

**melogit postestimation** — Postestimation tools for melogit

<a href="#">Postestimation commands</a> <a href="#">estat</a> <a href="#">Methods and formulas</a>	<a href="#">predict</a> <a href="#">Remarks and examples</a> <a href="#">Also see</a>	<a href="#">margins</a> <a href="#">Stored results</a>
--	---	---

## Postestimation commands

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

Command	Description
<code>estat group</code>	summarize the composition of the nested groups
<code>estat icc</code>	estimate intraclass correlations

The following standard postestimation commands are also available:

Command	Description
<code>contrast</code>	contrasts and ANOVA-style joint tests of estimates
<code>estat ic</code>	Akaike's and Schwarz's Bayesian information criteria (AIC and BIC)
<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>hausman</code>	Hausman's specification test
<code>lincom</code>	point estimates, standard errors, testing, and inference for linear combinations of coefficients
* <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 coefficients
<code>predict</code>	predictions, residuals, influence statistics, and other diagnostic measures
<code>predictnl</code>	point estimates, standard errors, testing, and inference for generalized predictions
<code>pwcompare</code>	pairwise comparisons of estimates
<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 mean responses; linear predictions; density and distribution functions; standard errors; and Pearson, deviance, and Anscombe residuals.

## Menu for predict

Statistics > Postestimation

## Syntax for predict

*Syntax for obtaining predictions of the outcome and other statistics*

```
predict [type] newvarsspec [if] [in] [, statistic options]
```

*Syntax for obtaining estimated random effects and their standard errors*

```
predict [type] newvarsspec [if] [in], reffects [re_options]
```

*Syntax for obtaining ML scores*

```
predict [type] newvarsspec [if] [in], scores
```

*newvarsspec* is *stub\** or *newvarlist*.

<i>statistic</i>	Description
------------------	-------------

Main

<code>mu</code>	mean response; the default
<code>eta</code>	fitted linear predictor
<code>xb</code>	linear predictor for the fixed portion of the model only
<code>stdp</code>	standard error of the fixed-portion linear prediction
<code><u>density</u></code>	predicted density function
<code><u>distribution</u></code>	predicted distribution function
<code><u>pearson</u></code>	Pearson residuals
<code><u>deviance</u></code>	deviance residuals
<code><u>anscombe</u></code>	Anscombe residuals

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

<i>options</i>	Description
Main	
<code>conditional(<i>ctype</i>)</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
pearson, deviance, anscombe may not be combined with marginal.	
<i>ctype</i>	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
<i>re_options</i>	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(<i>stub*</i>   <i>newvarlist</i>)</code>	calculate standard errors of empirical Bayes estimates
Integration	
<code>int_options</code>	integration options
<i>int_options</i>	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

`mu`, the default, calculates the predicted mean, that is, the probability of a positive outcome.  
`eta`, `xb`, `stdp`, `density`, `distribution`, `pearson`, `deviance`, `anscombe`, `scores`, `conditional()`, `marginal`, and `nooffset`; see [ME] [meglm postestimation](#).  
`reffects`, `ebmeans`, `ebmodes`, and `reses()`; see [ME] [meglm postestimation](#).

### Integration

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

# margins

## Description for margins

`margins` estimates margins of response for mean responses and linear predictions.

## Menu for margins

Statistics > Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
```

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

<i>statistic</i>	Description
<code>mu</code>	mean response; the default
<code>eta</code>	fitted linear predictor
<code>xb</code>	linear predictor for the fixed portion of the model only
<code>stdp</code>	not allowed with <code>margins</code>
<code>density</code>	not allowed with <code>margins</code>
<code>distribution</code>	not allowed with <code>margins</code>
<code>pearson</code>	not allowed with <code>margins</code>
<code>deviance</code>	not allowed with <code>margins</code>
<code>anscombe</code>	not allowed with <code>margins</code>
<code>reflects</code>	not allowed with <code>margins</code>
<code>scores</code>	not allowed with <code>margins</code>

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](#).

## estat

### Description for estat

`estat group` reports the number of groups and minimum, average, and maximum group sizes for each level of the model. Model levels are identified by the corresponding group variable in the data. Because groups are treated as nested, the information in this summary may differ from what you would get if you used the `tabulate` command on each group variable individually.

`estat icc` displays the intraclass correlation for pairs of latent linear responses at each nested level of the model. Intraclass correlations are available for random-intercept models or for random-coefficient models conditional on random-effects covariates being equal to 0. They are not available for crossed-effects models.

### Menu for estat

Statistics > Postestimation

### Syntax for estat

*Summarize the composition of the nested groups*

