Title

stata.com

eregress predict — predict after eregress and xteregress

Description

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

Syntax

You previously fit the model

```
eregress y x1 ..., ...
```

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

$$\texttt{y}_i = \beta_0 + \beta_1\texttt{x1}_i + \cdots + e_i.\texttt{y}$$

Or perhaps you had panel data and you fit the model with `xteregress` by typing

```
xteregress y x1 ..., ...
```

Then the main equation would be

$$\texttt{y}_{ij} = \beta_0 + \beta_1\texttt{x1}_{ij} + \cdots + u_i.\texttt{y} + v_{ij}.\texttt{y}$$

In either case, `predict` calculates predictions for `y` in the main equation. The other equations in the model are called auxiliary equations or complications. Our discussion follows the cross-sectional case with a single error term, but it applies to the panel-data case when we collapse the random effects and observation-level error terms, $e_{ij}.\texttt{y} = u_i.\texttt{y} + v_{ij}.\texttt{y}$.

All predictions after `xteregress` assume the panel-level random effects ($u_i.\texttt{y}$) are zero. Put another way, predictions condition on random effects being set to their mean.

The syntax of `predict` is

`predict` [*type*] *newvar* [*if*] [*in*] [`,` *statistic* *asfmethod* *counterfactual*]

statistic	Description
Main	
`mean`	linear prediction; the default
`xb`	linear prediction excluding all complications
`expmean`	expected value of exponential of the `mean`; $E\{\exp(\texttt{mean})\}$

asfmethod	Description
Main	
`asf`	average structural function; the default
`fixedasf`	fixed average structural function
`noasf`	no average structural function adjustment

counterfactual	Description
Main	
`target(`*valspecs*`)`	specify counterfactuals

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, `target(`*valspecs*`)` could be `target(w1=0)` or `target(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. It can also compute some rarely used statistics. See [ERM] **predict advanced**.

Options for statistics

Main

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

`expmean` calculates the expected value of the exponential of the mean. This is particularly useful when the dependent variable is estimated in the log metric but you want to express results in the natural metric of the dependent variable. `expmean` accounts for integrating over the error when forming the expected value of the exponential of the mean. That expectation is not zero.

As with the nonexponentiated mean, results depend on how complications are handled, which is determined by the *asfmethod* and *counterfactual* options. So, by default, the exponential mean has a structural interpretation because the default `asf` option has computed the average structural function of the exponential mean.

Options for asfmethod

Main

`asf`, `fixedasf`, and `noasf` determine whether and how the average structural function (ASF) of the specified statistic is computed. These options are not allowed with `xb`.

`asf`, the default, calculates the ASF of the statistic. Thus, the default when no *statistic* is specified is the ASF of the linear prediction.

`asf` computes the statistic conditional on the errors of the endogenous variable equations. Put another way, it is the statistic accounting for the correlation of the endogenous covariates with the errors of the outcome equation. Derivatives and contrasts based on `asf` have a structural interpretation. See `margins` for computing derivatives and contrasts.

`fixedasf` calculates a fixed ASF. It is the specified statistic computed using only the coefficients and variables of the outcome equation. `fixedasf` does not condition on the errors of the endogenous variable equations. Contrasts between two fixed counterfactuals averaged over the whole sample have a potential-outcome interpretation. Intuitively, it is as if the values of the covariates were fixed at a value exogenously by fiat. See `margins` for computing derivatives and contrasts.

To be clear, derivatives and contrasts between two fixed counterfactuals using the default `asf` option also have a potential-outcome interpretation. And, unlike `fixedasf`, they retain that interpretation when computed over subpopulations for both linear and nonlinear models.

`noasf` calculates the statistic using the linear prediction with no adjustment. For extended regression models, this is computationally equivalent to `fixedasf`. So `fixedasf` and `noasf` are synonyms.

Option for counterfactuals

Main

`target(`*valspecs*`)` specifies counterfactual predictions. You specify a list of variables from the main equation and values for them. Those values override the values of the variables calculating $\beta_0 + \beta_1 \texttt{x1}_i + \cdots$. Use of `target()` is discussed in *Remarks and examples* below.

Remarks and examples

stata.com

Remarks are presented under the following headings:

How to think about the model you fit

You have fit a model, perhaps by typing

```
. eregress y x1 x2                                              (1)
```

or

```
. eregress y x1 x2, endogenous(x1 = z1 z2, nomain)              (2)
```

The equation specified immediately after the `eregress` command is called the main equation. In the models above, it is

```
. eregress y x1 x2                                              (1)
. eregress y x1 x2                                              (2)
```

The equations specified in the options are called the auxiliary equations or complications. In the models above, they are

```
none                                                            (1)
. endogenous(x1 = z1 z2, nomain)                                (2)
```

The auxiliary equations arose because of complications in the data you used to fit the model. The focus of ERMs is on fitting the main equation correctly in the presence of complications.

The default asf mean calculation for predictions

When you use `predict` without options, you type

```
. predict yhat
```

`predict` calculates the expected values of $\texttt{y}_i$ that would be observed given the complications present in your data.

