**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.

**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, weight(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.

<table>
<thead>
<tr>
<th>fe_options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td></td>
</tr>
<tr>
<td><code>noconstant</code></td>
<td>suppress the constant term from the fixed-effects equation</td>
</tr>
<tr>
<td><code>exposure(varname_o)</code></td>
<td>include ln(varname_o) in model with coefficient constrained to 1</td>
</tr>
<tr>
<td><code>offset(varname_o)</code></td>
<td>include varname_o in model with coefficient constrained to 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>re_options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td></td>
</tr>
<tr>
<td><code>covariance(vartype)</code></td>
<td>variance–covariance structure of the random effects</td>
</tr>
<tr>
<td><code>noconstant</code></td>
<td>suppress constant term from the random-effects equation</td>
</tr>
<tr>
<td><code>fweight(varname)</code></td>
<td>frequency weights at higher levels</td>
</tr>
<tr>
<td><code>iweight(varname)</code></td>
<td>importance weights at higher levels</td>
</tr>
<tr>
<td><code>pweight(varname)</code></td>
<td>sampling weights at higher levels</td>
</tr>
</tbody>
</table>
### options

<table>
<thead>
<tr>
<th>Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>constraints(constraints)</code></td>
<td>apply specified linear constraints</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SE/Robust</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>vce(vcetype)</code></td>
<td><code>vcetype</code> may be <code>oim</code>, <code>robust</code>, or <code>cluster clustvar</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reporting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>level(#)</code></td>
<td>set confidence level; default is <code>level(95)</code></td>
</tr>
<tr>
<td><code>irr</code></td>
<td>report fixed-effects coefficients as incidence-rate ratios</td>
</tr>
<tr>
<td><code>nocnsreport</code></td>
<td>do not display constraints</td>
</tr>
<tr>
<td><code>notable</code></td>
<td>suppress coefficient table</td>
</tr>
<tr>
<td><code>noheader</code></td>
<td>suppress output header</td>
</tr>
<tr>
<td><code>nogroup</code></td>
<td>suppress table summarizing groups</td>
</tr>
<tr>
<td><code>display_options</code></td>
<td>control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Integration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>intmethod(intmethod)</code></td>
<td>integration method</td>
</tr>
<tr>
<td><code>intpoints(#)</code></td>
<td>set the number of integration (quadrature) points for all levels; default is <code>intpoints(7)</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Maximization</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>maximize_options</code></td>
<td>control the maximization process; seldom used</td>
</tr>
<tr>
<td><code>startvalues(svmethod)</code></td>
<td>method for obtaining starting values</td>
</tr>
<tr>
<td><code>startgrid[ (gridspec) ]</code></td>
<td>perform a grid search to improve starting values</td>
</tr>
<tr>
<td><code>noestimate</code></td>
<td>do not fit the model; show starting values instead</td>
</tr>
<tr>
<td><code>dnumerical</code></td>
<td>use numerical derivative techniques</td>
</tr>
<tr>
<td><code>collinear</code></td>
<td>keep collinear variables</td>
</tr>
<tr>
<td><code>coeflegend</code></td>
<td>display legend instead of statistics</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>intmethod</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mvaghermite</td>
<td>mean–variance adaptive Gauss–Hermite quadrature; the default unless a crossed random-effects model is fit</td>
</tr>
<tr>
<td>mcaghermite</td>
<td>mode-curvature adaptive Gauss–Hermite quadrature</td>
</tr>
<tr>
<td>ghermite</td>
<td>nonadaptive Gauss–Hermite quadrature</td>
</tr>
<tr>
<td>laplace</td>
<td>Laplacian approximation; the default for crossed random-effects models</td>
</tr>
</tbody>
</table>

### Options

- **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; \(\text{ln}(\text{varname}_e)\) 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 \(i,j\)th 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.

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); see [R] Estimation options.

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.

level(#) specifies the level 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.

irr reports estimated fixed-effects coefficients transformed to incidence-rate ratios, that is, \(exp(\beta)\) rather than \(\beta\). 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.
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; `hermite` 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.`

collinear, coeflegend; see [R] Estimation options.
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.

`mepoisson` allows for many levels of random effects. 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 \( u_j \),

\[
Pr(y_{ij} = y | x_{ij}, u_j) = \exp (-\mu_{ij}) \mu_{ij}^y / y!
\]

for \( \mu_{ij} = \exp(x_{ij} \beta + z_{ij} u_j) \), \( j = 1, \ldots, M \) clusters, with cluster \( j \) consisting of \( i = 1, \ldots, n_j \) observations. The responses are counts \( y_{ij} \). The \( 1 \times p \) row vector \( x_{ij} \) 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) \( \beta \). For notational convenience here and throughout this manual entry, we suppress the dependence of \( y_{ij} \) on \( x_{ij} \).

