meintreg — Multilevel mixed-effects interval regression						
Description	Quick start	Menu Methodo and formulae	Syntax	Options		

# Description

meintreg fits mixed-effects models for continuous responses where the dependent variable may be measured as point data, interval-censored data, left-censored data, or right-censored data. Thus, it is a generalization of the models fit by metobit. The dependent variable must be specified using two variables that indicate how it was measured.

# Quick start

Two-level interval regression on x with random intercepts by lev2 of the interval-measured dependent variable with lower endpoint y\_lower and upper endpoint y\_upper

meintreg y\_lower y\_upper x || lev2:

Same as above, but with random coefficients for x

meintreg y\_lower y\_upper x || lev2: x

- Three-level random-intercept model with lev2 nested within lev3 meintreg y\_lower y\_upper x || lev3: || lev2:
- Crossed-effects model with two-way crossed random effects by factors a and b meintreg y lower y upper x || all:R.a || b:

### Menu

Statistics > Multilevel mixed-effects models > Interval regression

### Syntax

meintreg depvar<sub>lower</sub> depvar<sub>upper</sub> 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 in a crossed-effects model

levelvar: R.varname

The values in *depvar*<sub>lower</sub> and *depvar*<sub>upper</sub> should have the following form:

Type of data		$depvar_{lower}$	<i>depvar</i> <sub>upper</sub>
point data	a = [a,a]	a	a
interval data	[a,b]	a	b
left-censored data	$(-\infty,b]$		b
right-censored data	$[a,+\infty)$	a	•
missing			•

*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			
<u>nocons</u> tant	suppress constant term from the fixed-effects equation include <i>varname</i> in model with coefficient constrained to 1		
offset(varname)			
re_options	Description		
Model			
<pre><u>cov</u>ariance(vartype)</pre>	variance-covariance structure of the random effects		
<u>nocons</u> tant	suppress constant term from the random-effects equation		
<u>fw</u> eight( <i>varname</i> )	frequency weights at higher levels		
<u>iw</u> eight( <i>varname</i> )	importance weights at higher levels		
<u>pweight(varname)</u> sampling weights at higher levels			

options	Description
Model	
<pre><u>const</u>raints(constraints)</pre>	apply specified linear constraints
SE/Robust	
vce( <i>vcetype</i> )	vcetype may be oim, opg, <u>r</u> obust, or <u>cl</u> uster <i>clustvar</i>
Reporting	
<u>l</u> evel(#)	set confidence level; default is level(95)
<u>nocnsr</u> eport	do not display constraints
<u>notab</u> le	suppress coefficient table
<u>nohead</u> er	suppress output header
nogroup	suppress table summarizing groups
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	
<u>intm</u> ethod( <i>intmethod</i> )	integration method
<pre>intpoints(#)</pre>	set the number of integration (quadrature) points for all levels; default is intpoints(7)
Maximization	
maximize_options	control the maximization process; seldom used
<pre>startvalues(symethod)</pre>	method for obtaining starting values
<pre>startgrid[(gridspec)]</pre>	perform a grid search to improve starting values
noestimate	do not fit the model; show starting values instead
dnumerical	use numerical derivative techniques
<u>col</u> linear	keep collinear variables
<u>coefl</u> egend	display legend instead of statistics
vartvne	Description
<u>ina</u> ependent	0; the default unless the R. notation is used
<u>exc</u> hangeable	equal variances for random effects and one common pairwise covariance
<u>id</u> entity	equal variances for random effects and all covariances 0; the default if the R. notation is used
<u>un</u> structured	all variances and covariances to be distinctly estimated
<pre>fixed(matname)</pre>	user-selected variances and covariances constrained to specified values; the remaining variances and covariances unrestricted
<pre>pattern(matname)</pre>	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	
<u>mc</u> aghermite	mode-curvature adaptive Gauss-Hermite quadrature	
ghermite	nonadaptive Gauss-Hermite quadrature	
laplace	Laplacian approximation; the default for crossed random-effects models	

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

*depvar*<sub>lower</sub>, *depvar*<sub>upper</sub>, *indepvars*, and *varlist* may contain time-series operators; see [U] **11.4.4 Time-series varlists**. bayes, by, collect, and svy are allowed; see [U] **11.1.10 Prefix commands**. For more details, see [BAYES] **bayes: mein-**

treg.

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, collinear, and coeflegend do not appear in the dialog box.

See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

### Options

∫ Model ]

noconstant suppresses the constant (intercept) term and may be specified for the fixed-effects equation and for any of or all the random-effects equations.

- offset (varname) specifies that varname 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 (mathame) covariance structure, (co)variance (i, j) is constrained to equal the value specified in the i, jth entry of mathame. In a pattern (mathame) covariance structure, (co)variances (i, j) and (k, l) are constrained to be equal if mathame[i, j] = mathame[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); see [R] Estimation options.

