

Postestimation commands

Remarks and examples

predict

Methods and formulas

margins

Also see

## Postestimation commands

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

| 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.

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

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: Predicting censored and uncensored means

In [example 1](#) of [\[ME\] metobit](#), we analyzed wages for a subpopulation from the National Longitudinal Survey. The dependent variable is the logarithm of wage, and we fit a model that assumes that the data are right-censored at 1.9.

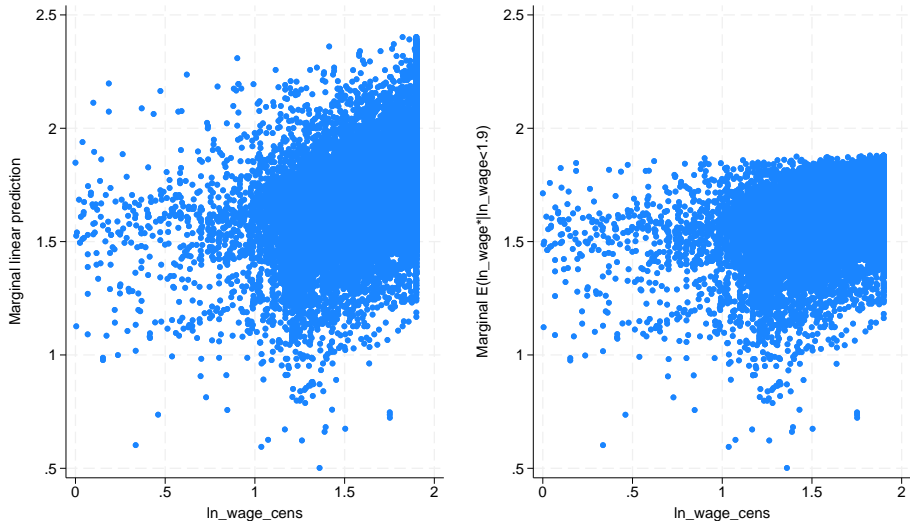
```
. use https://www.stata-press.com/data/r19/nlswork3
(National Longitudinal Survey of Young Women, 14-24 years old in 1968)
. metobit ln_wage union age south##c.grade || idcode:, ul(1.9)
(output omitted)
```

Below, we use `predict` to predict both the mean for the (unobserved) uncensored variable and the (censored) observed values. We also manually generate the censored version of `ln_wage`.

```
. predict uncens_pred, eta marginal
(9310 missing values generated)
. predict cens_pred, ystar(.,1.9) marginal
. generate double ln_wage_cens = min(ln_wage,1.9)
```

To see how the two predictions differ, we can plot them side by side against the censored wage (`ln_wage_cens`).

```
. scatter uncens_pred ln_wage_cens, name(gr1) xsize(4) ysize(4)
. scatter cens_pred ln_wage_cens, name(gr2) xsize(4) ysize(4)
. graph combine gr1 gr2, ycommon
```



We see that many of the predictions for the uncensored variable exceed the censoring point, while the predictions for the censored variable never fall above the upper-censoring limit.



## Methods and formulas

Methods and formulas are presented under the following headings:

- [Introduction](#)
- [Conditional predictions](#)
- [Marginal predictions](#)
- [Marginal variance of the linear predictor](#)

## Introduction

This postestimation entry presents the methods and formulas used to calculate the `pr()`, `e()`, and `ystar()` statistics. See [Methods and formulas](#) of [ME] **estat icc** for a discussion of intraclass correlations. See [Methods and formulas](#) of [ME] **meglm postestimation** for a discussion of the remaining postestimation features.

Recall that in a two-level model, the linear predictor for any  $i$ th observation in the  $j$ th cluster is defined as  $\eta_{ij} = \mathbf{x}_{ij}\boldsymbol{\beta} + \mathbf{z}_{ij}\mathbf{u}_j$ . Let  $\ell_{ij}$  represent a lower bound for  $y_{ij}$  and  $u_{ij}$  represent an upper bound.

## Conditional predictions

The probability that  $y_{ij}|\hat{\eta}_{ij}$  is observed in the interval  $(\ell\ell_{ij}, ul_{ij})$ —the  $\text{pr}(a, b)$  option—is calculated as

$$\text{pr}(\ell\ell_{ij}, ul_{ij}) = \Pr(\ell\ell_{ij} < \hat{\eta}_{ij} + \epsilon_{ij} < ul_{ij}) = \Phi\left(\frac{ul_{ij} - \hat{\eta}_{ij}}{\hat{\sigma}_\epsilon}\right) - \Phi\left(\frac{\ell\ell_{ij} - \hat{\eta}_{ij}}{\hat{\sigma}_\epsilon}\right)$$

where  $\hat{\sigma}_\epsilon$  is the estimated residual standard deviation.