The \( 1 \times q \) vector \( z_{ij} \) 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, \( z_{ij} \) is simply the scalar 1. The random effects \( u_j \) are \( M \) realizations from a multivariate normal distribution with mean 0 and \( q \times q \) variance matrix \( \Sigma \). The random effects are not directly estimated as model parameters but are instead summarized according to the unique elements of \( \Sigma \), known as variance components. One special case of (1) places \( z_{ij} = x_{ij} \) so that all covariate effects are essentially random and distributed as multivariate normal with mean \( \beta \) and variance \( \Sigma \).

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`.

Below we present examples of mixed-effects Poisson regression; refer to [ME] `me` and [ME] `meglm` for additional examples including crossed random-effects models. A two-level Poisson model can also be fit using `xtpoisson` with the `re` option; see [XT] `xtpoisson`. In the absence of random effects, mixed-effects Poisson regression reduces to standard Poisson regression; see [R] `poisson`.

Two-level models

> Example 1: Two-level random-intercept model

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 https://www.stata-press.com/data/r16/epilepsy
(Epilepsy data; progabide drug treatment)
. describe
Contains data from https://www.stata-press.com/data/r16/epilepsy.dta
   obs:    236 Epilepsy data; progabide drug treatment
   vars:    8 31 May 2018 14:09
   (_dta has notes)

<table>
<thead>
<tr>
<th>variable name</th>
<th>storage type</th>
<th>display format</th>
<th>value label</th>
<th>variable label</th>
</tr>
</thead>
<tbody>
<tr>
<td>subject</td>
<td>byte</td>
<td>%9.0g</td>
<td>Subject ID: 1-59</td>
<td></td>
</tr>
<tr>
<td>seizures</td>
<td>int</td>
<td>%9.0g</td>
<td>No. of seizures</td>
<td></td>
</tr>
<tr>
<td>treat</td>
<td>byte</td>
<td>%9.0g</td>
<td>1: progabide; 0: placebo</td>
<td></td>
</tr>
<tr>
<td>visit</td>
<td>float</td>
<td>%9.0g</td>
<td>Dr. visit; coded as (.3, -.1, .1, .3)</td>
<td></td>
</tr>
<tr>
<td>lage</td>
<td>float</td>
<td>%9.0g</td>
<td>log(age), mean-centered</td>
<td></td>
</tr>
<tr>
<td>lbas</td>
<td>float</td>
<td>%9.0g</td>
<td>log(0.25*baseline seizures), mean-centered</td>
<td></td>
</tr>
<tr>
<td>lbas_trt</td>
<td>float</td>
<td>%9.0g</td>
<td>lbas/treat interaction</td>
<td></td>
</tr>
<tr>
<td>v4</td>
<td>byte</td>
<td>%8.0g</td>
<td>Fourth visit indicator</td>
<td></td>
</tr>
</tbody>
</table>

Sorted by: subject

Originally from Thall and Vail (1990), data were collected on 59 subjects (31 on progabide, 28 on 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

\[
\log(\mu_{ij}) = \beta_0 + \beta_1 \text{treat}_{ij} + \beta_2 \text{lbas}_{ij} + \beta_3 \text{lbas}_{trt}_{ij} + \beta_4 \text{lage}_{ij} + \beta_5 \text{v4}_{ij} + u_j
\]

for \(j = 1, \ldots, 59\) subjects and \(i = 1, \ldots, 4\) visits. The random effects \(u_j\) are assumed to be normally distributed with mean 0 and variance \(\sigma_u^2\).
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: $\hat{\sigma}^2_u = 0.25$ with standard error 0.06.

