STATA EXTENDED REGRESSION MODELS REFERENCE MANUAL
RELEASE 18

The suggested citation for this software is

StataCorp. 2023. *Stata 18*. Statistical software. StataCorp LLC.

The suggested citation for this manual is

StataCorp. 2023. *Stata 18 Extended Regression Models Reference Manual*. College Station, TX: Stata Press.

www.stata.com

Contents

i

Cross-referencing the documentation

When reading this manual, you will find references to other Stata manuals, for example, [U] **27 Overview of Stata estimation commands**; [R] **regress**; and [D] **reshape**. The first example is a reference to chapter 27, *Overview of Stata estimation commands*, in the *User's Guide*; the second is a reference to the `regress` entry in the *Base Reference Manual*; and the third is a reference to the `reshape` entry in the *Data Management Reference Manual*.

All the manuals in the Stata Documentation have a shorthand notation:

[GSM]	*Getting Started with Stata for Mac*
[GSU]	*Getting Started with Stata for Unix*
[GSW]	*Getting Started with Stata for Windows*
[U]	*Stata User's Guide*
[R]	*Stata Base Reference Manual*
[ADAPT]	*Stata Adaptive Designs: Group Sequential Trials Reference Manual*
[BAYES]	*Stata Bayesian Analysis Reference Manual*
[BMA]	*Stata Bayesian Model Averaging Reference Manual*
[CAUSAL]	*Stata Causal Inference and Treatment-Effects Estimation Reference Manual*
[CM]	*Stata Choice Models Reference Manual*
[D]	*Stata Data Management Reference Manual*
[DSGE]	*Stata Dynamic Stochastic General Equilibrium Models Reference Manual*
[ERM]	*Stata Extended Regression Models Reference Manual*
[FMM]	*Stata Finite Mixture Models Reference Manual*
[FN]	*Stata Functions Reference Manual*
[G]	*Stata Graphics Reference Manual*
[IRT]	*Stata Item Response Theory Reference Manual*
[LASSO]	*Stata Lasso Reference Manual*
[XT]	*Stata Longitudinal-Data/Panel-Data Reference Manual*
[META]	*Stata Meta-Analysis Reference Manual*
[ME]	*Stata Multilevel Mixed-Effects Reference Manual*
[MI]	*Stata Multiple-Imputation Reference Manual*
[MV]	*Stata Multivariate Statistics Reference Manual*
[PSS]	*Stata Power, Precision, and Sample-Size Reference Manual*
[P]	*Stata Programming Reference Manual*
[RPT]	*Stata Reporting Reference Manual*
[SP]	*Stata Spatial Autoregressive Models Reference Manual*
[SEM]	*Stata Structural Equation Modeling Reference Manual*
[SVY]	*Stata Survey Data Reference Manual*
[ST]	*Stata Survival Analysis Reference Manual*
[TABLES]	*Stata Customizable Tables and Collected Results Reference Manual*
[TS]	*Stata Time-Series Reference Manual*
[I]	*Stata Index*
[M]	*Mata Reference Manual*

Title

Intro — Introduction to extended regression models manual

Description   Remarks and examples

Description

ERM stands for extended regression model. The ERMs are linear regression, interval regression, probit, and ordered probit. This manual introduces, explains, and documents ERM features.

Remarks and examples

The entries in this manual are organized as follows:

Introductions
Examples
ERM commands
Postestimation
Technical details
Glossary

Introductions

Read the introductions first.

We recommend reading [ERM] **Intro 1**–[ERM] **Intro 7** in order. In them, you will find introductions to the models that can be fit with the ERM commands, the syntax, the complications—endogenous covariates, sample selection, treatment assignment, and observations that are correlated within panels or groups—that ERM commands address, and the interpretation of results.

[ERM] **Intro 1**   An introduction to the ERM commands
[ERM] **Intro 2**   The models that ERMs fit
[ERM] **Intro 3**   Endogenous covariates features
[ERM] **Intro 4**   Endogenous sample-selection features
[ERM] **Intro 5**   Treatment assignment features
[ERM] **Intro 6**   Panel data and grouped data model features
[ERM] **Intro 7**   Model interpretation

The next introduction is a Rosetta stone for anyone who has used other Stata commands to account for endogenous covariates, sample selection, nonrandom treatment assignment, or panel data. It provides a simple mapping of syntax from commands such as `ivregress`, `heckman`, `xtreg`, `ivprobit`, `heckoprobit`, `xttobit` and `etregress` to the corresponding ERM command. If you are already familiar with these other commands, this entry may be all you need to get started using the ERM commands.

[ERM] **Intro 8**   A Rosetta stone for extended regression commands

Finally, we include an introduction to some of the important concepts in ERMs in the context of a worked example. Here, we discuss endogeneity, sample selection, and nonrandom treatment assignment. We fit models that account for each of these complications, and we show you how to use postestimation commands to interpret the results.

[ERM] **Intro 9**   Conceptual introduction via worked example

[ERM] **Intro 9** can be read either before or after [ERM] **Intro 1**–[ERM] **Intro 7**.

Examples

The example entries demonstrate how to fit models using eregress, eintreg, eprobit, eoprobit, xteregress, xteintreg, xteprobit, and xteoprobit.

We do not recommend selecting the examples you read based only on the type of outcome discussed in the example. The syntax of the ERM commands is interchangeable. Therefore, you can substitute eintreg, eoprobit, eprobit, or eregress for each other to fit a model that addresses the same complications. The xteintreg, xteoprobit, xteprobit, and xteregress commands address one additional complication—observations that are correlated within panels or groups. Again, the syntax is interchangeable. You can add xt to the beginning of any other ERM commands and fit random-effects models that address this additional complication. Remove the xt from the beginning of the command to fit the same model without random effects. The table below lists the command, the type of outcome variable, and the complications that are addressed in each example to help you locate examples that are of most interest to you.

Example	Command	Outcome	Complications
[ERM] **Example 1a**	eregress	continuous	continuous endogenous covariate
[ERM] **Example 1b**	eintreg	interval	continuous endogenous covariate
[ERM] **Example 1c**	eintreg	interval	continuous endogenous covariate, endogenous sample selection
[ERM] **Example 2a**	eregress	continuous	binary endogenous covariate
[ERM] **Example 2b**	eregress	continuous	exogenous treatment
[ERM] **Example 2c**	eregress	continuous	endogenous treatment
[ERM] **Example 3a**	eprobit	binary	continuous endogenous covariate
[ERM] **Example 3b**	eprobit	binary	continuous endogenous covariate, endogenous treatment
[ERM] **Example 4a**	eprobit	binary	endogenous sample selection
[ERM] **Example 4b**	eprobit	binary	endogenous sample selection, endogenous treatment
[ERM] **Example 5**	eprobit	binary	endogenous ordinal treatment
[ERM] **Example 6a**	eoprobit	ordinal	endogenous treatment
[ERM] **Example 6b**	eoprobit	ordinal	endogenous treatment, endogenous sample selection
[ERM] **Example 7**	xteregress	continuous	continuous endogenous covariate, random effects in all equations
[ERM] **Example 8a**	xteregress	continuous	continuous endogenous covariate, random effects in one equation
[ERM] **Example 8b**	xteregress	continuous	continuous endogenous covariate, endogenous sample selection, random effects in two equations
[ERM] **Example 9**	xteoprobit	ordinal	endogenous treatment, random effects in all equations

The type of outcome does play a role in the way results are interpreted, so examples with the same outcome type will be of interest for interpretation. If your main interest is in interpretation, also see [ERM] **Intro 7** and [ERM] **Intro 9**.