The  $\text{e}(a, b)$  option computes the expected value of  $y_{ij}|\hat{\eta}_{ij}$  conditional on  $y_{ij}|\hat{\eta}_{ij}$  being in the interval  $(\ell\ell_{ij}, ul_{ij})$ , that is, when  $y_{ij}|\hat{\eta}_{ij}$  is truncated. The expected value is calculated as

$$\begin{aligned} \text{e}(\ell\ell_{ij}, ul_{ij}) &= E(\hat{\eta}_{ij} + \epsilon_{ij} \mid \ell\ell_{ij} < \hat{\eta}_{ij} + \epsilon_{ij} < ul_{ij}) \\ &= \hat{\eta}_{ij} - \hat{\sigma}_\epsilon \frac{\phi\left(\frac{ul_{ij} - \hat{\eta}_{ij}}{\hat{\sigma}_\epsilon}\right) - \phi\left(\frac{\ell\ell_{ij} - \hat{\eta}_{ij}}{\hat{\sigma}_\epsilon}\right)}{\Phi\left(\frac{ul_{ij} - \hat{\eta}_{ij}}{\hat{\sigma}_\epsilon}\right) - \Phi\left(\frac{\ell\ell_{ij} - \hat{\eta}_{ij}}{\hat{\sigma}_\epsilon}\right)} \end{aligned}$$

where  $\phi$  is the normal density and  $\Phi$  is the cumulative normal distribution.

You can also compute  $\text{ystar}(a, b)$ —the expected value of  $y_{ij}|\hat{\eta}_{ij}$ , where  $y_{ij}$  is assumed censored at  $\ell\ell_{ij}$  and  $ul_{ij}$ :

$$y_{ij}^* = \begin{cases} \ell\ell_{ij} & \text{if } y_{ij} \leq \ell\ell_{ij} \\ \eta_{ij} + \epsilon_{ij} & \text{if } \ell\ell_{ij} < y_{ij} < ul_{ij} \\ ul_{ij} & \text{if } y_{ij} \geq ul_{ij} \end{cases}$$

This computation can be expressed in several ways, but the most intuitive formulation involves a combination of the two statistics just defined:

$$E(y_{ij}^*) = \text{pr}(-\infty, \ell\ell_{ij})\ell\ell_{ij} + \text{pr}(\ell\ell_{ij}, ul_{ij})\text{e}(\ell\ell_{ij}, ul_{ij}) + \text{pr}(ul_{ij}, +\infty)ul_{ij}$$

## Marginal predictions

When the `marginal` option is specified, the  $\text{pr}()$  statistic is calculated as

$$\text{pr}(\ell\ell_{ij}, ul_{ij}) = \Phi\left(\frac{ul_{ij} - \mathbf{x}_{ij}\hat{\boldsymbol{\beta}}}{\hat{s}_{ij}}\right) - \Phi\left(\frac{\ell\ell_{ij} - \mathbf{x}_{ij}\hat{\boldsymbol{\beta}}}{\hat{s}_{ij}}\right)$$

where  $\hat{s}_{ij}$  is the square root of the estimated marginal variance of the linear predictor, defined in detail below.

The marginal  $\text{e}()$  statistic is calculated as

$$\text{e}(\ell\ell_{ij}, ul_{ij}) = \mathbf{x}_{ij}\hat{\boldsymbol{\beta}} - \hat{s}_{ij} \frac{\phi\left(\frac{ul_{ij} - \mathbf{x}_{ij}\hat{\boldsymbol{\beta}}}{\hat{s}_{ij}}\right) - \phi\left(\frac{\ell\ell_{ij} - \mathbf{x}_{ij}\hat{\boldsymbol{\beta}}}{\hat{s}_{ij}}\right)}{\Phi\left(\frac{ul_{ij} - \mathbf{x}_{ij}\hat{\boldsymbol{\beta}}}{\hat{s}_{ij}}\right) - \Phi\left(\frac{\ell\ell_{ij} - \mathbf{x}_{ij}\hat{\boldsymbol{\beta}}}{\hat{s}_{ij}}\right)}$$

and the marginal `ystar()` statistic is calculated as above with marginal predictions used in place of the conditional ones.

## Marginal variance of the linear predictor

In a two-level model, the marginal variance for observation  $ij$  is given by

$$\sigma_{ij}^2 = \sigma_\epsilon^2 + \mathbf{z}_{ij} \Sigma_2 \mathbf{z}_{ij}'$$

where  $\sigma_\epsilon^2$  is the residual variance at level 1 and  $\Sigma_2$  is the variance matrix of the random effects at level 2. The marginal standard deviation is  $s_{ij} = \sqrt{\sigma_{ij}^2}$ .

In general, for a  $G$ -level random-effects model, the marginal variance for one observation is given by

$$\sigma^2 = \sigma_\epsilon^2 + \sum_{g=2}^G \mathbf{z}_g \Sigma_g \mathbf{z}_g'$$

where  $\mathbf{z}_g$  is a row vector of the covariates at level  $g$  for that observation and  $\Sigma_g$  is the variance matrix of the random effects at level  $g$ .

## Also see

[ME] [metobit](#) — Multilevel mixed-effects tobit regression

[ME] [meglm postestimation](#) — Postestimation tools for meglm

[U] [20 Estimation and postestimation commands](#)