∫ SE/Robust Ì

vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), 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(#), 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.
- 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(#),

<u>nrtol</u>erance(#), <u>nonrtol</u>erance, and from(*init\_specs*); see [R] Maximize. Those that require special mention for meintreg 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 meintreg but are not shown in the dialog box:

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

collinear, coeflegend; see [R] Estimation options.

### **Remarks and examples**

Mixed-effects interval regression is regression for censored data containing both fixed effects and random effects. meintreg fits mixed-effects regression models that account for left-, right-, and intervalcensoring. Thus, it is a generalization of the models fit by metobit. 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.

Interval data arise naturally in many contexts, such as wage data where often you know only that a person's salary is between two values. If one of the interval's endpoints is unknown, the observation is censored. Interval data and right-censored data also arise in the area of survival analysis. meintreg can

fit models for data where each observation represents interval data, left-censored data, right-censored data, or point data. Regardless of the type of observation, the data should be stored in the dataset as interval data; see *Syntax*.

Regardless of the type of censoring, the expected value of the underlying dependent variable—say,  $y^*$ —is modeled using the following linear prediction:

$$E(\mathbf{y}^{\star}|\mathbf{X},\mathbf{u}) = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} \tag{1}$$

**X** is an  $n \times p$  design/covariate matrix, analogous to the covariates you would find in a standard linear regression model, with regression coefficients (fixed effects)  $\beta$ . **Z** is the  $n \times q$  design/covariate matrix for the random effects **u**. This linear prediction also contains the offset when offset() is specified.

The columns of matrix  $\mathbf{Z}$  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-intercepts model,  $\mathbf{Z}$  is simply the scalar 1. The random effects  $\mathbf{u}$  are 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  $\mathbf{Z} = \mathbf{X}$  so that all covariate effects are essentially random and distributed as multivariate normal with mean  $\boldsymbol{\beta}$  and variance  $\boldsymbol{\Sigma}$ .

Below we present a short example of mixed-effects censored regression; refer to [ME] **me** and [ME] **meglm** for additional examples of random-effects models. A two-level interval regression model can also be fit using xtintreg; see [XT] **xtintreg**. In the absence of random effects, mixed-effects censored regression reduces to standard censored regression; see [R] intreg.

#### Example 1: Three-level random-intercept model

Mastitis is a disease affecting dairy cows, consisting of an inflammatory reaction of the udder tissue. Our fictional study was performed on 10 farms using a sample of 10 dairy cows taken from each farm, and time to infection was recorded for each udder quarter for each cow in the sample. The four udder quarters are clustered within the cow, and cows are nested within farms. This is loosely based on nonfictional studies by Goethals et al. (2009) and Elghafghuf et al. (2014).

Cows were examined periodically. Thus, if a cow developed an infection, we do not know the exact day the infection occurred; we only know that it occurred between the last infection-free examination and the first examination where the infection was present. Some udder quarters did not develop an infection by the end of the study, so these observations are right-censored. We include a binary covariate, multiparous, which is equal to 1 for cows that have experienced more than one calving, and 0 for cows with only one calving.

To fit a log-normal model to the data, which assumes that the outcome is always positive, we take the log of our dependent variables and then use meintreg to apply a multilevel Gaussian model for intervaland right-censored data.

```
. use https://www.stata-press.com/data/r19/mastitis
(Simulated data on udder infection of dairy cows)
. generate lnleft = ln(left)
(5 missing values generated)
. generate lnright = ln(right)
(82 missing values generated)
```

. meintreg lnle	eft lnright i	.multiparous	s    farm	:    cow:			
Fitting fixed-	effects model	:					
Iteration 0: 1 Iteration 1: 1 Iteration 2: 1 Iteration 3: 1 Iteration 4: 1	Log likelihoo Log likelihoo Log likelihoo Log likelihoo Log likelihoo	d = -912.930 d = -901.901 d = -901.482 d = -901.481 d = -901.481	005 .84 206 .76 .76				
Refining start:	ing values:						
Grid node 0: 1	Log likelihoo	d = -897.921	.67				
Fitting full mo	odel:						
Iteration 0: 1 Iteration 1: 1 Iteration 2: 1 Iteration 3: 1 Iteration 4: 1 Iteration 5: 1 Iteration 6: 1 Iteration 7: 1 Iteration 8: 1	Log likelihoo Log likelihoo Log likelihoo Log likelihoo Log likelihoo Log likelihoo Log likelihoo Log likelihoo	d = -897.921 $d = -863.20$ $d = -857.453$ $d = -855.181$ $d = -850.843$ $d = -846.319$ $d = -846.244$ $d = -846.244$	.67 (not 033 (not 304 (not 35 525 525 546 126 126	concave) concave) concave)			
Mixed-effects : Groupin	interval regr ng informatio	ession n		Number of Uncens Left-o Right- Interv	f obs sored censored -censore val-cens	= = = = = = = = = = = = = = = = = = =	400 0 5 82 313
Group	variable	No. of groups	Obse: Minimum	rvations p Averag	per grou ge Ma	p ximum	
	farm cow	10 100	40 4	40 4	.0 .0	40 4	
Integration met	thod: mvagher	mite		Integrati	ion pts.	=	7
Log likelihood	= -846.24426			Wald chi2 Prob > ch	2(1) ni2	= =	8.75 0.0031
	Coefficient	Std. err.	z	P> z	[95%	conf.	interval]
1.multiparous _cons	5689113 5.644119	.1923729 .1896383	-2.96 29.76	0.003	9459 5.272	552 435	1918674 6.015803
farm var(_cons)	.0246795	.0258621			.0031	648	.1924544
farm>cow var(_cons)	.2481394	.0497735			.1674	773	.367651
var(e.lnleft)	.2626232	.0257671			.2166	796	.3183084
LR test vs. int	terval model:	chi2(2) = 1	.10.47		Prob >	chi2	= 0.0000