ERM commands

The entries for the individual commands provide details on syntax and implementation. The *Methods and formulas* sections include full details on the models that can be fit using these commands. The xteintreg, xteoprobit, xteprobit, and xteregress commands are documented in these entries as well.

[ERM] **eintreg**	Extended interval regression
[ERM] **eoprobit**	Extended ordered probit regression
[ERM] **eprobit**	Extended probit regression
[ERM] **eregress**	Extended linear regression
[ERM] **ERM options**	Extended regression model options

Postestimation

The postestimation commands allow you to estimate treatment effects, obtain predictions, perform tests, and more. They are documented in the entries listed below.

Examples using postestimation commands are found in [ERM] **Intro 9** and in the example entries.

Technical details

ERM commands require that endogenous covariates form a triangular or recursive system. Here, we discuss triangular systems and possible solutions if your model does not have this required form.

Glossary

Finally, we provide a glossary that can be referred to as needed.

Title

Intro 1 — An introduction to the ERM commands

Description

ERM stands for extended regression model. It is our term to designate commands for fitting linear regression, interval regression, probit, and ordered probit models that allow

- continuous, binary, and ordinal endogenous covariates,
- polynomials of endogenous covariates,
- interactions of endogenous covariates,
- interactions of endogenous with exogenous covariates,
- endogenous sample selection,
- nonrandom exogenous or endogenous treatment assignment, and
- observations that are correlated within panels or within groups.

The features may be used separately or in any combination.

The estimation commands eregress, eintreg, eprobit, and eoprobit fit ERMs that allow all the features above except correlation within panels or groups. The commands xteregress, xteintreg, xteprobit, and xteoprobit fit random-effects models that allow for within-panel or within-group correlation in addition to all the other features.

Remarks and examples

Remarks are presented under the following headings:

> *The problems ERMs solve*
> *The simple syntax of ERMs*
> *Normality assumption underlying ERMs*
> *Learning more about ERMs*

The problems ERMs solve

The ERM commands fit the following models:

Command	Purpose
eregress	linear regression
eintreg	interval regression
eprobit	binary-outcome probit regression
eoprobit	ordinal-outcome probit regression
xteregress	random-effects linear regression
xteintreg	random-effects interval regression
xteprobit	random-effects binary-outcome probit regression
xteoprobit	random-effects ordinal-outcome probit regression

These models are described in [ERM] **Intro 2**.

The ERM commands provide the following features:

- **Endogenous covariates**

  Explanatory variables in the model—covariates—can be exogenous or endogenous.

  Endogenous covariates can themselves be continuous (linear), binary (probit), or ordinal (ordered probit).

  Endogenous covariates can be interacted with other covariates, whether endogenous or exogenous. They can even be interacted with themselves to form polynomials.

  Endogenous covariates can themselves be predicted by other endogenous covariates.

- **Endogenous selection**

  Models can be adjusted for situations in which outcomes are unobserved for endogenous reasons.

  In a medical trial, patients may skip the final visit, causing the final outcome to be unobserved. They may skip it for reasons correlated with the outcome.

  In economic data, wages are observed only for those who have a job. Those who do not have a job may not for reasons correlated with the wage they would have received.

- **Exogenous or endogenous treatment assignment**

  The purpose of models is often to measure the effect of a treatment, such as a drug that is administered or a training program that is attended. Ethics often prevent assignment from being random.

  In a medical trial, doctors might assign patients most likely to benefit to a trial based on observed characteristics. That is called exogenous treatment assignment.

  In another situation, subjects may volunteer, and subjects who perceive larger benefits will be more likely to benefit. If all the determinants of the perceptions are observed, then assignment is exogenous. It can be explained by the observed variables, just as in the previous case.

  If the determinants are unobserved, then treatment is endogenous. Errors in the assignment equation will be correlated with errors in the outcome equation.

- **Panel data or grouped data**

  Models can include random effects to account for within-panel or within-group correlation.

  In economic data, we might have the yearly profits for the same companies for 10 years. The repeated observations for one company are not independent.

  In a medical trial, the same individuals may be observed at multiple time points. The repeated observations on the same individual are not independent.

  Observations might be students. If the students are nested in classes, observations on the students in the same class are likely to be correlated.

  Models can account for the correlation by including random effects in the outcome equation and in equations for endogenous covariates, endogenous selection, and endogenous treatment.

Stata has other commands that address each of these issues in the case of linear regression, and it has still other commands that can address some of these issues for interval regression, probit, and ordered probit. But Stata has no other commands that can adjust for all the above when they occur together. Even if your problem has only one of the issues, you may still prefer to use the ERM commands because they all have the same simple syntax.

The simple syntax of ERMs

The basic syntax of the ERM commands is Stata's standard estimation syntax: the command followed by the dependent variable followed by the covariates. Typing

```
. eregress y1 x1 x2
```

fits a linear regression of y1 on x1 and x2. You can fit a linear regression of y1 on x1 and x2 with random effects for id by typing

```
. xtset id
. xteregress y1 x1 x2
```

If you need to use one or more other ERM features, you add options to the command.

Option	Purpose
endogenous()	add endogenous covariates
select()	add endogenous sample selection
tobitselect()	add endogenous selection using tobit
extreat()	add exogenous treatment assignment
entreat()	add endogenous treatment assignment

For instance, you can type

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

to add endogenous covariate w to the right-hand side of the model. The option specifies that w's instruments are variables x1, z1, and z2.

If you did not observe y but observed y0 and y1, where $y0 \le y \le y1$, you could fit the equivalent interval regression by typing

```
. eintreg y0 y1 x1 x2, endogenous(w = x1 z1 z2)
```

If you observed y but it contained a 0/1 binary outcome, you could fit the equivalent probit model by typing

```
. eprobit y x1 x2, endogenous(w = x1 z1 z2)
```

If y contained 1, 2, or 3 for ordered categories, such as not ambulatory, partially ambulatory, and fully ambulatory, you could fit the equivalent ordered probit model by typing

```
. eoprobit y x1 x2, endogenous(w = x1 z1 z2)
```

Syntax is the same regardless of model fit.

Now, let's imagine that the outcome y is observed only when variable selected is true (that is, not equal to 0). Consider a case where the outcome is observed when

$$\gamma_0 + \gamma_1 x2 + \gamma_2 w + e.\text{selected} > 0$$

and, just to make the problem more complicated, assume that w is endogenous. To fit the model with this added complication, type

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

You would use the same syntax with the other ERM commands:

```
. eintreg y0 y1 x1 x2, endogenous(w = x1 z1 z2) select(selected = x2 w)
. eprobit y x1 x2, endogenous(w = x1 z1 z2) select(selected = x2 w)
. eoprobit y x1 x2, endogenous(w = x1 z1 z2) select(selected = x2 w)
```

Now, let's complicate the model even more. We also have the variable `treatment`, which records whether the observation was treated. `treatment` also affects y. In fact, measuring the effect of `treatment` is the primary reason we are fitting this model. Type

```
. eregress y x1 x2, endogenous(w = x1 z1 z2) select(selected = x2 w)  ///
       extreat(treatment)
```

Option `extreat()` handles exogenous treatment. Exogenous treatment is more flexible than you might expect. It handles assignment based on all the covariates used in the model, which in this case are x1, x2, and w.

