

eintreg predict — predict after eintreg

- [Description](#)
[Options for statistics](#)
[Remarks and examples](#)
[Also see](#)
- [Syntax](#)
[Options for how results are calculated](#)
[Methods and formulas](#)

Description

In this entry, we show how to create new variables containing observation-by-observation predictions after fitting a model with `eintreg`.

Syntax

You previously fit the model

```
eintreg y1 yu x1 ... , ...
```

The equation specified immediately after the `eintreg` command is called the main equation. It is

$$y_i = \beta_0 + \beta_1 x_{1i} + \dots + e_i \cdot y$$

where $y_{1i} \leq y_i \leq y_{ui}$.

`predict` calculates predictions for `y` in the main equation. The other equations in the model are called auxiliary equations or complications.

The syntax of `predict` is

```
predict [type] newvar [if] [in] [, stdstatistics howcalculated]
```

<i>stdstatistics</i>	Description
<code>mean</code>	linear prediction; the default
<code>xb</code>	linear prediction excluding all complications
<code>ystar(a,b)</code>	$E(y^*_{j}), y^*_{j} = \max\{a, \min(y_j, b)\}$

a and *b* are numeric values, missing (`.`), or variable names.

<i>howcalculated</i>	Description
default	not fixed; base values from data
<code>fix(endogvars)</code>	fix specified endogenous covariates
<code>base(valspecs)</code>	specify base values of any variables
<code>target(valspecs)</code>	more convenient way to specify <code>fix()</code> and <code>base()</code>

Note: The `fix()` and `base()` options affect results only in models with endogenous variables in the main equation. The `target()` option is sometimes a more convenient way to specify the `fix()` and `base()` options.

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)`.

Notes:

- (1) `predict` can also calculate treatment-effect statistics. See [\[ERM\] `predict treatment`](#).
- (2) `predict` can also make predictions for the other equations in addition to the main-equation predictions discussed here. See [\[ERM\] `predict advanced`](#).

Options for statistics

`mean` specifies that the linear prediction be calculated. In each observation, the linear prediction is the expected value of the dependent variable y conditioned on the covariates. Results depend on how complications are handled, which is determined by the *howcalculated* options.

`xb` specifies that the linear prediction be calculated ignoring all complications. This prediction corresponds to what would be observed in data in which all the covariates in the main equation were exogenous.

`ystar(a, b)` specifies that the linear prediction be censored between a and b . If a is missing (`.`), then a is treated as $-\infty$. If b is missing (`.`), then b is treated as $+\infty$. a and b can be specified as numeric values, missing (`.`), or variable names.

Options for how results are calculated

By default, predictions are calculated taking into account all complications. This is discussed in [Remarks and examples](#) of [\[ERM\] `eregress predict`](#).

`fix(varname ...)` specifies a list of endogenous variables from the main equation to be treated as if they were exogenous. This was discussed in [\[ERM\] `intro 3`](#) and is discussed further in [Remarks and examples](#) of [\[ERM\] `eregress predict`](#).

`base(varname = ...)` specifies a list of variables from any equation and values for them. Those values will be used in calculating the expected value of $e_i.y$. 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 3`](#) and is discussed further in [Remarks and examples](#) of [\[ERM\] `eregress predict`](#).

`target(varname = ...)` is sometimes a more convenient way to specify the `fix()` and `base()` options. You specify a list of variables from the main equation and values for them. Those values override the values of the variables calculating $\beta_0 + \beta_1 x_{1i} + \dots$. Use of `target()` is discussed in [Remarks and examples](#) of [\[ERM\] `eregress predict`](#).

Remarks and examples

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

Predictions after fitting models with `eintreg` are handled the same as they are after fitting models with `eregress`. The issues are the same. See [ERM] [eregress predict](#).

Note that censoring is treated as a nuisance in `eintreg`. Predicted values are not `y1` and `yu`, they are `y`.

Methods and formulas

See *Methods and formulas* in [ERM] [eintreg postestimation](#).

Also see

[ERM] [eintreg postestimation](#) — Postestimation tools for `eintreg`

[ERM] [eintreg](#) — Extended interval regression