

Postestimation commands

Remarks and examples

predict

Methods and formulas

margins

Also see

Postestimation commands

The following postestimation commands are of special interest after `meintreg`:

Command	Description
<code>estat group</code>	summarize the composition of the nested groups
<code>estat icc</code>	estimate intraclass correlations
<code>estat sd</code>	display variance components as standard deviations and correlations

The following standard postestimation commands are also available:

Command	Description
<code>contrast</code>	contrasts and ANOVA-style joint tests of parameters
<code>estat ic</code>	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICC, and BIC, respectively)
<code>estat summarize</code>	summary statistics for the estimation sample
<code>estat vce</code>	variance–covariance matrix of the estimators (VCE)
<code>estat (svy)</code>	postestimation statistics for survey data
<code>estimates</code>	cataloging estimation results
<code>etable</code>	table of estimation results
* <code>hausman</code>	Hausman's specification test
<code>lincom</code>	point estimates, standard errors, testing, and inference for linear combinations of parameters
* <code>lrtest</code>	likelihood-ratio test
<code>margins</code>	marginal means, predictive margins, marginal effects, and average marginal effects
<code>marginsplot</code>	graph the results from margins (profile plots, interaction plots, etc.)
<code>nlcom</code>	point estimates, standard errors, testing, and inference for nonlinear combinations of parameters
<code>predict</code>	means, probabilities, densities, RES, residuals, etc.
<code>predictnl</code>	point estimates, standard errors, testing, and inference for generalized predictions
<code>pwcompare</code>	pairwise comparisons of parameters
<code>test</code>	Wald tests of simple and composite linear hypotheses
<code>testnl</code>	Wald tests of nonlinear hypotheses

* `hausman` and `lrtest` are not appropriate with `svy` estimation results.

predict

Description for predict

predict creates a new variable containing predictions such as linear predictions, standard errors, probabilities, and expected values.

Menu for predict

Statistics > Postestimation

Syntax for predict

Syntax for obtaining predictions of the outcome and other statistics

```
predict [type] { stub* | newvarlist } [if] [in] [ , statistic options ]
```

Syntax for obtaining estimated random effects and their standard errors

```
predict [type] { stub* | newvarlist } [if] [in] , reffects [re_options]
```

Syntax for obtaining ML scores

```
predict [type] { stub* | newvarlist } [if] [in] , scores
```

statistic	Description
Main	
eta	fitted linear predictor; the default
xb	linear predictor for the fixed portion of the model only
stdp	standard error of the fixed-portion linear prediction
pr(a,b)	$\Pr(a < y < b)$
e(a,b)	$E(y \mid a < y < b)$
ystar(a,b)	$E(y^*), y^* = \max\{a, \min(y, b)\}$

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

where *a* and *b* may be numbers or variables; *a* missing (*a* ≥ .) means $-\infty$, and *b* missing (*b* ≥ .) means $+\infty$; see [U] 12.2.1 Missing values.

<i>options</i>	Description
Main	
<u>conditional</u> (<i>ctype</i>)	compute <i>statistic</i> conditional on estimated random effects; default is <code>conditional(ebmeans)</code>
<u>marginal</u>	compute <i>statistic</i> marginally with respect to the random effects
<u>nooffset</u>	make calculation ignoring offset or exposure
Integration	
<i>int_options</i>	integration options
<i>ctype</i>	Description
<u>ebmeans</u>	empirical Bayes means of random effects; the default
<u>ebmodes</u>	empirical Bayes modes of random effects
<u>fixedonly</u>	prediction for the fixed portion of the model only
<i>re_options</i>	Description
Main	
<u>ebmeans</u>	use empirical Bayes means of random effects; the default
<u>ebmodes</u>	use empirical Bayes modes of random effects
<u>reses</u> (<i>stub*</i> <i>newvarlist</i>)	calculate standard errors of empirical Bayes estimates
Integration	
<i>int_options</i>	integration options
<i>int_options</i>	Description
<u>intpoints</u> (#)	use # quadrature points to compute marginal predictions and empirical Bayes means
<u>iterate</u> (#)	set maximum number of iterations in computing statistics involving empirical Bayes estimators
<u>tolerance</u> (#)	set convergence tolerance for computing statistics involving empirical Bayes estimators

Options for predict

Main

`eta`, the default, calculates the fitted linear prediction.

