

cmprobit postestimation — Postestimation tools for cmprobit

Postestimation commands
Remarks and examples

predict
Also see

margins estat

Postestimation commands

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

Command	Description
<code>estat covariance</code>	covariance matrix of the utility errors for the alternatives
<code>estat correlation</code>	correlation matrix of the utility errors for the alternatives
<code>estat facweights</code>	covariance factor weights matrix

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>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>	adjusted predictions, predictive margins, and 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>	probabilities, etc.
<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

predict

Description for predict

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

Menu for predict

Statistics > Postestimation

Syntax for predict

```
predict [type] newvar [if] [in] [, statistic]
```

```
predict [type] stub* [if] [in], scores
```

<i>statistic</i>	Description
Main	
<code>pr</code>	probability of each ranking, by case; the default
<code>pr1</code>	probability alternative is preferred
<code>xb</code>	linear prediction
<code>stdp</code>	standard error of the linear prediction

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

`predict` omits missing values casewise if `cmroprobit` used casewise deletion (the default); if `cmroprobit` used alternativewise deletion (option `altwise`), `predict` uses alternativewise deletion.

Options for predict

Main

`pr`, the default, calculates the probability of each ranking. For each case, one probability is computed for the ranks in `e(depvar)`.

`pr1` calculates the probability that each alternative is preferred.

`xb` calculates the linear prediction $\mathbf{x}_{ij}\beta + \mathbf{z}_i\alpha_j$ for alternative j and case i .

`stdp` calculates the standard error of the linear prediction.

`scores` calculates the scores for each coefficient in `e(b)`. This option requires a new variable list of length equal to the number of columns in `e(b)`. Otherwise, use the `stub*` syntax to have `predict` generate enumerated variables with prefix `stub`.

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]
```

<i>statistic</i>	Description
<code>pr</code>	not allowed with <code>margins</code>
<code>pr1</code>	probability alternative is preferred; the default
<code>xb</code>	linear prediction
<code>stdp</code>	not allowed with <code>margins</code>
<code>scores</code>	not allowed with <code>margins</code>

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

For more details, see [CM] [margins](#).

estat

Description for estat

`estat covariance` computes the estimated variance–covariance matrix of the utility (latent-variable) errors for the alternatives. The estimates are displayed, and the variance–covariance matrix is stored in `r(cov)`.

`estat correlation` computes the estimated correlation matrix of the utility (latent-variable) errors for the alternatives. The estimates are displayed, and the correlation matrix is stored in `r(cor)`.

`estat facweights` displays the covariance factor weights matrix and stores it in `r(C)`.

Menu for estat

Statistics > Postestimation

Syntax for estat

Covariance matrix of the utility errors for the alternatives

```
estat covariance [ , format(%fmt) border(bspec) left(#) ]
```

Correlation matrix of the utility errors for the alternatives

```
estat correlation [ , format(%fmt) border(bspec) left(#) ]
```

Covariance factor weights matrix

```
estat facweights [ , format(%fmt) border(bspec) left(#) ]
```

`collect` is allowed with all `estat` commands; see [\[U\] 11.1.10 Prefix commands](#).

Options for estat covariance, estat correlation, and estat facweights

`format(%fmt)` sets the matrix display format. The default for `estat covariance` and `estat facweights` is `format(%9.0g)`; the default for `estat correlation` is `format(%9.4f)`.

`border(bspec)` sets the matrix display border style. The default is `border(all)`. See [\[P\] matlist](#).

`left(#)` sets the matrix display left indent. The default is `left(2)`. See [\[P\] matlist](#).

Remarks and examples

stata.com

After fitting a rank-ordered probit choice model, you can use `predict` to obtain the probabilities of the observed rankings of the alternatives or the probabilities of each alternative being preferred.

When evaluating the multivariate normal probabilities via Monte Carlo, `predict` uses the same method to generate the random sequence of numbers as the previous call to `cmroprobit`. For example, if you specified `intmethod(halton)` when fitting the model, `predict` also uses Halton sequences.

In [example 1](#) of [\[CM\] cmroprobit](#), we fit a model of job characteristic preferences. This is a study of Wisconsin high school graduates who were asked to rate their relative preference of four job characteristics: esteem, variety, autonomy, and security. The alternatives are ranked such that 1 is the preferred alternative and 4 is the least preferred.

The case-specific covariates are `gender`, `female`, an indicator variable for females, and `score`, a score on a general mental ability test measured in standard deviations. From the variables `high` and `low`, we create an alternative-specific variable, `currentjob`, that indicates whether the respondent's current job is high, low, or neither in esteem, variety, autonomy, or security.

We load the data and `cmset` them. For speed of running this example, we keep only untied rankings. Then, we fit our `cmroprobit` model.