But let us assume in our data that subjects volunteered. Or perhaps health care professionals assigned subjects to being treated based on information not in the model. That would be reasonable: doctors meet their patients and so know more about them than what is recorded in our data. In any case, we will assume that treatment is a function of observed variables w, z2, and z3, and we will assume that the error in the treatment equation is correlated with the error in the outcome equation. It is that last assumption that handles doctors knowing more about their patients than what is recorded in our data. To fit the model, we type

```
. eregress y x1 x2, endogenous(w = x1 z1 z2) select(selected = x2 w)  ///
       entreat(treatment = w z2 z3)
```

We changed from exogenous to endogenous treatment by swapping option `extreat()` for `entreat()`.

Let's add yet another complication. The outcome y is observed at multiple time points for each subject. Observations within subject (`id`) are likely correlated. We can model the correlation using random effects. We just `xtset` our data and add `xt` to the beginning of our `eregress` command.

```
. xtset id
. xteregress y x1 x2, endogenous(w = x1 z1 z2) select(selected = x2 w)  ///
       entreat(treatment = w z2 z3)
```

Shall we continue? We are just trying to convince you how flexible ERMs are and how simple the syntax is to fit them. We will go one more step. Let's assume that y is not continuous but is ordinal. y contains 1, 2, and 3, meaning not ambulatory, partially ambulatory, and fully ambulatory. In that case, change `xteregress` to `xteoprobit`.

```
. xteoprobit y x1 x2, endogenous(w = x1 z1 z2) select(selected = x2 w)  ///
       entreat(treatment = w z2 z3)
```

Normality assumption underlying ERMs

If you are accustomed to fitting models with `regress` and `ivregress`, you expect that results do not require that the errors be normally distributed. They merely require that they be independent and identically distributed.

The results produced by ERMs share that feature when all the equations are linear. Linear excludes `eintreg`, `eprobit`, `eoprobit`, `xteintreg`, `xteprobit`, and `xteoprobit`, as well as endogenous selection and endogenous treatment, both of which depend on a secondary probit model.

The nonlinear models that ERMs fit depend on normality.

Learning more about ERMs

What follows is a useful footnote. Other Stata commands provide a subset of the features that ERMs provide. We list them below. We will discuss ERMs more in this manual, but ERMs provide so many statistical features that we do not tell you as much about them as you would like. If you would like to know more, read the documentation for the other commands and then use the ERM commands.

`eregress` provides the features of

Feature	Command
linear regression	`regress`
instrumental variables	`ivregress`
exogenous treatment assignment	`teffects ra`
endogenous treatment assignment	`eteffects` and `etregress`
endogenous sample selection	`heckman`

`xteregress` provides the features of

Feature	Command
random effects	`xtreg`
instrumental variables with panel data	`xtivreg`

`eintreg` provides the features of

Feature	Command
interval regression	`intreg`
tobit regression	`tobit`
instrumental-variables interval regression	—
instrumental-variables tobit regression	`ivtobit`
exogenous treatment assignment	—
endogenous treatment assignment	—
endogenous sample selection	—

`xteintreg` provides the features of

Feature	Command
random-effects interval regression	`xtintreg`
random-effects tobit regression	`xttobit`

`eprobit` provides the features of

Feature	Command
probit regression	`probit`
instrumental variables	`ivprobit`
exogenous treatment assignment	`teffects ra`
endogenous treatment assignment	—
endogenous sample selection	`heckprobit`

`xteprobit` provides the features of

Feature	Command
random-effects probit regression	`xtprobit`

`eoprobit` provides the features of

Feature	Command
ordered probit regression	`oprobit`
instrumental variables	—
exogenous treatment assignment	—
endogenous treatment assignment	—
endogenous sample selection	`heckoprobit`

`xteoprobit` provides the features of

Feature	Command
random-effects ordered probit regression	`xtoprobit`

Reference

Gould, W. W. 2018. Ermistatas and Stata's new ERMs commands. *The Stata Blog: Not Elsewhere Classified*. https://blog.stata.com/2018/03/27/ermistatas-and-statas-new-erms-commands/.

Also see

Title

Intro 2 — The models that ERMs fit

Description   Remarks and examples   Also see

Description

The ERM commands fit linear regressions, interval regressions, probit regressions, and ordered probit regressions. These models are described below.

Remarks and examples

Remarks are presented under the following headings:

Linear regression models
Interval regression models
Probit regression models
Ordered probit regression models

In what follows, the expression

$$\beta_1 \mathbf{x}1_i + \beta_2 \mathbf{x}2_i + \cdots + \beta_k \mathbf{x}k_i$$

arises so often that we will write it as

$$\mathbf{x}_i \boldsymbol{\beta}$$

x1, x2, ... are variables in your data. They are the explanatory variables—the covariates—of the models that you fit. $\mathbf{x}1_i$, $\mathbf{x}2_i$, ... are the values of the variables in observation i.

Linear regression models

Linear regression is for use with continuous dependent variables. To fit a linear regression, type

    . eregress y x1 x2 ... xk

The model fit is

$$\mathbf{y}_i = \beta_0 + \mathbf{x}_i \boldsymbol{\beta} + e_i.\mathbf{y}$$

where $e.\mathbf{y}$ is the error and is assumed to be normally distributed with mean 0 and variance σ^2.

The fitted parameters are β_0, $\boldsymbol{\beta}$, and σ^2.

When you make predictions based on linear regressions, what is predicted is the expected value of y given x.

Interval regression models

Interval regression is for use with continuous dependent variables. To fit an interval regression, type

    . eintreg y1 y2 x1 x2 ... xk

11

The model fit is the same as that for linear regression except that y is not a variable in the dataset:

$$y_i = \beta_0 + \mathbf{x}_i\boldsymbol{\beta} + e_i.\mathbf{y}$$

The assumptions are the same as for linear regression too. $e.\mathbf{y}$ is assumed to be normally distributed with mean 0 and variance σ^2.

The fitted parameters are β_0, $\boldsymbol{\beta}$, and σ^2.

When you use `eintreg`, rather than specify y, the value of the dependent variable, you specify y1 and y2, where

$$\mathtt{y1}_i \leq y_i \leq \mathtt{y2}_i$$

Variables y1 and y2 specify the interval in which y is known to lie. For instance, if subject 1's blood pressure were not precisely recorded but instead a box was checked reporting that the blood pressure was in the range 110 to 139, then $\mathtt{y1}_1$ would equal 110 and $\mathtt{y2}_1$ would equal 139.

If $\mathtt{y1}_i = \mathtt{y2}_i$ in all observations, `eintreg` is the same as linear regression. All values are precisely observed.

