meqrlogit postestimation — Postestimation tools for meqrlogit

Postestimation commands

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

Command	Description
`estat group`	summarize the composition of the nested groups
`estat icc`	estimate intraclass correlations
`estat recovariance`	display the estimated random-effects covariance matrices
`estat sd`	display variance components as standard deviations and correlations

The following standard postestimation commands are also available:

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

predict

Description for predict

predict creates a new variable containing predictions such as mean responses; linear predictions; standard errors; and Pearson, deviance, and Anscombe residuals.

Menu for predict

Statistics > Postestimation

Syntax for predict

Syntax for obtaining estimated random effects and their standard errors

predict [*type*] *newvarsspec* [*if*] [*in*] , reffects [reses(*newvarsspec*)
relevel(*levelvar*)]

Syntax for obtaining other predictions

predict [*type*] *newvar* [*if*] [*in*] [, *statistic* nooffset fixedonly]

newvarsspec is *stub** or *newvarlist*.

statistic	Description
Main	
mu	mean response; the default
xb	linear predictor for the fixed portion of the model only
stdp	standard error of the fixed-portion linear prediction
pearson	Pearson residuals
deviance	deviance residuals
anscombe	Anscombe residuals

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

Options for predict

Main

reffects calculates posterior modal estimates of the random effects. By default, estimates for all random effects in the model are calculated. However, if the relevel(*levelvar*) option is specified, then estimates for only level *levelvar* in the model are calculated. For example, if classes are nested within schools, then typing

```
. predict b*, reffects relevel(school)
```

would yield random-effects estimates at the school level. You must specify q new variables, where q is the number of random-effects terms in the model (or level). However, it is much easier to just specify *stub** and let Stata name the variables *stub*1, *stub*2, ..., *stubq* for you.

`reses(`*newvarsspec*`)` calculates standard errors for the random-effects estimates. By default, standard errors for all random effects in the model are calculated. However, if the `relevel(`*levelvar*`)` option is specified, then standard errors for only level *levelvar* in the model are calculated; see the `reffects` option.

You must specify q new variables, where q is the number of random-effects terms in the model (or level). However, it is much easier to just specify *stub*`*` and let Stata name the variables *stub*`1`, *stub*`2`, ..., *stubq* for you. The new variables will have the same storage type as the corresponding random-effects variables.

The `reffects` and `reses()` options often generate multiple new variables at once. When this occurs, the random effects (or standard errors) contained in the generated variables correspond to the order in which the variance components are listed in the output of `meqrlogit`. Still, examining the variable labels of the generated variables (with the `describe` command, for instance) can be useful in deciphering which variables correspond to which terms in the model.

`relevel(`*levelvar*`)` specifies the level in the model at which predictions for random effects and their standard errors are to be obtained. *levelvar* is the name of the model level and is either the name of the variable describing the grouping at that level or is `_all`, a special designation for a group comprising all the estimation data.

`mu`, the default, calculates the predicted mean. By default, this is based on a linear predictor that includes both the fixed effects and the random effects, and the predicted mean is conditional on the values of the random effects. Use the `fixedonly` option (see below) if you want predictions that include only the fixed portion of the model, that is, if you want random effects set to 0.

`xb` calculates the linear prediction $\mathbf{x}\boldsymbol{\beta}$ based on the estimated fixed effects (coefficients) in the model. This is equivalent to fixing all random effects in the model to their theoretical (prior) mean value of 0.

`stdp` calculates the standard error of the fixed-effects linear predictor $\mathbf{x}\boldsymbol{\beta}$.

`pearson` calculates Pearson residuals. Pearson residuals large in absolute value may indicate a lack of fit. By default, residuals include both the fixed portion and the random portion of the model. The `fixedonly` option modifies the calculation to include the fixed portion only.

`deviance` calculates deviance residuals. Deviance residuals are recommended by McCullagh and Nelder (1989) as having the best properties for examining the goodness of fit of a GLM. They are approximately normally distributed if the model is correctly specified. They may be plotted against the fitted values or against a covariate to inspect the model's fit. By default, residuals include both the fixed portion and the random portion of the model. The `fixedonly` option modifies the calculation to include the fixed portion only.

`anscombe` calculates Anscombe residuals, which are designed to closely follow a normal distribution. By default, residuals include both the fixed portion and the random portion of the model. The `fixedonly` option modifies the calculation to include the fixed portion only.

`nooffset` is relevant only if you specified `offset(`*varname*`)` with `meqrlogit`. It modifies the calculations made by `predict` so that they ignore the offset variable; the linear prediction is treated as $\mathbf{X}\beta + \mathbf{Zu}$ rather than $\mathbf{X}\beta + \mathbf{Zu} + \text{offset}$.