```
estat group
```

*Estimate intraclass correlations*

```
estat icc [ , level(#) ]
```

### Option for estat icc

`level(#)` specifies the confidence level, as a percentage, for confidence intervals. The default is `level(95)` or as set by `set level`; see [U] 20.7 [Specifying the width of confidence intervals](#).

### Remarks and examples

[stata.com](http://www.stata.com)

Various predictions, statistics, and diagnostic measures are available after fitting a logistic mixed-effects model with `melogit`. Here we show a short example of predicted probabilities and predicted random effects; refer to [ME] [meglm postestimation](#) for additional examples.

#### ▷ Example 1

In [example 2](#) of [ME] [melogit](#), we analyzed the cognitive ability (`dtlm`) of patients with schizophrenia compared with their relatives and control subjects, by using a three-level logistic model with random effects at the family and subject levels. Cognitive ability was measured as the successful completion of the “Tower of London”, a computerized task, measured at three levels of difficulty.

```
. use http://www.stata-press.com/data/r14/towerlondon
(Tower of London data)
. melogit dtlm difficulty i.group || family: || subject: , or
(output omitted)
```

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

```
. predict pr
(predictions based on fixed effects and posterior means of random effects)
(option mu assumed)
(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.

We obtain predictions of the posterior means themselves by typing

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

Because we have one random effect at the family level and another random effect at the subject level, Stata saved the predicted posterior means in the variables `re1` and `re2`, respectively. If you are not sure which prediction corresponds to which level, you can use the `describe` command to show the variable labels.

Here we list the data for family 16:

```
. list family subject dtlm pr re1 re2 if family==16, sepby(subject)
```

	family	subject	dtlm	pr	re1	re2
208.	16	5	1	.5337746	.8045555	.2204122
209.	16	5	0	.1804649	.8045555	.2204122
210.	16	5	0	.0406325	.8045555	.2204122
211.	16	34	1	.8956181	.8045555	1.430945
212.	16	34	1	.6226832	.8045555	1.430945
213.	16	34	1	.2409364	.8045555	1.430945
214.	16	35	0	.6627467	.8045555	-.042955
215.	16	35	1	.2742936	.8045555	-.042955
216.	16	35	0	.0677705	.8045555	-.042955

The predicted random effects at the family level (`re1`) are the same for all members of the family. Similarly, the predicted random effects at the individual level (`re2`) are constant within each individual. The predicted probabilities (`pr`) for this family seem to be in fair agreement with the response (`dtlm`) based on a cutoff of 0.5.

We can use `estat icc` to estimate the residual intraclass correlation (conditional on the difficulty level and the individual's category) between the latent responses of subjects within the same family or between the latent responses of the same subject and family:

```
. estat icc
Residual intraclass correlation
```

Level	ICC	Std. Err.	[95% Conf. Interval]	
family	.1139105	.0997727	.0181851	.4715289
subject family	.3416307	.0889471	.192923	.5297291

`estat icc` reports two intraclass correlations for this three-level nested model. The first is the level-3 intraclass correlation at the family level, the correlation between latent measurements of the

cognitive ability in the same family. The second is the level-2 intraclass correlation at the subject-within-family level, the correlation between the latent measurements of cognitive ability in the same subject and family.

There is not a strong correlation between individual realizations of the latent response, even within the same subject.

◀

## Stored results

`estat icc` stores the following in `r()`:

### Scalars

<code>r(icc#)</code>	level-# intraclass correlation
<code>r(se#)</code>	standard errors of level-# intraclass correlation
<code>r(level)</code>	confidence level of confidence intervals

### Macros

<code>r(label#)</code>	label for level #
------------------------	-------------------

### Matrices

<code>r(ci#)</code>	vector of confidence intervals (lower and upper) for level-# intraclass correlation
---------------------	---

For a  $G$ -level nested model, # can be any integer between 2 and  $G$ .

## Methods and formulas

Methods and formulas are presented under the following headings:

[Prediction](#)

[Intraclass correlations](#)

### Prediction

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

### Intraclass correlations

Consider a simple, two-level random-intercept model, stated in terms of a latent linear response, where only  $y_{ij} = I(y_{ij}^* > 0)$  is observed for the latent variable,

$$y_{ij}^* = \beta + u_j^{(2)} + \epsilon_{ij}^{(1)}$$

with  $i = 1, \dots, n_j$  and level-2 groups  $j = 1, \dots, M$ . Here  $\beta$  is an unknown fixed intercept,  $u_j^{(2)}$  is a level-2 random intercept, and  $\epsilon_{ij}^{(1)}$  is a level-1 error term. Errors are assumed to be logistic with mean 0 and variance  $\sigma_1^2 = \pi^2/3$ ; random intercepts are assumed to be normally distributed with mean 0 and variance  $\sigma_2^2$  and to be independent of error terms.