```
. use https://www.stata-press.com/data/r17/wlsrank
(1992 Wisconsin Longitudinal Study data on job values)
. cmset id jobchar
      Case ID variable: id
      Alternatives variable: jobchar
. keep if noties
(11,244 observations deleted)
. generate currentjob = 1 if low==1
(1,304 missing values generated)
. replace currentjob = 2 if low==0 & high==0
(805 real changes made)
. replace currentjob = 3 if high==1
(499 real changes made)
. label define current 1 "Low" 2 "Neither" 3 "High"
. label values currentjob current
. cmroprobit rank i.currentjob, casevars(i.female score) reverse
note: variable 2.currentjob has 69 cases that are not alternative-specific;
      there is no within-case variability.
note: variable 3.currentjob has 107 cases that are not alternative-specific;
      there is no within-case variability.

Iteration 0:  log simulated-likelihood = -1102.9667
Iteration 1:  log simulated-likelihood = -1089.1146 (backed up)
Iteration 2:  log simulated-likelihood = -1085.7877 (backed up)
Iteration 3:  log simulated-likelihood = -1083.0085 (backed up)
Iteration 4:  log simulated-likelihood = -1082.5081 (backed up)
Iteration 5:  log simulated-likelihood = -1082.1977 (backed up)
Iteration 6:  log simulated-likelihood = -1082.1208 (backed up)
Iteration 7:  log simulated-likelihood = -1082.0995
Iteration 8:  log simulated-likelihood = -1082.0442
Iteration 9:  log simulated-likelihood = -1081.8316 (backed up)
Iteration 10: log simulated-likelihood = -1081.6816
Iteration 11: log simulated-likelihood = -1081.5777
Iteration 12: log simulated-likelihood = -1081.5137
Iteration 13: log simulated-likelihood = -1081.1495
Iteration 14: log simulated-likelihood = -1081.0503
Iteration 15: log simulated-likelihood = -1080.7247
```


We obtain the probabilities of the observed alternative rankings using `predict` with the `pr` option. The probabilities of each alternative being preferred is given by the `pr1` option.

```
. predict prob, pr
. predict prob1, pr1
. list id jobchar rank prob prob1 in 1/12, sepby(id)
```

	id	jobchar	rank	prob	prob1
1.	13	Esteem	4	.0424396	.0159974
2.	13	Variety	2	.0424396	.6024934
3.	13	Autonomy	1	.0424396	.1029332
4.	13	Security	3	.0424396	.278576
5.	19	Esteem	3	.0942127	.0140184
6.	19	Variety	2	.0942127	.4026075
7.	19	Autonomy	4	.0942127	.1232482
8.	19	Security	1	.0942127	.4601093
9.	22	Esteem	4	.1416861	.0255156
10.	22	Variety	1	.1416861	.455048
11.	22	Autonomy	2	.1416861	.2565435
12.	22	Security	3	.1416861	.2629159

The `prob` variable is constant for each case because it contains the probability of the observed ranking in the `rank` variable. The `prob1` variable contains the estimated probability of each alternative being preferred. The sum of the values in `prob1` will be approximately 1 for each case. They do not add up to exactly 1 because of approximations due to the GHK algorithm.

For examples of the specialized `estat` subcommands `covariance` and `correlation`, see [\[CM\] Intro 6](#) and [\[CM\] cmroprobit](#).

Also see

[\[CM\] cmroprobit](#) — Rank-ordered probit choice model

[\[CM\] cmmprobit](#) — Multinomial probit choice model

[\[CM\] cmmprobit postestimation](#) — Postestimation tools for cmmprobit

[\[CM\] margins](#) — Adjusted predictions, predictive margins, and marginal effects

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