`fixedonly` modifies predictions to include only the fixed portion of the model, equivalent to setting all random effects equal to 0; see the `mu` option.

margins

Description for margins

`margins` estimates margins of response for linear predictions.

Menu for margins

Statistics > Postestimation

Syntax for margins

margins [*marginlist*] [, *options*]

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

statistic	Description
xb	linear predictor for the fixed portion of the model only; the default
reffects	not allowed with margins
mu	not allowed with margins
stdp	not allowed with margins
pearson	not allowed with margins
deviance	not allowed with margins
anscombe	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

stata.com

Various predictions, statistics, and diagnostic measures are available after fitting a logistic mixed-effects model with meqrlogit. For the most part, calculation centers around obtaining estimates of the subject/group-specific random effects. Random effects are not provided as estimates when the model is fit but instead need to be predicted after estimation. Calculation of intraclass correlations, estimating the dependence between latent linear responses for different levels of nesting, may also be of interest.

▷ Example 1: Obtaining predicted probabilities and random effects

In example 3 of [ME] **melogit**, we represented the probability of contraceptive use among Bangladeshi women by using the model (stated with slightly different notation here)

$$\text{logit}(\pi_{ij}) = \beta_0\texttt{rural}_{ij} + \beta_1\texttt{urban}_{ij} + \beta_2\texttt{age}_{ij} + \beta_3\texttt{child1}_{ij} + \beta_4\texttt{child2}_{ij} + \beta_5\texttt{child3}_{ij} + a_j\texttt{rural}_{ij} + b_j\texttt{urban}_{ij}$$

where π_{ij} is the probability of contraceptive use, $j = 1, \dots, 60$ districts, $i = 1, \dots, n_j$ women within each district, and a_j and b_j are normally distributed with mean 0 and variance–covariance matrix

$$\mathbf{\Sigma} = \text{Var}\begin{bmatrix} a_j \\ b_j \end{bmatrix} = \begin{bmatrix} \sigma_a^2 & \sigma_{ab} \\ \sigma_{ab} & \sigma_b^2 \end{bmatrix}$$

The purpose of using this particular model was to allow for district random effects that were specific to the rural and urban areas of that district and that could be interpreted as such. Below we fit the model using `meqrlogit`.

```
. use http://www.stata-press.com/data/r15/bangladesh
(Bangladesh Fertility Survey, 1989)

. generate byte rural = 1 - urban

. meqrlogit c_use rural urban age child*, noconstant || district: rural urban,
> noconstant covariance(unstructured)

Refining starting values:

Iteration 0:   log likelihood = -1208.3924
Iteration 1:   log likelihood = -1204.1317
Iteration 2:   log likelihood = -1200.6012

Performing gradient-based optimization:

Iteration 0:   log likelihood = -1200.6012
Iteration 1:   log likelihood = -1199.3332
Iteration 2:   log likelihood =  -1199.315
Iteration 3:   log likelihood =  -1199.315

Mixed-effects logistic regression               Number of obs     =      1,934
Group variable: district                        Number of groups  =         60

                                                Obs per group:
                                                              min =          2
                                                              avg =       32.2
                                                              max =        118

Integration points =   7                        Wald chi2(6)      =     120.24
Log likelihood =  -1199.315                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
       c_use |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       rural |   -1.71165   .1605618   -10.66   0.000    -2.026345   -1.396954
       urban |  -.8958623   .1704961    -5.25   0.000    -1.230028   -.5616961
         age |   -.026415    .008023    -3.29   0.001    -.0421398   -.0106902
      child1 |    1.13252   .1603285     7.06   0.000      .818282    1.446758
      child2 |   1.357739   .1770522     7.67   0.000     1.010724    1.704755
      child3 |   1.353827   .1828801     7.40   0.000     .9953882    1.712265
------------------------------------------------------------------------------

------------------------------------------------------------------------------
  Random-effects Parameters  |   Estimate   Std. Err.     [95% Conf. Interval]
-----------------------------+------------------------------------------------
district: Unstructured       |
                  var(rural) |   .3897439   .1292459      .2034726    .7465393
                  var(urban) |   .2442965   .1450674      .0762886     .782303
            cov(rural,urban) |  -.0161411   .1057469     -.2234012    .1911189
------------------------------------------------------------------------------
LR test vs. logistic model: chi2(3) = 58.42               Prob > chi2 = 0.0000

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

We can now obtain predicted probabilities, the default prediction:

```
. predict p
(option mu assumed; predicted means)
```

These predictions are based on a linear predictor that includes both the fixed effects and the random effects due to district. Specifying the `fixedonly` option gives predictions that set the random effects to their prior mean of 0. Below we compare both over the first 20 observations:

```
. predict p_fixed, fixedonly
(option mu assumed; predicted means)
. list c_use p p_fixed age child* in 1/20
```

	c_use	p	p_fixed	age	child1	child2	child3
1.	no	.3579543	.4927183	18.44	0	0	1
2.	no	.2134724	.3210403	-5.56	0	0	0
3.	no	.4672256	.6044016	1.44	0	1	0
4.	no	.4206505	.5584864	8.44	0	0	1
5.	no	.2510909	.3687281	-13.56	0	0	0
6.	no	.2412878	.3565185	-11.56	0	0	0
7.	no	.3579543	.4927183	18.44	0	0	1
8.	no	.4992191	.6345999	-3.56	0	0	1
9.	no	.4572049	.594723	-5.56	1	0	0
10.	no	.4662518	.6034657	1.44	0	0	1
11.	yes	.2412878	.3565185	-11.56	0	0	0
12.	no	.2004691	.3040173	-2.56	0	0	0
13.	no	.4506573	.5883407	-4.56	1	0	0
14.	no	.4400747	.5779263	5.44	0	0	1
15.	no	.4794194	.6160359	-0.56	0	0	1
16.	yes	.4465936	.5843561	4.44	0	0	1
17.	no	.2134724	.3210403	-5.56	0	0	0
18.	yes	.4794194	.6160359	-0.56	0	0	1
19.	yes	.4637673	.6010735	-6.56	1	0	0
20.	no	.5001973	.6355067	-3.56	0	1	0

◁

❑ Technical note

Out-of-sample predictions are permitted after `meqrlogit`, but if these predictions involve estimated random effects, the integrity of the estimation data must be preserved. If the estimation data have changed since the model was fit, `predict` will be unable to obtain predicted random effects that are appropriate for the fitted model and will give an error. Thus to obtain out-of-sample predictions that contain random-effects terms, be sure that the data for these predictions are in observations that augment the estimation data.

❑

Methods and formulas

Continuing the discussion in *Methods and formulas* of [ME] **meqrlogit**, and using the definitions and formulas defined there, we begin by considering the prediction of the random effects $\mathbf{u}_j$ for the jth cluster in a two-level model.

Given a set of estimated `meqrlogit` parameters $(\widehat{\boldsymbol\beta}, \widehat{\boldsymbol\Sigma})$, a profile likelihood in $\mathbf{u}_j$ is derived from the joint distribution $f(\mathbf{y}_j, \mathbf{u}_j)$ as

$$\mathcal{L}_j(\mathbf{u}_j) = \exp\{c(\mathbf{y}_j, \mathbf{r}_j)\}(2\pi)^{-q/2}|\widehat{\boldsymbol\Sigma}|^{-1/2}\exp\left\{g\left(\widehat{\boldsymbol\beta}, \widehat{\boldsymbol\Sigma}, \mathbf{u}_j\right)\right\} \tag{1}$$

The conditional maximum likelihood estimator of $\mathbf{u}_j$—conditional on fixed $(\widehat{\boldsymbol\beta}, \widehat{\boldsymbol\Sigma})$—is the maximizer of $\mathcal{L}_j(\mathbf{u}_j)$ or, equivalently, the value of $\widehat{\mathbf{u}}_j$ that solves

$$\mathbf{0} = g'\left(\widehat{\boldsymbol\beta}, \widehat{\boldsymbol\Sigma}, \widehat{\mathbf{u}}_j\right) = \mathbf{Z}_j'\left\{\mathbf{y}_j - \mathbf{m}(\widehat{\boldsymbol\beta}, \widehat{\mathbf{u}}_j)\right\} - \widehat{\boldsymbol\Sigma}^{-1}\widehat{\mathbf{u}}_j$$

Because (1) is proportional to the conditional density $f(\mathbf{u}_j|\mathbf{y}_j)$, you can also refer to $\widehat{\mathbf{u}}_j$ as the conditional mode (or posterior mode if you lean toward Bayesian terminology). Regardless, you are referring to the same estimator.

Conditional standard errors for the estimated random effects are derived from standard theory of maximum likelihood, which dictates that the asymptotic variance matrix of $\widehat{\mathbf{u}}_j$ is the negative inverse of the Hessian, which is estimated as

$$g''\left(\widehat{\boldsymbol\beta}, \widehat{\boldsymbol\Sigma}, \widehat{\mathbf{u}}_j\right) = -\left\{\mathbf{Z}_j'\mathbf{V}(\widehat{\boldsymbol\beta}, \widehat{\mathbf{u}}_j)\mathbf{Z}_j + \widehat{\boldsymbol\Sigma}^{-1}\right\}$$

