

**cmclogit postestimation** — Postestimation tools for `cmclogit`

[Postestimation commands](#)   [predict](#)   [margins](#)   [Remarks and examples](#)  
Also see

**Postestimation commands**

The following postestimation commands are available after `cmclogit`:

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

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

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

---

Main

|                   |  |
|-------------------|--|
| <code>pr</code>   | probability alternative is chosen; the default |
| <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 `cmlogit` used casewise deletion (the default); if `cmlogit` used alternativewise deletion (option `altwise`), `predict` uses alternativewise deletion.

### Options for predict

Main

`pr`, the default, calculates the probability of choosing each alternative.

`xb` calculates the linear prediction.

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

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

`scores` calculates the scores for each coefficient in  $\mathbf{e}(\mathbf{b})$ . This option requires a new variable list of length equal to the number of columns in  $\mathbf{e}(\mathbf{b})$ . 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>     | probability alternative is chosen; 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](#).

### Remarks and examples

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

Remarks are presented under the following headings:

*Testing coefficient estimates*

*Predicted probabilities*

*Casewise versus alternativewise sample selection*

*Obtaining estimation statistics for the alternatives*

### Testing coefficient estimates

The output of `cmclogit` is displayed and stored as a multiple-equation model. Let's illustrate this with [example 1](#) of [CM] [cmclogit](#).

#### ► Example 1

We load the data, `cmset` the data, and run `cmclogit`.

```
. use https://www.stata-press.com/data/r17/carchoice
(Car choice data)
. cmset consumerid car
note: alternatives are unbalanced across choice sets; choice sets of different
      sizes found.
      Case ID variable: consumerid
      Alternatives variable: car
. cmclogit purchase dealers, casevars(i.gender income)
(output omitted)
```

The coefficient estimates for `i.gender` and `income` are stored under the equation names `Japanese`, `European`, and `Korean`, that is, the names of the alternatives, except for the base alternative `American`. To test whether the coefficient estimates for `i.gender` are the same for the Japanese and Korean alternatives relative to the American base alternative, we type

```
. test [Japanese]:1.gender = [Korean]:1.gender
( 1) [Japanese]1.gender - [Korean]1.gender = 0
      chi2( 1) =      1.00
      Prob > chi2 =    0.3169
```

The following shorthand syntax is useful for testing across the alternatives:

```
. test [Japanese = European = Korean]:1.gender
( 1) [Japanese]1.gender - [European]1.gender = 0
( 2) [Japanese]1.gender - [Korean]1.gender = 0
      chi2( 2) =    15.62
      Prob > chi2 =    0.0004
```

See [\[R\] test](#) for details.

◀

## Predicted probabilities

After running `cmclgfit`, you can use `predict` to obtain the estimated probability that each alternative is chosen for each case conditional on its observed data.

### ▶ Example 2

Continuing with the [previous example](#), we calculate predicted probabilities and list them for the first four cases:

```
. predict p
(option pr assumed; Pr(car))
. format p %6.3f
. list consumerid car purchase gender income p
> if consumerid <= 4, sepby(consumerid) abbr(10)
```

|     | consumerid | car      | purchase | gender | income | p     |
|-----|------------|----------|----------|--------|--------|-------|
| 1.  | 1          | American | 1        | Male   | 46.7   | 0.391 |
| 2.  | 1          | Japanese | 0        | Male   | 46.7   | 0.374 |
| 3.  | 1          | European | 0        | Male   | 46.7   | 0.183 |
| 4.  | 1          | Korean   | 0        | Male   | 46.7   | 0.053 |
| 5.  | 2          | American | 1        | Male   | 26.1   | 0.493 |
| 6.  | 2          | Japanese | 0        | Male   | 26.1   | 0.274 |
| 7.  | 2          | European | 0        | Male   | 26.1   | 0.095 |
| 8.  | 2          | Korean   | 0        | Male   | 26.1   | 0.138 |
| 9.  | 3          | American | 0        | Male   | 32.7   | 0.524 |
| 10. | 3          | Japanese | 1        | Male   | 32.7   | 0.337 |
| 11. | 3          | European | 0        | Male   | 32.7   | 0.138 |
| 12. | 4          | American | 1        | Female | 49.2   | 0.391 |
| 13. | 4          | Japanese | 0        | Female | 49.2   | 0.496 |
| 14. | 4          | European | 0        | Female | 49.2   | 0.113 |

To get predicted probabilities and marginal effects averaged across the sample or for hypothetical cases (that is, predictor values set to particular values), use the `margins` postestimation command; see [CM] [Intro 1](#) and [CM] [margins](#) for more information and examples.



### Casewise versus alternativewise sample selection

Missing values in CM data are handled in two possible ways: casewise deletion (the default) and alternativewise (`altwise`) deletion. Casewise deletion omits the whole case whenever any observation within the case has a missing value. Alternativewise deletion omits only the observations with missing values.

`predict` uses whatever was used with `cmlogit`. If `cmlogit` used casewise deletion, `predict` uses casewise deletion. If `cmlogit` was used with the option `altwise`, `predict` uses alternativewise deletion. Should you wish to select the sample for `predict` yourself, you can use an `if` restriction with `predict`.

See [example 3](#) in [CM] [cmlogit](#) for more on casewise versus alternativewise deletion.

### Obtaining estimation statistics for the alternatives

#### ▷ Example 3

`cmtab` can be used to obtain a table of the alternatives for the estimation sample. If there are missing values in the data used to fit the model, you will need to restrict `cmtab` to the estimation sample by specifying `if e(sample)`.

```
. cmtab if e(sample), choice(purchase)
Tabulation of chosen alternatives (purchase = 1)
```

| Nationality<br>of car | Freq. | Percent | Cum.   |
|-----------------------|-------|---------|--------|
| American              | 376   | 43.62   | 43.62  |
| Japanese              | 316   | 36.66   | 80.28  |
| European              | 130   | 15.08   | 95.36  |
| Korean                | 40    | 4.64    | 100.00 |
| Total                 | 862   | 100.00  |        |

`cmchoiceset` is useful for obtaining a table of the choice-set patterns.

```
. cmchoiceset if e(sample)
Tabulation of choice-set possibilities
```

| Choice set | Freq. | Percent | Cum.   |
|------------|-------|---------|--------|
| 1 2 3      | 373   | 43.27   | 43.27  |
| 1 2 3 4    | 489   | 56.73   | 100.00 |
| Total      | 862   | 100.00  |        |

Note: Total is number of cases.

If you have missing data or see notes mentioning cases being dropped, `cmsample` can identify omitted observations and show the reason they were omitted from the estimation sample. See [CM] [cmsample](#).



## Also see

[CM] [cmlogit](#) — Conditional logit (McFadden's) choice model

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

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