Title

stata.com

predict advanced — predict's advanced features

Description

`predict`'s features are documented in

[ERM] **eregress predict**

[ERM] **eintreg predict**

[ERM] **eprobit predict**

[ERM] **eoprobit predict**

[ERM] **predict treatment**

Here, we document `predict`'s advanced features.

Syntax

`predict` [*type*] *newvar* [*if*] [*in*] [`,` *statistic treatstatistic asfmethod counterfactual treatmodifier oprobitmodifier advanced*]

In some cases, more than one new variable needs to be specified:

`predict` [*type*] {*stub*`*`|*newvarlist*} [*if*] [*in*] [`,` *statistic treatstatistic asfmethod counterfactual treatmodifier oprobitmodifier advanced*]

With the exception of *advanced*, you have seen this syntax in the other `predict` manual entries. We will not cover old ground.

advanced	Description
Main	
`equation(`*depvar*`)`	calculate results for specified dependent variable
`nooffset`	ignore option `offset()` specified when model was fit in making calculation
`pr(`*a*`,` *b*`)`	calculate $\Pr(a < \mathbf{x}_i\boldsymbol{\beta} + e_i.depvar < b)$; a and b are numbers or variable names
`e(`*a*`,` *b*`)`	calculate $E(y_i \mid a < y_i < b)$, where $y_i = \mathbf{x}_i\boldsymbol{\beta} + e_i.depvar$; a and b are numbers or variable names
`fix(`*endogvars*`)`	fix specified endogenous covariates
`base(`*valspecs*`)`	specify base values of any variables
`scores`	calculate equation-level score variables for cross-sectional models and parameter-level score variables for panel-data models

base() and fix() exist so margins can manipulate variables to form counterfactuals. You should not use them to calculate predictions. Use target() if you want to compute counterfactuals using predict.

base() and fix() imply option noasf.

endogvars are names of one or more endogenous variables appearing in the main equation.

valspecs specify the values for variables at which predictions are to be evaluated. Each *valspec* is of the form

varname = #

varname = (*exp*)

varname = *othervarname*

For instance, base(*valspecs*) could be base(w1=0) or base(w1=0 w2=1).

Also note that even though option mean was not included in *treatstatistic* for eprobit, eoprobit, xteprobit, and xteoprobit, it is allowed with them. mean returns the probability of a positive outcome after eprobit and xteprobit and returns the expected value of the outcome after eoprobit and xteoprobit.

Options

Main

equation(*depvar*) specifies the dependent variable for which predictions are to be calculated. By default, predictions are made for the dependent variable of the main equation.

nooffset is relevant only if you specified offset() when you fit the model. It modifies the calculations made by predict so that they ignore the offset variable.

pr(*a*, *b*) calculates $\Pr(a < \mathbf{x}_i\boldsymbol{\beta} + e_i.depvar < b)$, the probability that the linear prediction is between *a* and *b*.

a and *b* may be specified as numbers or variable names. If *a* is missing ($a \geq .$), then *a* is treated as $-\infty$. If *b* is missing ($b \geq .$), then *b* is treated as $+\infty$.

e(*a*, *b*) calculates $E(y_i|a < y_i < b)$, where $y_i = \mathbf{x}_i\boldsymbol{\beta} + e_i.depvar$. This is the linear prediction conditional on the outcome being between *a* and *b*.

a and *b* may be specified as numbers or variable names. If *a* is missing ($a \geq .$), then *a* is treated as $-\infty$. If *b* is missing ($b \geq .$), then *b* is treated as $+\infty$.

fix(*endogvars*) is an advanced option seldom used for prediction. If you want to specify counterfactuals, you should use target(). fix() specifies a list of endogenous variables from the main equation to be treated as if they were exogenous. This was discussed in [ERM] **Intro 7**.

base(*valspecs*) is an advanced option seldom used for prediction. If you want to specify counterfactuals, you should use target(). base() specifies a list of variables from any equation and values for them. Those values will be used in calculating the expected value of e_i.y (or e_{ij}.y in the panel case). Errors from other equations spill over into the main equation because of correlations between errors. The correlations were estimated when the model was fit. The amount of spillover depends on those correlations and the values of the errors. This issue was discussed in [ERM] **Intro 7**.

scores calculates equation-level scores for cross-sectional models (eintreg, eoprobit, eprobit, and eregress) and parameter-level scores for panel-data models (xteintreg, xteoprobit, xteprobit, and xteregress).

Remarks and examples

stata.com

The most important of the advanced features is the `equation()` option. Previously, we documented that `predict` calculates results for the main equation only. That was not true. The `equation()` option can be used to target the other equations. The `equation()` option is important because it can apply so many of `predict`'s features to them.

ERMs provide three types of equations. The `endogenous()` option names two of them and leaves the other unnamed:

```
endogenous(..., none specified ...)
endogenous(..., probit ...)
endogenous(..., oprobit ...)
```

none specified should have been called `linear`. Meanwhile, `entreat()` adds `probit` or `oprobit` equations, `select()` adds `probit` equations, and `tobitselect()` adds `linear` equations. Thus, there are three types of equations in total: `linear`, `probit`, and `oprobit`.

`equation()` can be used to provide the following `predict` features with the other equations in the model:

Option	Description
Linear equations	
`mean`	linear prediction
`xb`	linear prediction excluding complications
`ystar()`	censored prediction
`e()`	constrained expected value
`pr()`	probability in range
`expmean`	mean of exponentiated outcome
Probit equations	
`xb`	linear prediction excluding complications
`pr`	probability of positive outcome
`mean`	synonym for `pr`
Ordered probit equations	
`xb`	linear prediction excluding complications
`pr`	probability of each outcome
`mean`	expected value of outcome

Note 1: Option `outlevel(#)` is used with `pr` in `oprobit` equations to restrict the calculation to the specified outcome.

Note 2: When `equation(`*depvar*`)` is the main equation, you can use any of `predict`'s options.

Note 3: For the main equation, options `e()` and `pr()` can be used with *howcalculated* options `fix()`, `base()`, and `target()`.

Options not allowed with `equation()` include `predict`'s treatment options as well as `fix()`, `base()`, and `target()`.

For an example of `predict` with the `equation()` option, see [ERM] **Example 6b**.

Methods and formulas

See *Methods and formulas* of [ERM] **eprobit postestimation**.

Also see

[ERM] **eintreg postestimation** — Postestimation tools for eintreg and xteintreg

[ERM] **eintreg predict** — predict after eintreg and xteintreg

[ERM] **eoprobit postestimation** — Postestimation tools for eoprobit and xteoprobit

[ERM] **eoprobit predict** — predict after eoprobit and xteoprobit

[ERM] **eprobit postestimation** — Postestimation tools for eprobit and xteprobit

[ERM] **eprobit predict** — predict after eprobit and xteprobit

[ERM] **eregress postestimation** — Postestimation tools for eregress and xteregress

[ERM] **eregress predict** — predict after eregress and xteregress