Similar calculations extend to models with more than one level of random effects; see Pinheiro and Chao (2006).

For any observation i in the jth cluster in a two-level model, define the linear predictor as

$$\widehat{\eta}_{ij} = \mathbf{x}_{ij}\widehat{\boldsymbol\beta} + \mathbf{z}_{ij}\widehat{\mathbf{u}}_j$$

In a three-level model, for the ith observation within the jth level-two cluster within the kth level-three cluster,

$$\widehat{\eta}_{ijk} = \mathbf{x}_{ijk}\widehat{\boldsymbol\beta} + \mathbf{z}_{ijk}^{(3)}\widehat{\mathbf{u}}_k^{(3)} + \mathbf{z}_{ijk}^{(2)}\widehat{\mathbf{u}}_{jk}^{(2)}$$

where $\mathbf{z}^{(p)}$ and $\mathbf{u}^{(p)}$ refer to the level p design variables and random effects, respectively. For models with more than three levels, the definition of $\widehat{\eta}$ extends in the natural way, with only the notation becoming more complicated.

If the `fixedonly` option is specified, $\widehat{\eta}$ contains the linear predictor for only the fixed portion of the model, for example, in a two-level model $\widehat{\eta}_{ij} = \mathbf{x}_{ij}\widehat{\boldsymbol\beta}$. In what follows, we assume a two-level model, with the only necessary modification for multilevel models being the indexing.

The predicted mean conditional on the random effects $\widehat{\mathbf{u}}_j$ is

$$\widehat{\mu}_{ij} = r_{ij}H(\widehat{\eta}_{ij})$$

Pearson residuals are calculated as

$$\nu_{ij}^P = \frac{y_{ij} - \widehat{\mu}_{ij}}{\{V(\widehat{\mu}_{ij})\}^{1/2}}$$

for $V(\widehat{\mu}_{ij}) = \widehat{\mu}_{ij}(1 - \widehat{\mu}_{ij}/r_{ij})$.

Deviance residuals are calculated as

$$\nu_{ij}^D = \text{sign}(y_{ij} - \widehat{\mu}_{ij})\sqrt{\widehat{d}_{ij}^2}$$

where

$$\widehat{d}_{ij}^2 = \begin{cases} 2r_{ij}\log\left(\dfrac{r_{ij}}{r_{ij}-\widehat{\mu}_{ij}}\right) & \text{if } y_{ij}=0 \\ 2y_{ij}\log\left(\dfrac{y_{ij}}{\widehat{\mu}_{ij}}\right) + 2(r_{ij}-y_{ij})\log\left(\dfrac{r_{ij}-y_{ij}}{r_{ij}-\widehat{\mu}_{ij}}\right) & \text{if } 0 < y_{ij} < r_{ij} \\ 2r_{ij}\log\left(\dfrac{r_{ij}}{\widehat{\mu}_{ij}}\right) & \text{if } y_{ij} = r_{ij} \end{cases}$$

Anscombe residuals are calculated as

$$\nu_{ij}^A = \frac{3\left\{y_{ij}^{2/3}\mathcal{H}(y_{ij}/r_{ij}) - \widehat{\mu}^{2/3}\mathcal{H}(\widehat{\mu}_{ij}/r_{ij})\right\}}{2\left(\widehat{\mu}_{ij} - \widehat{\mu}_{ij}^2/r_{ij}\right)^{1/6}}$$

where $\mathcal{H}(t)$ is a specific univariate case of the Hypergeometric2F1 function (Wolfram 2003, 780). For Anscombe residuals for binomial regression, the specific form of the Hypergeometric2F1 function that we require is $\mathcal{H}(t) = {}_2F_1(2/3, 1/3, 5/3, t)$.

For a discussion of the general properties of the above residuals, see Hardin and Hilbe (2018, chap. 4).

References

Hardin, J. W., and J. M. Hilbe. 2018. *Generalized Linear Models and Extensions*. 4th ed. College Station, TX: Stata Press.

McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models*. 2nd ed. London: Chapman & Hall/CRC.

Pinheiro, J. C., and E. C. Chao. 2006. Efficient Laplacian and adaptive Gaussian quadrature algorithms for multilevel generalized linear mixed models. *Journal of Computational and Graphical Statistics* 15: 58–81.

Wolfram, S. 2003. *The Mathematica Book*. 5th ed. Champaign, IL: Wolfram Media.

Also see

[ME] **meqrlogit** — Multilevel mixed-effects logistic regression (QR decomposition)

[U] 20 Estimation and postestimation commands