`pr(a, b)` calculates estimates of $\Pr(a < y < b)$, which is the probability that y would be observed in the interval (a, b) .

a and b may be specified as numbers or variable names; lb and ub are variable names;

`pr(20, 30)` calculates $\Pr(20 < y < 30)$;

`pr(lb, ub)` calculates $\Pr(lb < y < ub)$; and

`pr(20, ub)` calculates $\Pr(20 < y < ub)$.

a missing ($a \geq .$) means $-\infty$; `pr(., 30)` calculates $\Pr(-\infty < y < 30)$;

`pr(lb, 30)` calculates $\Pr(-\infty < y < 30)$ in observations for which $lb \geq .$

(and calculates $\Pr(lb < y < 30)$ elsewhere).

b missing ($b \geq .$) means $+\infty$; `pr(20, .)` calculates $\Pr(+\infty > y > 20)$;
`pr(20, ub)` calculates $\Pr(+\infty > y > 20)$ in observations for which $ub \geq .$
(and calculates $\Pr(20 < y < ub)$ elsewhere).

`e(a, b)` calculates estimates of $E(y | a < y < b)$, which is the expected value of y conditional on y being in the interval (a, b) , meaning that y is truncated. a and b are specified as they are for `pr()`.

`ystar(a, b)` calculates estimates of $E(y^*)$, where $y^* = a$ if $y \leq a$, $y^* = b$ if $y \geq b$, and $y^* = y$ otherwise, meaning that y^* is the censored version of y . a and b are specified as they are for `pr()`.

`xb`, `stdp`, `scores`, `conditional()`, `marginal`, and `nooffset`; see [ME] [meglm postestimation](#).

`reffects`, `ebmeans`, `ebmodes`, and `reses()`; see [ME] [meglm postestimation](#).

Integration

`intpoints()`, `iterate()`, `tolerance()`; see [ME] [meglm postestimation](#).

margins

Description for margins

`margins` estimates margins of response for linear predictions, probabilities, and expected values.

Menu for margins

Statistics > Postestimation

Syntax for margins

```
margins [marginlist] [ , options ]  
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]
```

statistic	Description
eta	fitted linear predictor; the default
xb	linear predictor for the fixed portion of the model only
pr(a, b)	$\Pr(a < y < b)$
e(a, b)	$E(y a < y < b)$
ystar(a, b)	$E(y^*), y^* = \max\{a, \min(y, b)\}$
stdp	not allowed with margins

Statistics not allowed with `margins` are functions of stochastic quantities other than `e(b)`.

For the full syntax, see [R] [margins](#).

Remarks and examples

Various predictions, statistics, and diagnostic measures are available after fitting a mixed-effects interval regression model with `meintreg`.

The `predict` command allows us to compute marginal and conditional predictions. Unless stated differently, we use the word “conditional” to mean “conditional on the empirical Bayes predictions of the random effects”. The default prediction is the linear prediction, `eta`, which is the expected value of the unobserved censored variable. Predictions of expected values for censored and truncated versions of the response are also available.