If $\mathtt{y1}_i = \mathtt{y2}_i$ in some observations, those observations are precisely observed.

$\mathtt{y1}_i$ may contain a missing value and that means $\mathtt{y1}_i = -\infty$. In such observations, all that is known is that $y_i \leq \mathtt{y2}_i$. The observation is left-censored. If the box was checked for subject 2's blood pressure being below 120, then $\mathtt{y1}_2$ would equal . (missing value) and $\mathtt{y2}_2$ would equal 119.

$\mathtt{y2}_i$ may contain a missing value and that means $\mathtt{y2}_i = +\infty$. In such observations, all that is known is that $y_i \geq \mathtt{y1}_i$. The observations are right-censored. If the box was checked that subject 3's blood pressure was above 160, then $\mathtt{y1}_3$ would equal 161 and $\mathtt{y2}_3$ would equal . (missing value).

If both $\mathtt{y1}_i$ and $\mathtt{y2}_i$ contain missing values, then all that is known is that $-\infty \leq y_i \leq \infty$, and the observation is ignored when fitting the model.

`eintreg` can be used to fit tobit models. Assume that you have data in which y is left-censored at 0. To fit a tobit model, type

```
. generate y1 = cond(y==0, ., y)
. generate y2 = y
. eintreg y1 y2 x1 x2 ... xk
```

When you make predictions based on interval regressions, predicted is the expected value of the dependent variable, the unobserved y, conditioned on the covariates.

Probit regression models

Probit regression is for use with binary dependent variables. To fit a probit regression, type

```
. eprobit y x1 x2 ... xk
```

Variable y in theory should contain the values 0 and 1, but `eprobit` does not require that. It treats all nonzero (and nonmissing) values as if they were 1, which means a positive outcome, such as "subject was hired" or "subject tested positive". The positive result can be a negative event, such as "subject died".

The model is

$$p_i = \Pr(\text{positive outcome in obs. } i) = \Pr(\beta_0 + \mathbf{x}_i\boldsymbol{\beta} + e_i.\mathbf{y} > 0)$$

where $e.\mathtt{y}$ is assumed to be normally distributed with mean 0 and variance 1. With that assumption, the probability of a positive outcome is

$$p_i = \mathtt{normal}(\beta_0 + \mathbf{x}_i\boldsymbol{\beta})$$

The fitted parameters are β_0 and $\boldsymbol{\beta}$.

When you make predictions based on probit regressions, predicted is the probability of a positive outcome conditional on the covariates.

Ordered probit regression models

Ordered probit regression is for use with ordinal dependent variables. To fit an ordered probit regression, type

```
. eoprobit y x1 x2 ... xk
```

Variable y is expected to contain 1, 2, ..., M indicating category number although, just like oprobit, eoprobit is less demanding. y could contain values 2, 3, 5, and 8 to indicate four ordered categories. What is important is that the categories have a natural ordering and that the numbers used to represent them order the categories in the same way. eoprobit could be used with the ordered categories 1) not ambulatory, 2) partially ambulatory, and 3) fully ambulatory. Or the order of the categories could be reversed: 1) fully ambulatory, 2) partially ambulatory, and 3) not ambulatory. Reversing the order reverses the signs of the fitted coefficients but does not substantively change the model.

The model fit is
$$p_{m,i} = \Pr(\text{outcome } m \text{ in obs. } i)$$
$$= \Pr(c_{m-1} \leq \mathbf{x}_i\boldsymbol{\beta} + e_i.\mathtt{y} \leq c_m)$$

where $e.\mathtt{y}$ is assumed to be normally distributed with mean 0 and variance 1. Thus, the probability that the outcome is m is

$$p_{m,i} = \mathtt{normal}(c_m - \mathbf{x}_i\boldsymbol{\beta}) - \mathtt{normal}(c_{m-1} - \mathbf{x}_i\boldsymbol{\beta})$$

where c_0 and c_M are $-\infty$ and $+\infty$, and c_1, ..., c_{M-1} are fit from the data. The c values play the role of intercepts and are called cutpoints.

The fitted parameters are $\boldsymbol{\beta}$ and c_1, ..., c_{M-1}.

When $M = 2$, the ordered probit model reduces to the probit model with $c_0 = -\beta_0$.

When you make predictions based on ordered probit regressions, predicted are the probabilities of the dependent variable equaling each category conditional on the covariates.

Also see

[ERM] **eintreg** — Extended interval regression

[ERM] **eoprobit** — Extended ordered probit regression

[ERM] **eprobit** — Extended probit regression

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

Title

Description

Whether you fit linear regressions, interval regressions, probits, or ordered probits, the ERM commands provide the same features. One of those features is endogenous covariates, which are explained below.

Remarks and examples

Remarks are presented under the following headings:

What are endogenous and exogenous covariates?

Consider the model

$$y = \beta_0 + \beta_1 x1 + \beta_2 x2 + e.y$$

In models like this one, y is called the dependent variable or the outcome variable. x1 and x2 are called explanatory variables, exogenous variables, or (exogenous) covariates; we will simply call them covariates. $e.y$ is called the error.

For ERMs or any regression estimator to meaningfully fit models like the one above, it is required

1. that there be no omitted (confounding) variables that are correlated with x1 or x2.

2. that x1 and x2 be measured without error.

3. that there be no reverse causation. x1 and x2 affect y, but y must not affect x1 or x2.

4. that x1 and x2 not be correlated with $e.y$.

Any covariate that meets these requirements is called exogenous. Covariates that are not exogenous are endogenous.

Solving the problem of endogenous covariates

What if x1 violated some of or all the requirements? What if x1 was endogenous? Solving the problem of endogenous covariates is straightforward. You find a variable or set of variables that affect x1 but do not affect y except through their effect on x1. As those variables change, they induce a change in x1. That change in turn induces a change in y, and because that change is known to be caused only by the change in x1, the change can be used to disentangle the problem.

The variables that you use to solve the endogenous covariate problem are called instrumental variables.

In this manual, we tend to use the following notation:

Name starts with	Signifies
y	dependent variable
x	exogenous covariate
w	endogenous covariate
z	instrumental variable

Note: The above is notation, not a naming requirement. The software does not require that variables be named this way.

Because we are now assuming that x1 is an endogenous covariate, let us rename it w1 and rewrite our model:

$$y = \beta_0 + \beta_1 w1 + \beta_2 x2 + e.y$$

To fit this model, we need one or more variables to serve as instruments for w1. Those variables need to be correlated with w1 and uncorrelated with y. Let z1 and z2 be two such variables. Finding z1 and z2 is more easily said than done, and how you find them is beyond the scope of this manual. Nonetheless, two examples would not be out of order.

1. An economist needed an instrument for income and used spouse's income. Incomes of spouses are correlated, and in the research problem, there was no reason to suspect that spouse's income would affect the outcome other than through the correlation.

2. A health researcher needed an instrument for whether patients were prescribed a new drug. In the research problem, that variable might be endogenous because doctors are more likely to prescribe drugs they expect will be beneficial to patients based on characteristics unobserved in the data. The researcher used whether the drug was on formulary for the patients' insurance as an instrument because it is expected to be correlated with whether the drug was prescribed but not with the outcome.