The intraclass correlation for this model is

$$\rho = \text{Corr}(y_{ij}^*, y_{i'j}^*) = \frac{\sigma_2^2}{\pi^2/3 + \sigma_2^2}$$

It corresponds to the correlation between the latent responses  $i$  and  $i'$  from the same group  $j$ .

Now consider a three-level nested random-intercept model,

$$y_{ijk}^* = \beta + u_{jk}^{(2)} + u_k^{(3)} + \epsilon_{ijk}^{(1)}$$

for measurements  $i = 1, \dots, n_{jk}$  and level-2 groups  $j = 1, \dots, M_{1k}$  nested within level-3 groups  $k = 1, \dots, M_2$ . Here  $u_{jk}^{(2)}$  is a level-2 random intercept,  $u_k^{(3)}$  is a level-3 random intercept, and  $\epsilon_{ijk}^{(1)}$  is a level-1 error term. The error terms have a logistic distribution with mean 0 and variance  $\sigma_1^2 = \pi^2/3$ . The random intercepts are assumed to be normally distributed with mean 0 and variances  $\sigma_2^2$  and  $\sigma_3^2$ , respectively, and to be mutually independent. The error terms are also independent of the random intercepts.

We can consider two types of intraclass correlations for this model. We will refer to them as level-2 and level-3 intraclass correlations. The level-3 intraclass correlation is

$$\rho^{(3)} = \text{Corr}(y_{ijk}^*, y_{i'j'k}^*) = \frac{\sigma_3^2}{\pi^2/3 + \sigma_2^2 + \sigma_3^2}$$

This is the correlation between latent responses  $i$  and  $i'$  from the same level-3 group  $k$  and from different level-2 groups  $j$  and  $j'$ .

The level-2 intraclass correlation is

$$\rho^{(2)} = \text{Corr}(y_{ijk}^*, y_{i'jk}^*) = \frac{\sigma_2^2 + \sigma_3^2}{\pi^2/3 + \sigma_2^2 + \sigma_3^2}$$

This is the correlation between latent responses  $i$  and  $i'$  from the same level-3 group  $k$  and level-2 group  $j$ . (Note that level-1 intraclass correlation is undefined.)

More generally, for a  $G$ -level nested random-intercept model, the  $g$ -level intraclass correlation is defined as

$$\rho^{(g)} = \frac{\sum_{l=g}^G \sigma_l^2}{\pi^2/3 + \sum_{l=2}^G \sigma_l^2}$$

The above formulas also apply in the presence of fixed-effects covariates  $\mathbf{X}$  in a random-effects model. In this case, intraclass correlations are conditional on fixed-effects covariates and are referred to as residual intraclass correlations. `estat icc` also uses the same formulas to compute intraclass correlations for random-coefficient models, assuming 0 baseline values for the random-effects covariates, and labels them as conditional intraclass correlations.

Intraclass correlations will always fall in  $[0,1]$  because variance components are nonnegative. To accommodate the range of an intraclass correlation, we use the logit transformation to obtain confidence intervals. We use the delta method to estimate the standard errors of the intraclass correlations.



Let  $\hat{\rho}^{(g)}$  be a point estimate of the intraclass correlation and  $\widehat{SE}(\hat{\rho}^{(g)})$  be its standard error. The  $(1 - \alpha) \times 100\%$  confidence interval for  $\text{logit}(\rho^{(g)})$  is

$$\text{logit}(\hat{\rho}^{(g)}) \pm z_{\alpha/2} \frac{\widehat{SE}(\hat{\rho}^{(g)})}{\hat{\rho}^{(g)}(1 - \hat{\rho}^{(g)})}$$

where  $z_{\alpha/2}$  is the  $1 - \alpha/2$  quantile of the standard normal distribution and  $\text{logit}(x) = \ln\{x/(1-x)\}$ . Let  $k_u$  be the upper endpoint of this interval, and let  $k_l$  be the lower. The  $(1 - \alpha) \times 100\%$  confidence interval for  $\rho^{(g)}$  is then given by

$$\left( \frac{1}{1 + e^{-k_l}}, \frac{1}{1 + e^{-k_u}} \right)$$

## Also see

[ME] [melogit](#) — Multilevel mixed-effects logistic regression

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

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