Note: LR test is conservative and provided only for reference.

We see that infection was observed in 318 udder quarters, the 5 observations that are left-censored and the 313 that are interval censored. The coefficient for multiparous is negative, which means that the time to infection will be about 56.9% shorter for cows that experienced multiple calvings.

The within-cow variance is 0.248, and the residual variance is 0.263, while the within-farm variance is smaller, about 0.025. A likelihood-ratio test comparing the model to an interval regression model without random effects is provided under the table and indicates that the three-level interval regression model is preferred.

# **Stored results**

meintreg stores the following in e():

#### Scalars

	e(N)	number of observations
	e(N_unc)	number of uncensored observations
	e(N_lc)	number of left-censored observations
	e(N_rc)	number of right-censored observations
	e(N_int)	number of interval-censored 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(11)	log likelihood
	e(N_clust)	number of clusters
	e(chi2)	$\chi^2$
	e(p)	<i>p</i> -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)	<i>p</i> -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
Mac	ros	
	e(cmd)	meglm
	e(cmd2)	meintreg
	e(cmdline)	command as typed
	e(depvar)	names of dependent variables
	e(wtype)	weight type
	e(wexp)	weight expression (first-level weights)
	e(fweightk)	fweight variable for kth highest level, if specified
	e(iweightk)	iweight variable for kth highest level, if specified
	e(pweightk)	pweight variable for kth highest level, if specified
	e(covariates)	list of covariates
	e(ivars)	grouping variables
	e(model)	interval
	e(title)	title in estimation output
	e(link)	identity
	e(family)	gaussian
	e(clustvar)	name of cluster variable
	e(offset)	offset
	e(intmethod)	integration method
	e(n_quad)	number of integration points
	-	

4

	e(chi2type)	Wald; type of model $\chi^2$
	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
Mat	rices	
	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
Fun	ctions	
	e(sample)	marks estimation sample
	-	*

In addition to the above, the following is stored in r():

Matrices

```
r(table)
```

matrix containing the coefficients with their standard errors, test statistics, *p*-values, and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

### Methods and formulas

Without a loss of generality, consider a two-level regression model

$$E(\mathbf{y}_{j}^{\star}|\mathbf{X}_{j},\mathbf{u}_{j}) = \mathbf{X}_{j}\boldsymbol{\beta} + \mathbf{Z}_{j}\mathbf{u}_{j}, \qquad \mathbf{y}^{\star} \sim \text{normal}$$

for j = 1, ..., M clusters, with the *j*th cluster consisting of  $n_j$  observations, where, for the *j*th cluster,  $\mathbf{y}_j^*$  is the  $n_j \times 1$  response vector,  $\mathbf{X}_j$  is the  $n_j \times p$  matrix of fixed predictors,  $\mathbf{Z}_j$  is the  $n_j \times q$  matrix of random predictors,  $\mathbf{u}_j$  is the  $q \times 1$  vector of random effects, and  $\beta$  is the  $p \times 1$  vector of regression coefficients on the fixed predictors. The random effects,  $\mathbf{u}_j$ , are assumed to be multivariate normal with mean **0** and variance  $\Sigma$ .

Let  $\eta_j$  be the linear predictor,  $\eta_j = \mathbf{X}_j \beta + \mathbf{Z}_j \mathbf{u}_j$ , that also includes the offset variable when offset() is specified. Let  $y_{ij}^{\star}$  and  $\eta_{ij}$  be the *i*th individual elements of  $\mathbf{y}_i^{\star}$  and  $\eta_i$ ,  $i = 1, ..., n_j$ .