Anyway, find one or more variables that are correlated with w1 but not with the dependent variable except through the effect on w1. We will assume variables z1 and z2 meet the criteria. We can then fit a model with w1 as a covariate by typing

```
. eregress y x2, endogenous(w1 = z1 z2)
```

The model has two covariates: exogenous covariate x2 and endogenous covariate w1. w1 was added to the model by the endogenous() option. If we wished, we could type w1 among the covariates, but then we have to specify endogenous()'s option nomain so that it does not add w1 for us. We could type

```
. eregress y x2 w1, endogenous(w1 = z1 z2, nomain)
```

Whichever syntax we use, we are using z1 and z2 as instruments for w1. There is a third instrument we could add to z1 and z2. If we wanted, we could add x2 by typing

```
. eregress y x2 w1, endogenous(w1 = z1 z2 x2, nomain)
```

We can add x2 because it is probably correlated with w1, and it most certainly affects y, and it is exogenous. We at StataCorp would add x2 almost by reflex. We explain why below.

Solving the problem of reverse causation

Instrumental variables can solve the four problems we mentioned at the beginning of this section.

1. They can solve the problem of omitted variables that are correlated with `w1`.

2. They can solve the problem of `w1` being measured with error.

3. They can solve the problem of reverse causation, meaning that `y` affects `w1`.

4. They can solve the problem of `x1` and `x2` being correlated with $e.y$.

We are not saying that we have all those problems, but instrumental variables can solve them if we do.

If we do not include `x2` among the instruments, however, problem 3 is not handled. We must include all the exogenous variables predicting `y` to handle reverse causation. In the model above, we have only one exogenous variable. If our model had been

$$y = \beta_0 + \beta_1 w1 + \beta_2 x2 + \beta_3 x3 + \beta_4 x4 + e.y$$

we would have included all of them:

```
. eregress y w1 x2 x3 x4, endogenous(w1 = z1 z2  x2 x3 x4, nomain)
```

This solution to reverse causation works with linear models, meaning `eregress` and `eintreg`. It does not work with `eprobit` and `eoprobit`. There is no solving the reverse-causation problem for those models.

You can interact endogenous covariates

What we have said so far about endogenous covariates applies not only to ERM commands but also to all of Stata's estimation commands for endogenous regressors.

A feature unique to ERMs is that you can use endogenous covariates in interactions. For instance, `eregress` can fit a model including

```
. eregress y w1 i.x2 i.x2#c.w1, endogenous(w1 = z1 i.x2, nomain)
```

In this model, we are assuming that `x2` is a dummy variable, such as attends school. `x2` is 1 when subjects attend school and is 0 otherwise. Therefore, we use `i.` factor-variable notation when we include `x2` in the model. The right-hand-side variables in this model are

`w1`	a continuous, endogenous variable
`i.x2`	attends school
`i.x2#c.w1`	attends school multiplied by `w1`

The coefficients on these variables are

β_1	effect of the endogenous continuous covariate
β_2	effect of attending school
β_3	extra effect of `w1` when attending school

`eregress` can fit this model. Stata's other instrumental-variable regression command `ivregress` could not. It would complain about the interaction `i.x2#c.w1` because of a limitation on how the usual statistical formulas work. Interactions with endogenous covariates are not allowed.

`eregress` has no difficulty with such models.

Now, we will tell a different backstory about y, w1, and x2:

y	income, job satisfaction, etc.
w1	years of schooling after high school
i.x2	dummy for schooling, whatever the level, being in a STEM subject

STEM stands for science, technology, engineering, and math. In a model such as

```
. eregress y i.x2 w1 i.x2#c.w1, endogenous(w1 = z1 i.x2, nomain)
```

extra years of schooling increase y by β_2 for non-STEM and by $\beta_2 + \beta 3$ for STEM.

ERMs not only allow interactions of endogenous with exogenous covariates but also allow interactions of endogenous with endogenous covariates and even allow interactions of endogenous covariates with themselves! Here is an example:

```
. eregress y w1 c.w1#c.w1 i.x2, endogenous(w1 = z1 i.x2, nomain)
```

In this model, the term c.w1#c.w1 means $w1^2$. Years of schooling after high school would increase y by $\beta_2 w1 + \beta_3 w1^2$.

You can also interact endogenous covariates with other endogenous covariates, such as

```
. eregress y w1 w2 c.w1#c.w2 i.x2, endogenous(w1 = z1 i.x2, nomain)  ///
                                   endogenous(w2 = z2 i.x2, nomain)
```

You can tell your own story about this model.

You can have continuous, binary, and ordered endogenous covariates

We have discussed continuous endogenous covariates. ERMs also allow binary and ordinal covariates. Consider the model

```
. eregress y w1 i.x2, endogenous(w1 = z1 i.x2, nomain)
```

Obviously, w1 is an endogenous covariate. In the previous section, we speculated that w1 was years of schooling beyond high school, but what if w1 was instead a dummy variable for having a college degree?

If you used the above model as typed, you would be using the linear probability model to handle w1. Saying that makes the situation sound better than it is. Probabilities are bounded by 0 and 1, and you would be using a linear model to fit them, meaning that some of the predicted probabilities could be below 0 or above 1. You ordinarily would have to live with that. With ERMs, you have a better alternative. You can tell eregress to use the probit model to handle w1! You type

```
. eregress y i.w1 i.x2, endogenous(w1 = z1 i.x2, probit nomain)
```

In the equation for y, we now include w1 as a factor variable, i.w1.

Interactions are allowed with binary endogenous covariates just as they are allowed with continuous endogenous covariates. You could type

```
. eregress y i.x2 i.w1 i.x2#i.w1, endogenous(w1 = z1 i.x2, probit nomain)
```

w1 could even be an ordered categorical variable. We have imagined that w1 contains values 0 and 1, with 1 meaning schooling in a STEM subject. Let's imagine that w1 contains the values 1, 2, and 3, with 1 meaning a non-STEM program, 2 meaning a mixed program with some courses from a STEM program, and 3 meaning a STEM program. To fit this model, all we have to do is change probit to oprobit:

```
. eregress y i.x2 i.w1 i.x2#i.w1, endogenous(w1 = z1 i.x2, oprobit nomain)
```

Including `i.x2#i.w1` allowed the effect of `x2` to differ across the levels of the binary or ordinal endogenous variable `w1`. However, in the models above, the variance of $e.y$ and its correlation with the other errors are assumed to be the same for each level of `w1`. If we wanted to allow $e.y$ to be heteroskedastic with different variances for different levels of `w1`, we could add the `povariance` suboption.

```
. eregress y i.x2 i.w1 i.x2#i.w1, ///
        endogenous(w1 = z1 i.x2, oprobit nomain povariance)
```

In our story with the ordered endogenous variable, this model estimates different error variances for non-STEM programs, mixed programs, and STEM programs.

We could also allow the correlations of $e.y$ with the other errors to vary across the levels of `w1` by including the `pocorrelation` suboption. You may think that `povariance` and `pocorrelation` are unusual names. To understand these names, consider that once parameters such as coefficients, variances, and correlations differ across levels of `w1`, we have entered a treatment-effects setting with treatment `w1`. Thus, we can think of this model using the potential-outcomes framework. `povariance` and `pocorrelation` request potential-outcome specific variances and correlations. See *Treatment-effect models and potential outcomes* in [ERM] **Intro 5** for more information on treatment effects and potential outcomes.