► Example 1: Obtaining conditional and marginal probabilities

In [example 1](#) of [\[ME\] meintreg](#), we fit a three-level mixed-effects interval regression to model log time to udder tissue infection in dairy cows.

```
. 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 lnleft lnright i.multiparous || farm: || cow:
(output omitted)
```

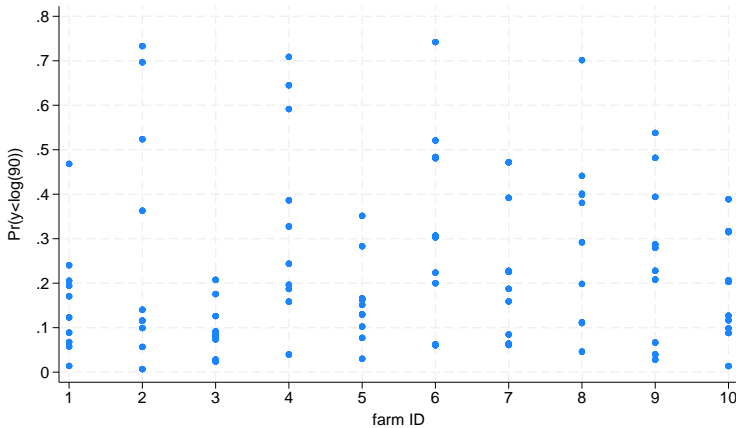
Let’s assume that we want to predict the probability of infection within the first 90 days. Because our dependent variable is $\log(y)$, we need to compute

$$\Pr(0 < y < 90) = \Pr\{-\infty < \log(y) < \log(90)\}$$

We can use the `pr()` option for `predict` to compute the probability that our dependent variable lies in the interval $[-\infty, \log(90)]$.

We first compute the probability conditional on the random effects. Because the lower level on which we are conditioning on is `cow`, and we have only cow-level covariates, these predictions will be constant within cow. We can see that all the predicted probabilities for farm 3 are below 0.21, while the probabilities for farms 2 and 6 reach above 0.70 in some cases.

```
. predict pr_cond, pr(.,log(90))
(predictions based on fixed effects and posterior means of random effects)
. twoway scatter pr_cond farm, ylabel(0(.1).8) xlabel(1(1)10)
```



Now, we compute the marginal probabilities of infection within the first 90 days.

```
. predict pr_marg, pr(.,log(90)) marginal
. tabulate pr_marg multiparous
```

Marginal Pr(y<log(90))	=1 if cows experienced more than one calving, 0 otherwise		Total
	0	1	
.0589298	40	0	40
.2158333	0	360	360
Total	40	360	400

Marginal predictions depend only on the covariate pattern (including covariates in the random-effects part, if present in the model). Because we included only a binary covariate in the model, there are only two predicted values, one for each value of the covariate. We see that the probability of developing an infection in the first 90 days is higher for multiparous cows.

Alternatively, we can use margins to calculate the marginal probabilities. One advantage of using margins is that we can obtain confidence intervals for the probabilities and the difference between them.

```
. margins multiparous, predict(pr(.,log(90)))
```

Adjusted predictions

Number of obs = 400

Model VCE: OIM

Expression: $\Pr(y < \log(90))$, $\text{predict}(\text{pr}(., \log(90)))$

	Delta-method		z	P> z	[95% conf. interval]	
	Margin	std. err.				
multiparous						
0	.0589298	.0305541	1.93	0.054	-.0009551	.1188147
1	.2158333	.0314158	6.87	0.000	.1542595	.2774071

```
. margins, dydx(multiparous) predict(pr(.,log(90)))
```

Conditional marginal effects

Number of obs = 400

Model VCE: OIM

Expression: $\Pr(y < \log(90))$, $\text{predict}(\text{pr}(., \log(90)))$

dy/dx wrt: 1.multiparous

	Delta-method		z	P> z	[95% conf. interval]	
	dy/dx	std. err.				
1.multipar~s	.1569036	.0396889	3.95	0.000	.0791148	.2346923

Note: dy/dx for factor levels is the discrete change from the base level.