The dependent variable,  $y_{ij}$ , is a possibly left-, right-, or interval-censored version of  $y_{ij}^{\star}$ , and it is recorded using two variables.

The conditional density function for the response at observation ij is then,

$$f(y_{ij}^L, y_{ij}^U | \eta_{ij}) = \begin{cases} \left(\sqrt{2\pi}\sigma_\epsilon\right)^{-1} \exp^{-(y_{ij}^L - \eta_{ij})^2 / (2\sigma_\epsilon^2)} & \text{if } (y_{ij}^L, y_{ij}^U) \in C \\ \Phi\left(\frac{y_{ij}^U - \eta_{ij}}{\sigma_\epsilon}\right) & \text{if } (y_{ij}^L, y_{ij}^U) \in L \\ 1 - \Phi\left(\frac{y_{ij}^L - \eta_{ij}}{\sigma_\epsilon}\right) & \text{if } (y_{ij}^L, y_{ij}^U) \in R \\ \Phi\left(\frac{y_{ij}^U - \eta_{ij}}{\sigma_\epsilon}\right) - \Phi\left(\frac{y_{ij}^L - \eta_{ij}}{\sigma_\epsilon}\right) & \text{if } (y_{ij}^L, y_{ij}^U) \in I \end{cases}$$

where C is the set of uncensored observations  $(y_{ij}^L = y_{ij}^U$  and both nonmissing), L is the set of leftcensored observations  $(y_{ij}^L \text{ missing and } y_{ij}^U \text{ nonmissing})$ , R is the set of right-censored observations  $(y_{ij}^L \text{ nonmissing and } y_{ij}^U \text{ missing})$ , I is the set of interval-censored observations  $(y_{ij}^L < y_{ij}^U \text{ and both nonmissing})$ , and  $\Phi(\cdot)$  is the cumulative normal distribution.

Because the observations are assumed to be conditionally independent, the conditional log density function for cluster j is

$$\mathrm{log}f(\mathbf{y}_j|\boldsymbol{\eta}_j) = \sum_{j=1}^{n_i} \ \mathrm{log}f(y_{ij}|\boldsymbol{\eta}_{ij})$$

and the likelihood function for cluster j is given by

$$\begin{split} \mathcal{L}_{j}(\boldsymbol{\beta},\boldsymbol{\Sigma}) &= (2\pi)^{-q/2} |\boldsymbol{\Sigma}|^{-1/2} \int_{\Re^{q}} f(\mathbf{y}_{j} | \boldsymbol{\eta}_{j}) \exp\left(-\frac{1}{2} \mathbf{u}_{j}' \boldsymbol{\Sigma}^{-1} \mathbf{u}_{j}\right) \, d\mathbf{u}_{j} \\ &= (2\pi)^{-q/2} |\boldsymbol{\Sigma}|^{-1/2} \int_{\Re^{q}} \exp\left\{\log f(\mathbf{y}_{j} | \boldsymbol{\eta}_{j}) - \frac{1}{2} \mathbf{u}_{j}' \boldsymbol{\Sigma}^{-1} \mathbf{u}_{j}\right\} \, d\mathbf{u}_{j} \end{split}$$
(2)

where  $\mathfrak{R}$  denotes the set of values on the real line and  $\mathfrak{R}^q$  is the analog in q-dimensional space.

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

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

### References

- Elghafghuf, A., S. Dufour, K. Reyher, I. Dohoo, and H. Stryhn. 2014. Survival analysis of clinical mastitis data using a nested frailty Cox model fit as a mixed-effects Poisson model. *Preventive Veterinary Medicine* 117: 456–468. https: //doi.org/10.1016/j.prevetmed.2014.09.013.
- Goethals, K., B. Ampe, D. Berkvens, H. Laevens, P. Janssen, and L. Duchateau. 2009. Modeling interval-censored, clustered cow udder quarter infection times through the shared gamma frailty model. *Journal of Agricultural, Biological,* and Environmental Statistics 14: 1–14. https://doi.org/10.1198/jabes.2009.0001.

### Also see

- [ME] meintreg postestimation Postestimation tools for meintreg
- [ME] metobit Multilevel mixed-effects tobit regression
- [ME] me Introduction to multilevel mixed-effects models
- [BAYES] **bayes: meintreg** Bayesian multilevel interval regression
- [R] intreg Interval regression
- [SEM] Intro 5 Tour of models (Multilevel mixed-effects models)
- [ST] stintreg Parametric models for interval-censored survival-time data
- [SVY] svy estimation Estimation commands for survey data
- [XT] **xtintreg** Random-effects interval-data regression model
- [U] 20 Estimation and postestimation commands

Stata, Stata Press, Mata, NetCourse, and NetCourseNow are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow is a trademark of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.



For suggested citations, see the FAQ on citing Stata documentation.