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

\section*{Example 2: Two-level random-slope model}

In their study of \textit{PQL}, \cite{breslow1993} also fit a model where they dropped the fixed effect on v4 and replaced it with a random subject-specific linear trend over the four doctor visits. The model they fit is

\begin{equation}
\text{seizures} \sim \text{Poisson} \left( \exp \left( \beta_0 + \beta_1 \text{treat} + \beta_2 \text{lbas} + \beta_3 \text{lbas_trt} + \beta_4 \text{lage} + \beta_5 v4 \right) \right)
\end{equation}
\[
\log(\mu_{ij}) = \beta_0 + \beta_1 \text{treat}_{ij} + \beta_2 \text{lbas}_{ij} + \beta_3 \text{lbas_trt}_{ij} + \\
\beta_4 \text{lage}_{ij} + \beta_5 \text{visit}_{ij} + u_j + v_{j, \text{visit}_{ij}}
\]

where \((u_j, v_j)\) are bivariate normal with 0 mean and variance–covariance matrix

\[
\Sigma = \text{Var} \begin{bmatrix}
    u_j \\
    v_j
\end{bmatrix} = \begin{bmatrix}
    \sigma_u^2 & \sigma_{uv} \\
    \sigma_{uv} & \sigma_v^2
\end{bmatrix}
\]

```
. mepoisson seizures treat lbas lbas_trt lage visit || subject: visit,
   > covariance(unstructured) intpoints(9) nolog
```

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. = 9
Log likelihood = -655.68103 Wald chi2(5) = 115.56
Prob > chi2 = 0.0000

|          | Coef.     | Std. Err. | z     | P>|z| | [95% Conf. Interval] |
|----------|-----------|-----------|-------|------|---------------------|
| seizures |           |           |       |      |                     |
| treat    | -.9286592 | .4021715  | -2.31 | 0.021| -1.716901 to -.1404175 |
| lbas     | .8849762  | .1312535  | 6.74  | 0.000| .627724 to 1.142228  |
| lbas_trt | .3379759  | .2044471  | 1.65  | 0.098| -.062733 to .7386849 |
| lage     | .4767192  | .3536276  | 1.35  | 0.178| -.2163781 to 1.169817 |
| visit    | -.2664098 | .1647098  | -1.62 | 0.106| -.5892352 to .0564156 |
| _cons    | 2.099555  | .2203749  | 9.53  | 0.000| 1.667629 to 2.531482  |

| subject  |           |           |       |      |                     |
| var(visit)| .5314803  | .229385   | 2.32  | 0.021| .2280928 to 1.238405 |
| var(_cons)| .2514923  | .0587902  | 4.39  | 0.000| .1590534 to .3976549 |

| subject  |           |           |       |      |                     |
| cov(visit,_cons)| .0028715  | .0887037  | 0.03  | 0.974| -.1709846 to .1767276 |

LR test vs. Poisson model: chi2(3) = 324.54 Prob > chi2 = 0.0000
Note: LR test is conservative and provided only for reference.

In the above, we specified the covariance(unstructured) option to allow correlation between \(u_j\) and \(v_j\), although on the basis of the above output it probably was not necessary—the default independent structure would have sufficed. In the interest of getting more accurate estimates, we also increased the number of quadrature points to nine, although the estimates do not change much when compared with estimates based on the default seven quadrature points.

The essence of the above-fitted model is that after adjusting for other covariates, the log trend in seizures is modeled as a random subject-specific line, with intercept distributed as \(N(\beta_0, \sigma_u^2)\) and slope distributed as \(N(\beta_5, \sigma_v^2)\). From the above output, \(\hat{\beta}_0 = 2.10, \hat{\sigma}_u^2 = 0.25, \hat{\beta}_5 = -0.27,\) and \(\hat{\sigma}_v^2 = 0.53\).

You can predict the random effects \(u_j\) and \(v_j\) by using predict after mepoisson; see [ME] mepoisson postestimation. Better still, you can obtain a predicted number of seizures that takes these random effects into account.
Higher-level models

Example 3: Three- and four-level random-intercept model


`. use https://www.stata-press.com/data/r16/melanoma
(Skin cancer (melanoma) data)
`. describe
Contains data from https://www.stata-press.com/data/r16/melanoma.dta
          obs:     354  Skin cancer (melanoma) data
          vars:    6  30 May 2018 17:10
            (_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,

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

for nation $k$, region $j$, and county $i$. The model includes an exposure term for expected deaths.
. mepoisson deaths uv, exposure(expected) || nation: || region:

Fitting fixed-effects model:
Iteration 0:  log likelihood = -2136.5847
Iteration 1:  log likelihood = -1723.8955
Iteration 2:  log likelihood = -1723.7727
Iteration 3:  log likelihood = -1723.7727

Refining starting values:
Grid node 0:  log likelihood = -1166.6536

Refining starting values (unscaled likelihoods):
Grid node 0:  log likelihood = -1166.6536