You can have instruments that are themselves endogenous

When we type

```
. eregress y w1 x2, endogenous(w1 = z1 z2, nomain)
```

we are specifying a model with an endogenous covariate and handling the problem of its endogeneity with the instruments `z1` and `z2`. The instruments we specified are exogenous in this example, but the ERM commands do not require that. If `z1` had one more of the problems we outlined at the beginning of this manual entry, then it would be endogenous and we might solve the problem that it raises by typing

```
. eregress y w1 x2, endogenous(w1 = z1 z2, nomain) endogenous(z1 = z3, nomain)
```

That could be the end of the story. ERMs can fit the above model.

We would have yet another problem, however, if `z1` also depended on `w1`. ERMs cannot fit models in which one dependent variable depends on another that depends on the first. The following model has that problem:

```
. eregress y w1 x2, endogenous(w1 = z1 z2, nomain)  ///
                    endogenous(z1 = w1 z3, nomain)
```

If we tried to fit the model, the command would complain:

```
. eregress y w1 x2, endogenous(w1 = z1 z2, nomain)
>                   endogenous(z1 = w1 z3, nomain)
 endogenous variables do not form a triangular system
    The problem may be fixable.  See triangularizing the system.
r(459);
```

The message says that the system needs to be triangular, which is another way of saying the system cannot have simultaneous causation. Do not confuse simultaneous causation with reverse causation, which we previously discussed. Reverse causation concerns one equation, its dependent variable, and a covariate. The covariate affects the dependent variable, and the dependent variable affects the covariate. Simultaneous causation concerns two or more equations. Their dependent variables are mutually dependent.

Nonetheless, the workaround for simultaneous causation is a variation on the workaround for reverse causation. If the equations involved are both linear, take one of them, remove the offending endogenous variable, and substitute the removed variable's exogenous variables.

The two equations involved in this example are

```
endogenous(w1 = z1 z2, nomain)
endogenous(z1 = w1 z3, nomain)
```

We could remove `z1` from the first equation and substitute `z3`. Or we could remove `w1` from the second equation and substitute `z2`. Doing the former results in

```
. eregress y w1 x2, endogenous(w1 = z3 z2, nomain)  ///      (1)
               endogenous(z1 = w1 z3, nomain)
```

Doing the latter results in

```
. eregress y w1 x2, endogenous(w1 = z1 z2, nomain)  ///      (2)
               endogenous(z1 = z2 z3  nomain)
```

ERMs can fit either model, and results for the main equation will be the same.

The first solution's equation for `z1` has an odd feature. The equation for variable `z1` is irrelevant because `z1` appears nowhere else in the model. We could omit the unnecessary equation and fit the model by typing

```
. eregress y w1 x2, endogenous(w1 = z3 z2, nomain)          (3)
```

That will produce the same result too.

Statistically, all the solutions are equally good. Numerically, (3) is sometimes better because it is easier for ERMs to fit models with fewer equations.

In any case, these solutions were available to us because the models involved were linear. Had they been nonlinear, there would have been no solution.

If you want to read more about this problem and its solution, see [ERM] **Triangularize**.

Video example

Extended regression models: Endogenous covariates

Also see

[ERM] **Intro 9** — Conceptual introduction via worked example

[ERM] **Triangularize** — How to triangularize a system of equations

Title

Intro 4 — Endogenous sample-selection features

Description

Endogenous sample-selection problems are handled by the `select()` option. ERMs provide probit and tobit selection. Probit selection is discussed below. Tobit selection is a variation on probit selection that uses censoring of a normal variable as an indicator of selection.

Remarks and examples

Remarks are presented under the following headings:

Is sample selection a concern in your research problem?

Say that you wish to fit the model

```
. eregress y x1 x2
```

We will tell you two stories about it. In the first, `y` is wage-and-salary income. In the second, `y` is a health outcome for people with a certain malady.

Both of these stories have issues of sample selection. Wages are observed only for people who work. Health outcomes are observed only for people with the malady who seek treatment. Do you care? You might not.

If you are an economist studying the effects of education, you might be perfectly satisfied measuring the return to schooling in terms of increased income of those who work. This would certainly be the situation if you were performing research to determine how schools could be improved.

If you are a medical researcher studying the effect of a treatment, you might be perfectly satisfied measuring the effect of the treatment on those who currently seek it. This would certainly be the situation if you were performing research to determine how the treatment could be improved.

Sample selection is of concern only when changing the selected population—those who work or those who are treated—is under consideration.

20

The problem and solution of endogenous sample selection

We wish to fit the model

```
. eregress y x1 x2
```

We observe y for some of but not all the sample. We observe x1 and x2 for the entire sample.

For instance, we might be doing a study of a walking program run by hospitals for patients after heart attacks. Doctors prescribe the program to patients who they believe will benefit. After six months in the program, recorded for each patient is

y	Meaning
1	I feel worse (tired)
2	I feel about like I did when I started the program
3	I feel better

The variable y will be missing for some of the observations in the data. Those observations correspond to the patients who were not prescribed the program. y could also be missing if patients were prescribed but dropped out of the program—were lost to follow-up—but we will ignore that right now. We will discuss lost to follow-up in [ERM] **Intro 5**.

Variable y is an ordinal variable, so rather than fitting the model by using eregress, we will fit it by using eoprobit:

```
. eoprobit y x1 x2
```

Do not type that command yet. If you did, the model would be fit using only the observations on patients who were prescribed the program, because y is missing otherwise. We are about to discuss those other patients. In fact, let's create a variable indicating whether patients were selected for inclusion in the program—we will need it later.

```
. generate selected = !missing(y)
```

There are two types of sample selection: exogenous and endogenous. Hardly any issues are created by exogenous sample selection. The real problems are raised by endogenous selection, and to discuss those issues, we need to tell you more about the walking program.

Doctors prescribed the program to their patients based on each patient's x1 and x2 values. Those variables are believed to predict how much a patient would benefit from the program. Indeed, patients in especially poor health might actually be harmed by the program. Say that we are conducting research to evaluate how well x1 and x2 predict a benefit and to consider whether the criteria for being prescribed the program should be loosened or tightened. Would extending the program to more patients be beneficial? Or is the program already being used by too many?

That the sample was selected on x1 and x2 causes no statistical issues, although it can cause complications. Assume that doctors also based their decisions on x3 but that was for administrative reasons. That sounds horrible, but it is not necessarily bad; for example, if a patient lives far from the hospital, the doctor might not prescribe the hospital's walking program as readily. In any case, x3, the distance a patient lives from the hospital, affected the decision but is not believed to affect how beneficial the program is for the patient. If we are certain about that, we can ignore x3. If we are uncertain, we should add x3 to the model to verify that the effect really is 0.