The default option for `predict`, `eta`, computes the fitted linear prediction; we can use this option to perform predictions for the uncensored unobserved response. We compute the conditional and marginal predictions for the log time to infection.

```
. predict eta_cond
```

(option `eta` assumed)

(predictions based on fixed effects and posterior means of random effects)

(using 7 quadrature points)

```
. predict eta_marg, marginal
```

(option `eta` assumed)

```
. sort cow
```

```
. list cow multiparous eta_cond eta_marg in 1/8, sepby(cow)
```

	cow	multip~s	eta_cond	eta_marg
1.	1	0	5.486386	5.644119
2.	1	0	5.486386	5.644119
3.	1	0	5.486386	5.644119
4.	1	0	5.486386	5.644119
5.	2	1	5.101668	5.075207
6.	2	1	5.101668	5.075207
7.	2	1	5.101668	5.075207
8.	2	1	5.101668	5.075207

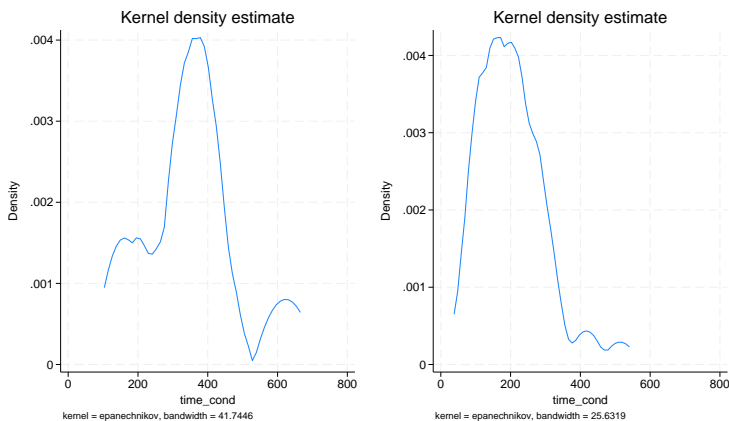
Comparing the conditional and marginal predictions, we see that the predicted log time to infection for the first cow is slightly shorter than the one expected for a cow with this covariate pattern, and the log time to infection for the second cow is slightly longer.

► Example 2: Calculating transformed predictions

Because our dependent variable is log transformed, we might want to compute predictions on the original scale. To do that, we need to obtain predictions for the exponentiated dependent variable.

This exercise is helpful to understand the distribution of the different statistics. If we want to predict the individual conditional time to infection, we need to obtain the conditional mean for $\exp(y)$. Because the conditional distribution of $\exp(y)$ is lognormal with location parameter equal to $\hat{\eta}$ and scale parameter equal to σ_ϵ (residual variance), then its (conditional) expected value is equal to $\exp(\hat{\eta} + \sigma_\epsilon^2/2)$. Here we calculate the conditional time to infection and plot kernel densities for multiparous and uniparous cows.

```
. generate time_cond = exp(eta_cond + _b[/var(e.lnleft)]/2)
. kdensity time_cond if multiparous == 0, xlabel(0(200)800) name(gr1)
. kdensity time_cond if multiparous == 1, xlabel(0(200)800) name(gr2)
. graph combine gr1 gr2
```



The density estimator of the time to infection shows that multiparous cows tend to have shorter times to infection than uniparous cows.

The marginal distribution of y is lognormal with location parameter $\mathbf{x}\beta$ and the scale parameter equal to the marginal variance; see [Methods and formulas](#) of [\[ME\] metobit postestimation](#) for the description of the marginal variance. Thus the marginal expected value of the time to infection is calculated as

```
. predict xb, xb
. generate time_marg = exp( xb + (_b[/var(_cons[farm])]) +
> _b[/var(_cons[farm>cow])] + _b[/var(e.lnleft)]/2)
. tabulate time_marg multiparous
```

time_marg	=1 if cows experienced more than one calving, 0 otherwise		Total
	0	1	
209.1242	0	360	360
369.3851	40	0	40
Total	40	360	400

As before, we see that the unconditional expected value for the time to infection is shorter for multiparous cows.



Methods and formulas

Methods and formulas for predicting random effects and other statistics are given in *Methods and formulas* of [ME] **metobit postestimation**.

Also see

[ME] **meintreg** — Multilevel mixed-effects interval regression

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

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

For suggested citations, see the FAQ on [citing Stata documentation](#).