Fitting full model:
Iteration 0:  log likelihood = -1166.6536 (not concave)
Iteration 1:  log likelihood = -1152.2741 (not concave)
Iteration 2:  log likelihood = -1146.3094 (not concave)
Iteration 3:  log likelihood = -1119.8479 (not concave)
Iteration 4:  log likelihood = -1108.0129 (not concave)
Iteration 5:  log likelihood = -1098.8067
Iteration 6:  log likelihood = -1095.7563
Iteration 7:  log likelihood = -1095.3164
Iteration 8:  log likelihood = -1095.31
Iteration 9:  log likelihood = -1095.31

Mixed-effects Poisson regression  Number of obs  =  354

<table>
<thead>
<tr>
<th>Group Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Minimum</td>
<td>Average</td>
</tr>
<tr>
<td>nation</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>region</td>
<td>78</td>
<td>1</td>
</tr>
</tbody>
</table>

Integration method: mvaghermite  Integration pts.  =  7
Wald chi2(1)  =  6.12
Log likelihood =  -1095.31  Prob > chi2  =  0.0134

| deaths     | Coef.  | Std. Err.  | z      | P>|z|     | [95% Conf. Interval] |
|------------|--------|------------|--------|--------|----------------------|
| uv         | -.0282041 | .0113998   | -2.47  | 0.013  | -.0505473 -.0058608 |
| _cons      | -.0639672 | .1335515   | -0.48  | 0.632  | -.3257234 .197789   |
| ln(expected) | 1   (exposure) |        |        |        |                      |

| nation      | var(_cons) | .1371732 | .0723303 | .048802 | .3855676 |
| region      | var(_cons) | .0483483 | .0109079 | .0310699 | .0752353 |

LR test vs. Poisson model: chi2(2)  =  1256.93  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. Here the reference population is all nine nations.

Looking at the estimated variance components, we can see 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.
We now add a random intercept for counties nested within regions, making this a four-level model. Because counties also identify the observations, the corresponding variance component can be interpreted as a measure of overdispersion, variability above and beyond that allowed by a Poisson process; see \textit{R nbreg} and \textit{[ME] menbreg}.

.mepoisson deaths uv, exposure(expected) || nation: || region: || county:, > intmethod(mcaghermite)

Fitting fixed-effects model:
Iteration 0:  log likelihood = -2136.5847
Iteration 1:  log likelihood = -1723.8955
Iteration 2:  log likelihood = -1723.7727
Iteration 3:  log likelihood = -1723.7727

Refining starting values:
Grid node 0:  log likelihood = -1379.3466

Refining starting values (unscaled likelihoods):
Grid node 0:  log likelihood = -1379.3466

Fitting full model:
Iteration 0:  log likelihood = -1379.3466 (not concave)
Iteration 1:  log likelihood = -1310.4947 (not concave)
Iteration 2:  log likelihood = -1245.534 (not concave)
Iteration 3:  log likelihood = -1218.5474 (not concave)
Iteration 4:  log likelihood = -1207.881 (not concave)
Iteration 5:  log likelihood = -1122.0585 (not concave)
Iteration 6:  log likelihood = -1092.4049
Iteration 7:  log likelihood = -1088.0486
Iteration 8:  log likelihood = -1086.7175
Iteration 9:  log likelihood = -1086.6756
Iteration 10: log likelihood = -1086.6754
Iteration 11: log likelihood = -1086.6754

Mixed-effects Poisson regression

<table>
<thead>
<tr>
<th>Group Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Minimum  Average   Maximum</td>
</tr>
<tr>
<td>nation</td>
<td>9</td>
<td>3  39.3   95</td>
</tr>
<tr>
<td>region</td>
<td>78</td>
<td>1  4.5     13</td>
</tr>
<tr>
<td>county</td>
<td>354</td>
<td>1  1.0     1</td>
</tr>
</tbody>
</table>
Integration method: mcaghermite Integration pts. = 7
Wald chi2(1) = 8.62
Log likelihood = -1086.6754 Prob > chi2 = 0.0033

|                  | Coef. | Std. Err. | z     | P>|z|   | [95% Conf. Interval] |
|------------------|-------|-----------|-------|-------|----------------------|
| deaths           |       |           |       |       |                      |
| uv               | -0.0334702 | 0.0113968 | -2.94 | 0.003 | -0.0558075 to -0.0111329 |
| _cons            | -0.0864583 | 0.1299275 | -0.67 | 0.506 | -0.3411115 to 0.168195 |
| ln(expected)     | 1     | (exposure)|       |       |                      |
|                  |       |           |       |       |                      |
| nation           |       |           |       |       |                      |
| var(_cons)       | 0.1288627 | 0.0681643 |       |       | 0.0456949 to 0.3634011 |
| region           |       |           |       |       |                      |
| var(_cons)       | 0.0406279 | 0.0105154 |       |       | 0.0244633 to 0.0674735 |
| nation>region    |       |           |       |       |                      |
| var(_cons)       | 0.0146672 | 0.0050979 |       |       | 0.0074215 to 0.0289867 |