The above situation is called exogenous sample selection. It is not a reasonable story, but perhaps you do not yet see why. Anyway, if the only problem is exogenous sample selection, we can ignore it, and the only issue we have is to decide whether to include x3 in our model. We can fit the model by typing

```
. eoprobit y x1 x2
```

or

```
. eoprobit y x1 x2 x3
```

Typing those commands is equivalent to typing

```
. eoprobit y x1 x2 if selected
```

or

```
. eoprobit y x1 x2 x3 if selected
```

We mention this merely to emphasize that because y is missing in the group for which selected is 0, all observations for which selected is 0 are omitted from the estimation subsample.

The problem with the above story is that doctors know more about their patients than we do. They know more than what is recorded in our database. Doctors meet with their patients and get to know them, and doctors factor everything they know into their decisions. Doctors prescribed the walking program to patients who they believed would benefit. They predicted the benefit on the basis of x1, x2, and x3, as well as on information they know about the patients that is not recorded in the data.

Think of the decision that doctors make as a probit model:

$$
\begin{aligned}
p &= \text{Pr(prescribed)} \\
&= \text{Pr}(\beta_0 + \beta_1 \text{x1}_i + \beta_2 \text{x2}_i + \beta_3 \text{x3}_i + e_i.\text{selected} > 0)
\end{aligned}
$$

The important part of this model is e.selected. The error includes everything doctors know about their patients that is not recorded in the data. Because doctors presumably are making decisions in the patients' best interest, e.selected will be correlated positively with e.y, which is the error in the model's main equation fit by

```
. eoprobit y x1 x2
```

If we fit the model ignoring this correlation, we would obtain results suitable for predicting outcomes among those who participated in the program but not among those who did not participate.

It is the nonzero correlation of e.y and e.selected that makes the sample-selection endogenous. eoprobit will produce estimates accounting for the correlation if we specify the select() option:

```
. eoprobit y x1 x2, select(selected = x1 x2 x3)
```

eoprobit will report $\widehat{\rho}$—the estimate of the correlation between the two errors—and it will report the coefficients in the outcome and selection models. Because we have now accounted for the endogenous sample selection, we can interpret the results in terms of the full population, not just those who were prescribed the treatment.

Endogenous sample selection handles missing not at random

select() can handle cases in which data are missing not at random (MNAR), also known as nonignorable missing data. It can handle them as long as that missingness is modeled in the select() equation. It can solve the problem of missing on unobservables.

Endogenous sample selection can be used with other features of ERMs

You can use select() with other features of ERMs, that is, with endogenous covariates, with treatment effects, and with observations that are correlated within panels or within groups. We have not discussed treatment effects or within-panel correlation yet. We will get to those in [ERM] **Intro 5** and [ERM] **Intro 6**.

In the meantime, we will show you one way that endogenous() can be used with select(). Above, we fit the model

```
. eoprobit y x1 x2, select(selected = x1 x2 x3)
```

In the story we told, x3 measured an administrative reason we think affected doctors' decisions to prescribe the walking program. Let's imagine that x3 was endogenous for one reason or another. In the original story, x3 was the distance a patient lived from the hospital. Perhaps its value is measured with error. Or perhaps x3 represents some other administrative reason we think is correlated with y. Because it is endogenous, we will now refer to this variable as w3 instead of x3. We can address the problem by using the endogenous() option:

```
. eoprobit y x1 x2, select(selected = x1 x2 w3) endogenous(w3 = z1 z2, nomain)
```

We included suboption nomain because we do not want w3 to be added to the main equation. w3 appears only in the selection equation in this model.

Be careful not to omit nomain when it is necessary. Endogenous covariates can appear in the main equation, the selection equation, or both. Consider another example in which x3 is not endogenous but x2 is. Let's call it w2 instead of x2. We could fit that model by typing

```
. eoprobit y x1, select(selected = x1 w2 x3) endogenous(w2 = z3 z4)
```

w2 will appear in the main equation because we did not also specify nomain. Some users always type nomain and explicitly specify all the covariates that appear in the main equation. You could fit the same model by typing

```
. eoprobit y x1 w2, select(selected = x1 w2 x3) endogenous(w2 = z3 z4, nomain)
```

Mechanical notes

When you specify

```
. eoprobit y ..., select(selected = ...)
```

you can specify variables in just the y equation, just the selected equation, or both. When the same variables are specified in both equations, it is called functional form identification. Statistically speaking, the situation would be better if there were some covariates that appeared in the selected equation that did not appear in the main equation, but no one is suggesting that you add irrelevant covariates to your model. Still, you should think about whether you have any such variables. We found such a variable (x3) in the story above.

Video example

Extended regression models: Endogenous sample selection

Also see

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Description

ERMs can fit treatment-effect models. Treatment can be binary (not treated or treated) or ordinal (not treated or treated or treated extremely).

Option `extreat()` specifies exogenous treatment effects.

Option `entreat()` specifies endogenous treatment effects.

ERM's treatment-effect features are explained below.

Remarks and examples

Remarks are presented under the following headings:

What are treatment-effect models?
Treatment-effect models and potential outcomes
Endogenous and exogenous treatment effects
Binary and ordinal treatment effects
Sample versus population standard errors
Using treatment effects with other ERMs
Using treatment effects with other features of ERMs
Using treat() and select() to handle lost to follow-up
Treatment statistics reported by estat teffects
Video example

What are treatment-effect models?

Let's consider a simple binary treatment-effect problem. A treatment is applied to some patients, and we want to measure its effect. We start by imagining that patients are assigned randomly to the treated group. We observe a continuous outcome y, such as blood pressure, and we think the treatment affects y. We think the treatment's effect varies with patients' age, x1.

Here is one way we could fit the model:

```
. eregress y x1 i.treated i.treated#c.x1
```

Variable `treated` specifies which patients were treated. It contains 1 or 0. The model we just fit is

$$y_i = \beta_0 + \beta_1 x1_i + \beta_2 treated_i + \beta_3 treated_i x1_i + e_i.y$$

This model says that the outcome for patients who are not treated is

$$y_i = \beta_0 + \beta_1 x1_i + e_i.y \tag{1}$$

For those treated, the outcome is

$$y_i = (\beta_0 + \beta_2) + (\beta_1 + \beta_3)x1_i + e_i.y \tag{2}$$

β_2 and β_3x1 measure the effect of being treated. That effect varies observation by observation with the patient's age. Many researchers would stop here, satisfied to have the fitted coefficients.

Researchers who fit treatment-effect models, however, usually want to know the average treatment effect (ATE). We could obtain it.

We would calculate the average outcome when being treated over the entire dataset, the average outcome when not being treated over the entire dataset, and subtract the two results to obtain an ATE for this group of patients.

