma}$, the likelihood contribution for the jth cluster is obtained by integrating $\mathbf{u}_j$ out of the joint density $f(\mathbf{y}_j, \mathbf{u}_j)$,

$$\begin{aligned}\mathcal{L}_j(\boldsymbol{\beta}, \boldsymbol{\Sigma}) &= (2\pi)^{-q/2}|\boldsymbol{\Sigma}|^{-1/2}\int f(\mathbf{y}_j|\mathbf{u}_j)\exp\left(-\mathbf{u}_j'\boldsymbol{\Sigma}^{-1}\mathbf{u}_j/2\right)d\mathbf{u}_j \\ &= \exp\left\{-c(\mathbf{y}_j)\right\}(2\pi)^{-q/2}|\boldsymbol{\Sigma}|^{-1/2}\int \exp\left\{h(\boldsymbol{\beta}, \boldsymbol{\Sigma}, \mathbf{u}_j)\right\}d\mathbf{u}_j\end{aligned} \qquad (2)$$

where

$$h(\boldsymbol{\beta}, \boldsymbol{\Sigma}, \mathbf{u}_j) = \mathbf{y}_j' (\mathbf{X}_j\boldsymbol{\beta} + \mathbf{Z}_j\mathbf{u}_j) - \mathbf{1}' \exp(\mathbf{X}_j\boldsymbol{\beta} + \mathbf{Z}_j\mathbf{u}_j) - \mathbf{u}_j'\boldsymbol{\Sigma}^{-1}\mathbf{u}_j/2$$

and for convenience, in the arguments of $h(\cdot)$ we suppress the dependence on the observable data $(\mathbf{y}_j, \mathbf{X}_j, \mathbf{Z}_j)$.

The integration in (2) has no closed form and thus must be approximated. `mepoisson` offers four approximation methods: mean–variance adaptive Gauss–Hermite quadrature (default unless a crossed random-effects model is fit), mode-curvature adaptive Gauss–Hermite quadrature, nonadaptive Gauss–Hermite quadrature, and Laplacian approximation (default for crossed random-effects models).

The Laplacian approximation is based on a second-order Taylor expansion of $g(\boldsymbol{\beta}, \boldsymbol{\Sigma}, \mathbf{u}_j)$ about the value of $\mathbf{u}_j$ that maximizes it; see *Methods and formulas* in [ME] **meglm** for details.

Gaussian quadrature relies on transforming the multivariate integral in (2) into a set of nested univariate integrals. Each univariate integral can then be evaluated using a form of Gaussian quadrature; see *Methods and formulas* in [ME] **meglm** for details.

The log likelihood for the entire dataset is simply the sum of the contributions of the M individual clusters, namely, $\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\Sigma}) = \sum_{j=1}^{M} \mathcal{L}_j(\boldsymbol{\beta}, \boldsymbol{\Sigma})$.

Maximization of $\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\Sigma})$ is performed with respect to $(\boldsymbol{\beta}, \boldsymbol{\sigma}^2)$, where $\boldsymbol{\sigma}^2$ is a vector comprising the unique elements of $\boldsymbol{\Sigma}$. Parameter estimates are stored in `e(b)` as $(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\sigma}}^2)$, with the corresponding variance–covariance matrix stored in `e(V)`.

`mepoisson` supports multilevel weights and survey data; see *Methods and formulas* in [ME] **meglm** for details.

References

Andrews, M. J., T. Schank, and R. Upward. 2006. Practical fixed-effects estimation methods for the three-way error-components model. *Stata Journal* 6: 461–481.

Breslow, N. E., and D. G. Clayton. 1993. Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association* 88: 9–25.

Demidenko, E. 2004. *Mixed Models: Theory and Applications*. Hoboken, NJ: Wiley.

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.

Hedeker, D., and R. D. Gibbons. 2006. *Longitudinal Data Analysis*. Hoboken, NJ: Wiley.

Joe, H. 2008. Accuracy of Laplace approximation for discrete response mixed models. *Computational Statistics & Data Analysis* 52: 5066–5074.

Laird, N. M., and J. H. Ware. 1982. Random-effects models for longitudinal data. *Biometrics* 38: 963–974.

Langford, I. H., G. Bentham, and A. McDonald. 1998. Multi-level modelling of geographically aggregated health data: A case study on malignant melanoma mortality and UV exposure in the European community. *Statistics in Medicine* 17: 41–57.

Leyland, A. H., and H. Goldstein, ed. 2001. *Multilevel Modelling of Health Statistics*. New York: Wiley.

Lin, X., and N. E. Breslow. 1996. Bias correction in generalized linear mixed models with multiple components of dispersion. *Journal of the American Statistical Association* 91: 1007–1016.

Marchenko, Y. V. 2006. Estimating variance components in Stata. *Stata Journal* 6: 1–21.

McCulloch, C. E., S. R. Searle, and J. M. Neuhaus. 2008. *Generalized, Linear, and Mixed Models*. 2nd ed. Hoboken, NJ: Wiley.

McLachlan, G. J., and K. E. Basford. 1988. *Mixture Models: Inference and Applications to Clustering*. New York: Dekker.

Rabe-Hesketh, S., and A. Skrondal. 2012. *Multilevel and Longitudinal Modeling Using Stata*. 3rd ed. College Station, TX: Stata Press.

Rabe-Hesketh, S., A. Skrondal, and A. Pickles. 2005. Maximum likelihood estimation of limited and discrete dependent variable models with nested random effects. *Journal of Econometrics* 128: 301–323.

Raudenbush, S. W., and A. S. Bryk. 2002. *Hierarchical Linear Models: Applications and Data Analysis Methods*. 2nd ed. Thousand Oaks, CA: Sage.

Searle, S. R., G. Casella, and C. E. McCulloch. 1992. *Variance Components*. New York: Wiley.

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.

Skrondal, A., and S. Rabe-Hesketh. 2004. *Generalized Latent Variable Modeling: Multilevel, Longitudinal, and Structural Equation Models*. Boca Raton, FL: Chapman & Hall/CRC.

Smans, M., C. S. Mair, and P. Boyle. 1993. *Atlas of Cancer Mortality in the European Economic Community*. Lyon, France: IARC Scientific Publications.

Thall, P. F., and S. C. Vail. 1990. Some covariance models for longitudinal count data with overdispersion. *Biometrics* 46: 657–671.

Verbeke, G., and G. Molenberghs. 2000. *Linear Mixed Models for Longitudinal Data*. New York: Springer.

Also see

[ME] **mepoisson postestimation** — Postestimation tools for mepoisson

[ME] **menbreg** — Multilevel mixed-effects negative binomial regression

[ME] **meqrpoisson** — Multilevel mixed-effects Poisson regression (QR decomposition)

[ME] **me** — Introduction to multilevel mixed-effects models

[SEM] **intro 5** — Tour of models (*Multilevel mixed-effects models*)

[SVY] **svy estimation** — Estimation commands for survey data

[XT] **xtpoisson** — Fixed-effects, random-effects, and population-averaged Poisson models

[U] **20 Estimation and postestimation commands**