LR test vs. Poisson model: chi2(3) = 1274.19 Prob > chi2 = 0.0000
Note: LR test is conservative and provided only for reference.

In the above, we used intmethod(mcaghermite), which is not only faster but also produces estimates that closely agree with those obtained with the default mvaghermite integration method.

### Stored results

mepoisson stores the following in `e()`:

**Scalars**
- `e(N)`: number of observations
- `e(k)`: number of parameters
- `e(k_eq)`: number of dependent variables
- `e(k_eq_model)`: 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_rc)`: number of covariances
- `e(df_m)`: model degrees of freedom
- `e(ll)`: log likelihood
- `e(N_clust)`: number of clusters
- `e(chi2)`: \( \chi^2 \)
- `e(p)`: \( p \)-value for model test
- `e(ll_c)`: log likelihood, comparison model
- `e(chi2_c)`: \( \chi^2 \), comparison test
- `e(df_c)`: degrees of freedom, comparison test
- `e(p_c)`: \( p \)-value for comparison test
- `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
Methods and formulas

mepoisson is a convenience command for meglm with a log link and an poisson family; see [ME] meglm.

In a two-level Poisson model, for cluster $j$, $j = 1, \ldots, M$, the conditional distribution of $Y_j = (y_{j1}, \ldots, y_{jn_j})'$, given a set of cluster-level random effects $u_j$, is
\[
f(y_j | u_j) = \prod_{i=1}^{n_j} \left[ \left\{ \exp(x_{ij}\beta + z_{ij}u_j) \right\}^{y_{ij}} \exp\left\{-\exp(x_{ij}\beta + z_{ij}u_j)\right\} \right] / y_{ij}!
\]

\[
= \exp \left[ \sum_{i=1}^{n_j} \left\{ y_{ij} \left( x_{ij}\beta + z_{ij}u_j \right) - \exp(x_{ij}\beta + z_{ij}u_j) - \log(y_{ij}) \right\} \right]
\]

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

\[
f(y_j | u_j) = \exp \left\{ y_j' (X_j\beta + Z_j u_j) - 1' \exp(X_j\beta + Z_j u_j) - c(y_j) \right\}
\]

where \( X_j \) is formed by stacking the row vectors \( x_{ij} \) and \( Z_j \) is formed by stacking the row vectors \( z_{ij} \). We extend the definition of \( \exp(\cdot) \) to be a vector function where necessary.

Because the prior distribution of \( u_j \) is multivariate normal with mean \( 0 \) and \( q \times q \) variance matrix \( \Sigma \), the likelihood contribution for the \( j \)th cluster is obtained by integrating \( u_j \) out of the joint density \( f(y_j, u_j) \),

\[
\mathcal{L}_j(\beta, \Sigma) = (2\pi)^{-q/2} |\Sigma|^{-1/2} \int f(y_j | u_j) \exp\left(-\frac{1}{2}u_j'\Sigma^{-1}u_j \right) du_j
\]

\[
= \exp\left\{ -c(y_j) \right\} (2\pi)^{-q/2} |\Sigma|^{-1/2} \int \exp\left\{ h(\beta, \Sigma, u_j) \right\} du_j
\]

(2)

where

\[
h(\beta, \Sigma, u_j) = y_j' (X_j\beta + Z_j u_j) - 1' \exp(X_j\beta + Z_j u_j) - \frac{1}{2}u_j'\Sigma^{-1}u_j
\]

and for convenience, in the arguments of \( h(\cdot) \) we suppress the dependence on the observable data \((y_j, X_j, Z_j)\).

The integration in (2) has no closed form and thus must be approximated; see Methods and formulas in [ME] \texttt{meglm} for details.

\texttt{mepoisson} supports multilevel weights and survey data; see Methods and formulas in [ME] \texttt{meglm} for details.

References


Also see

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

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

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

[BAYES] **bayes: mepoisson** — Bayesian multilevel Poisson regression

[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**