Let's consider the two models we mentioned earlier.

```
. eregress y x1 x2                                              (1)
. eregress y x1 x2, endogenous(x1 = z1 z2, nomain)              (2)
```

The result from typing `predict yhat` without options will be

1. The expected values of $\texttt{y}_i$ given `x1` and `x2`.
2. The expected values of $\texttt{y}_i$ given `x1` and `x2` and taking into account that `x1` is endogenous and predicted by `z1` and `z2`.

`predict` without options can be used to calculate expected values with the data used in fitting the model and with other data that include the same complications. After fitting the model, you can type

```
. use anotherdataset
. predict yhat
```

You will sometimes use `predict` to calculate counterfactuals, although most of the time you can get the answers you want using `margins`.

You cannot haphazardly change the value of an endogenous variable such as x1 and expect to produce meaningful results. What would happen if you did? In (2) above, there is an equation for x1. It is

```
endogenous(x1 = z1 z2, nomain)
```

which, written mathematically, is

$$\texttt{x1}_i = \gamma_0 + \gamma_1 \texttt{z1}_i + \gamma_2 \texttt{z2}_i + e_i.\texttt{x1}$$

Say you type the following:

```
. replace x1 = x1 + 1
. predict yhat
```

You increased x1 by 1 but did not change anything else. The equation above still holds, and so incrementing x1 increased e_i.x1 by 1 too.

What does it mean to increase e_i.x1? You are assuming that x1 increased by 1 because the subjects decided to choose x1+1 instead of x1. The only way that could happen is if they were different subjects.

Here is the thought experiment you just performed. You have data on subjects. What if you had different data on different subjects, each with the same characteristics as the current subjects, but who had chosen a value of x1 that was one unit larger. Well, if these alternate subjects had chosen a value one unit larger than the current subjects, they would have done so for good reason, and their larger e.x1 would have passed along its effect to the e.y because of the correlation. The new value of y would be the direct effect of x1 in the y equation plus the change in e.y.

predict yhat without options produces the answer to the question that you never wanted to ask. What you wanted to ask was what would be the effect on y for the current subjects if endogenous variable x1 was "exogenously" incremented by 1.

The subjects in your data are who they are because of their errors. Errors such as e.x1 are the unobserved things about them that affect their choice of x1. You cannot change their errors without changing those unobserved things that make them who they are. If you want to ask about the effects of changes in x1 holding the subjects constant, you need to ask about changes in x1 holding e_i.x1 constant.

To compute the counterfactual you want, you would type

```
. predict yhat, target(x1=(x1+1))
```

target() makes its changes to form the counterfactuals after the estimates of all errors like e_i.x1 and e.y and their implied unobserved components have been formed from the observed data. So your subjects retain the estimates of their original unobserved components when the counterfactual is computed.

All of this works because by default predict computes values on the ASF. See Blundell and Powell (2004) and Wooldridge (2010) for detailed discussions on ASFs and their interpretation.

The fixedasf calculation for predictions

The purpose of the *counterfactual* and *asfmethod* options is to make meaningful counterfactuals when you change the values of endogenous covariates. The fixedasf option makes predictions as if the complications associated with *varname* were removed.

Assume you have fit (2):

```
. eregress y x1 x2 selected, endogenous(x1 = z1 z2, nomain)
```

Then, typing

```
. predict yhat1, fixedasf
```

would produce predictions that correspond to "what would have been observed" if the complication for `x1` had not been present either in the data or in the fitted model.

In this counterfactual world, `x1` is no longer endogenous. This switch from being endogenous to being exogenous is not a technicality. It is full of import. In the real world, $e.$`x1` is correlated with $e.$`y`. When we made the default prediction in the previous section, that correlation was taken into account. In this alternative world, there is no correlation. Perhaps `x1` records each subject's amount of health insurance coverage, and `y` is a health outcome. In the world of the data used to fit the model, subjects chose to purchase health insurance, and presumably those who perceived a larger benefit would purchase more. Thus, the correlation between $e.$`x1` and $e.$`y` was positive. In the counterfactual world, we want to assume that everyone has $1 million coverage, perhaps because the purchase of this level of health insurance is mandatory or its free. Either way, the correlation between $e.$`x1` and $e.$`y` becomes 0.

Let's now calculate a counterfactual prediction under this scenario. To fix `x1` at $1 million `x1 = 1`, you would type

```
. predict yhat2, target(x1=1) fixedasf
```

We have predicted a counterfactual for which all individuals have a value of $1 million for `x1` and for which the correlation of the unobservables and the covariates is zero.

Methods and formulas

See *Methods and formulas* in [ERM] **eregress postestimation**.

References

Blundell, R. W., and J. L. Powell. 2004. Endogeneity in semiparametric binary response models. *Review of Economic Studies* 71: 655–679. https://doi.org/10.1111/j.1467-937X.2004.00299.x.

Wooldridge, J. M. 2010. *Econometric Analysis of Cross Section and Panel Data*. 2nd ed. Cambridge, MA: MIT Press.

Also see

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

[ERM] **eregress** — Extended linear regression