Postestimation commands

The following postestimation command is of special interest after `meoprobit`:

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

The following standard postestimation commands are also available:

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

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

Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, density and distribution functions, and standard errors.

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
```

<table>
<thead>
<tr>
<th>statistic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pr</td>
<td>predicted probabilities; the default</td>
</tr>
<tr>
<td>eta</td>
<td>fitted linear predictor</td>
</tr>
<tr>
<td>xb</td>
<td>linear predictor for the fixed portion of the model only</td>
</tr>
<tr>
<td>stdp</td>
<td>standard error of the fixed-portion linear prediction</td>
</tr>
<tr>
<td>density</td>
<td>predicted density function</td>
</tr>
<tr>
<td>distribution</td>
<td>predicted distribution function</td>
</tr>
</tbody>
</table>

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

Main
conditional(ctype)  compute statistic conditional on estimated random effects; default is conditional(ebmeans)
marginal  compute statistic marginally with respect to the random effects
nooffset  make calculation ignoring offset or exposure
outcome(outcome)  outcome category for predicted probabilities

Integration
int_options  integration options

You specify one or \( k \) new variables in newvarlist with pr, where \( k \) is the number of outcomes. If you do not specify outcome(), these options assume outcome(#1).

cctype

Description

ebmeans  empirical Bayes means of random effects; the default
ebmodes  empirical Bayes modes of random effects
fixedonly  prediction for the fixed portion of the model only

re_options

Main
ebmeans  use empirical Bayes means of random effects; the default
ebmodes  use empirical Bayes modes of random effects
reses(stub* | newvarlist)  calculate standard errors of empirical Bayes estimates

Integration
int_options  integration options

int_options

intpoints(#)  use \# quadrature points to compute marginal predictions and empirical Bayes means
iterate(#)  set maximum number of iterations in computing statistics involving empirical Bayes estimators
tolerance(#)  set convergence tolerance for computing statistics involving empirical Bayes estimators

Options for predict

pr, the default, calculates the predicted probabilities.

You specify one or \( k \) new variables, where \( k \) is the number of categories of the dependent variable. If you specify the outcome() option, the probabilities will be predicted for the requested outcome only, in which case you specify only one new variable. If you specify one new variable and do not specify outcome(), outcome(#1) is assumed.

eta, xb, stdp, density, distribution, scores, conditional(), marginal, and nooffset; see [ME] meglm postestimation.
outcome(*outcome*) specifies the outcome for which the predicted probabilities are to be calculated. outcome() should contain either one value of the dependent variable or one of #1, #2, ..., with #1 meaning the first category of the dependent variable, #2 meaning the second category, etc.

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 probabilities and linear predictions.

Menu for margins

Statistics > Postestimation

Syntax for margins

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

<table>
<thead>
<tr>
<th>statistic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>probabilities for each outcome</td>
</tr>
<tr>
<td>pr</td>
<td>predicted probabilities for a specified outcome</td>
</tr>
<tr>
<td>eta</td>
<td>fitted linear predictor</td>
</tr>
<tr>
<td>xb</td>
<td>linear predictor for the fixed portion of the model only</td>
</tr>
<tr>
<td>stdp</td>
<td>not allowed with margins</td>
</tr>
<tr>
<td>density</td>
<td>not allowed with margins</td>
</tr>
<tr>
<td>distribution</td>
<td>not allowed with margins</td>
</tr>
<tr>
<td>reffects</td>
<td>not allowed with margins</td>
</tr>
<tr>
<td>scores</td>
<td>not allowed with margins</td>
</tr>
</tbody>
</table>

pr defaults to the first outcome.

Options conditional(ebmeans) and conditional(ebmodes) are not allowed with margins.

Option marginal is assumed where applicable if conditional(fixedonly) is not specified.

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 an ordered probit mixed-effects model using `meoprobit`. Here we show a short example of predicted probabilities and predicted random effects; refer to [ME] `meglm postestimation` for additional examples applicable to mixed-effects generalized linear models.

Example 1: Obtaining predicted probabilities and random effects

In example 2 of [ME] `meoprobit`, we modeled the tobacco and health knowledge (thk) score—coded 1, 2, 3, 4—among students as a function of two treatments (cc and tv) using a three-level ordered probit model with random effects at the school and class levels.

```
. use https://www.stata-press.com/data/r16/tvsfpors
. meoprobit thk prethk cc##tv || school: || class:
(output omitted)
```

We obtain predicted probabilities for all four outcomes based on the contribution of both fixed effects and random effects by typing

```
. predict pr*
   (option pr assumed)
   (predictions based on fixed effects and posterior means of random effects)
   (using 7 quadrature points)
```

As the note says, the predicted values are based on the posterior means of random effects. You can use the `modes` option to obtain predictions based on the posterior modes of random effects.

Because we specified a stub name, Stata saved the predicted random effects in variables `pr1` through `pr4`. Here we list the predicted probabilities for the first two classes for school 515:
For each observation, our best guess for the predicted outcome is the one with the highest predicted probability. For example, for the very first observation in the table above, we would choose outcome 4 as the most likely to occur.

We obtain predictions of the posterior means themselves at the school and class levels by typing

```
. predict re_s re_c, reffects
```
(calculating posterior means of random effects)
(using 7 quadrature points)

Here we list the predicted random effects for the first two classes for school 515:

```
. list class re_s re_c if school==515 & (class==515101 | class==515102), > seby(class)
```

We can see that the predicted random effects at the school level (re_s) are the same for all classes and that the predicted random effects at the class level (re_c) are constant within each class.
Methods and formulas

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

Also see

[ME] meoprobit — Multilevel mixed-effects ordered probit regression
[ME] meglm postestimation — Postestimation tools for meglm
[U] 20 Estimation and postestimation commands