In an equivalent method of obtaining the ATE, we would calculate new variable if_not_treated equal to

$$\texttt{if\_not\_treated}_i = \widehat{\beta}_0 + \widehat{\beta}_1 \texttt{x1}_i$$

We would calculate new variable if_treated equal to

$$\texttt{if\_treated}_i = \texttt{if\_not\_treated}_i + \widehat{\beta}_2 + \widehat{\beta}_3 \texttt{x1}_i$$

Then, we would subtract them:

$$\texttt{diff}_i = \texttt{if\_treated}_i - \texttt{if\_not\_treated}_i$$

We would finally calculate the mean of diff:

```
. summarize diff
```

Stata's margins command is wonderful at doing things like this, and it even reports the standard error! And we do not even have to calculate if_not_treated, if_treated, and diff. Instead, we just type

```
. margins r.treated
```

Treatment-effect models and potential outcomes

We can write the equations for the untreated and the treated, (1) and (2), in another way. For the untreated, the first potential outcome, we write

$$\texttt{y}_{0i} = \gamma_{00} + \gamma_{01}\texttt{x1} + e_i \cdot \texttt{y}_0 \tag{3}$$

For the treated, the second potential outcome, we write

$$\texttt{y}_{1i} = \gamma_{10} + \gamma_{11}\texttt{x1} + e_i \cdot \texttt{y}_1 \tag{4}$$

We can also write the ATE in terms of these potential outcomes as

$$\text{ATE} = E\left(\texttt{y}_{1i} - \texttt{y}_{0i}\right)$$

Instead of using the eregress and margins commands above, we can type

```
. eregress y x1, extreat(treated)
```

The coefficients reported are now the γ coefficients from (3) and (4). Because we fit the model using the extreat() option, we can now type the following to estimate the ATE:

```
. estat teffects
```

Endogenous and exogenous treatment effects

ERMs can fit models far more complicated than the model we just fit. That is good because the story we told above is too simple. For instance, we said that patients were assigned randomly. Ethical considerations sometimes prevent that.

There are two types of treatment effects: exogenous and endogenous. What distinguishes them is the same thing that distinguished them in [ERM] **Intro 4**, where we discussed exogenous and endogenous selection effects. It matters whether the error in the selection (treatment assignment) equation is correlated with the error in the main equation.

Here are four examples of treatment assignment.

1. Assignment is random (as above). In this case, the assignment equation contains only an error, e.treated, and it is uncorrelated with e.y. Treatment is exogenous.

2. Assignment is determined by hard-and-fast rules. There is no e.treated, or if you prefer, it is 0. Either way, it is uncorrelated with e.y. Treatment is exogenous.

3. Assignment is determined in part by hard-and-fast rules, but if the patient meets them, a coin is flipped to determine whether the patient is treated or untreated. e.treated is the coin flip, and it is uncorrelated with e.y. Treatment is exogenous.

4. Assignment is by whatever rules, if any, plus unobserved judgment. Thus, judgment appears in e.treated, and we must consider the possibility that it is correlated with e.y. Treatment is endogenous.

ERMs can fit models with exogenous or endogenous treatment assignment. You specify the extreat() or entreat() option. In the four examples above, you would specify

1. extreat(treated)

2. extreat(treated)

3. extreat(treated)

4. entreat(treated = ...)

You could fit

```
. eregress y x1 x2 x3, extreat(treated)
```

or

```
. eregress y x1 x2 x3, entreat(treated = x1 z1 z2)
```

These models estimate distinct intercepts and distinct coefficients on x1, x2, and x3 in the equation for y. They also estimate the variance of e.y, but it is assumed to be equal across treatment groups. In the potential-outcomes framework, this means that the variance of e_i.y$_0$ and that of e_i.y$_1$ are assumed to be equal. This may not be reasonable. Perhaps the variance of the error for the treated group is larger than the variance of the error for the untreated group. The povariance suboption relaxes this constraint so that distinct error variances are estimated for each potential outcome. We can type

```
. eregress y x1 x2 x3, extreat(treated, povariance)
```

and

```
. eregress y x1 x2 x3, entreat(treated = x1 z1 z2, povariance)
```

We might want to allow the correlation between error terms to differ across potential outcomes. The pocorrelation suboption specifies that distinct correlations are estimated. For this, we type

```
. eregress y x1 x2 x3, entreat(treated = x1 z1 z2, pocorrelation)
```

More likely, we would want to let both error variances and correlations vary across potential outcomes by typing

```
. eregress y x1 x2 x3, entreat(treated = x1 z1 z2, povariance pocorrelation)
```

Whichever you type, you can obtain the ATE by typing

```
. estat teffects
```

Binary and ordinal treatment effects

We have been assuming that treatment is binary. ERMs can also fit ordinal treatment models. Think of these models as all being the same treatment but of different intensities. For instance,

1. A rehabilitative exercise program might be attended not at all, once a week, or twice a week.

2. A drug might be administered in different dosages.

3. A jobs program might be attended not at all, once a week, or twice a week.

4. The amount of post-secondary education could be none, some college, graduated, or graduated plus postgraduate.

When treatment is ordinal, variable `treated` contains more than two values. The variable might contain 0, 1, or 2; or 1, 2, or 3; or even 2, 3, or 5. If there are four ordered treatments, the variable contains four different values. The particular values do not matter as long as the numeric values order the treatments in the way they should be ordered.

When the treated variable takes on more than two values, `entreat()` fits the endogenous treatment equation by using ordered probit instead of binary probit.

Sample versus population standard errors

Researchers who fit treatment models usually want population standard errors.

When we fit the treatment model by hand, not using the `extreat()` or `entreat()` option, we typed

```
. eregress y x1 i.treated i.treated#c.x1
. margins r.treated
```

When we used the `extreat()` option to fit the same model, we used `estat teffects` to obtain the ATE. We typed

```
. eregress y x1, extreat(treated)
. estat teffects
```

In both cases, the standard errors reported for the ATE were the same. The data were treated as fixed and not as a draw from the underlying population.

The standard error would also be treated that way if we fit a model with endogenous instead of exogenous treatment assignment.

```
. eregress y x1 x2 x3, entreat(treated = x1 z1 z2)
. estat teffects
```

Researchers fitting treatment-effect models often want standard errors for ATEs suitable for predicting to the entire population and not just this particular sample. If you want population-based standard errors, you must fit the model by using the `vce(robust)` option:

```
. eregress y x1, extreat(treated) vce(robust)
. estat teffects
```

Do that and `estat teffects` will report population-based standard errors.

Returning to the `eregress` command, when you do not specify `vce(robust)`, it reports OIM standard errors. OIM stands for observed information matrix. The alternative `robust` standard errors assume less and are therefore less efficient. While less efficient, robust standard errors still have correct coverage. The standard errors themselves just have more sampling variability. Robust standard errors are absolutely required if `estat teffects` is to report standard errors for the effect in the population.

Requesting the ATE with population standard errors makes sense only if the sample you are using is an unbiased random draw from the population for which you wish to make predictions. If the sample is not, you need to specify your data's probability sampling weights as well. Type

```
. eregress y x1 [pw = weight], extreat(treated) vce(robust)
. estat teffects
```

In this case, you can omit the `vce(robust)` option because it is assumed when probability sampling weights are specified.

Variable `weight` contains inverse probabilities that the observations were sampled from the population. For instance, if some observations were sampled with probability 0.001 and others with 0.0001, then `weight` contains 1,000 and 10,000. For our purposes here, the scale of weights does not matter, so `weight` could just as well contain 1 and 10. Scale of weights matters when you request totals, which `estat teffects` does not produce.

Using treatment effects with other ERMs

The outcome variable y need not be continuous. It can be interval, binary, or ordinal, meaning that you can use the `eintreg`, `eprobit`, or `eoprobit` command to fit the model.

If we had a binary outcome variable, we would type

```
. eprobit y x1 x2 x3, entreat(treated = x1 z1 z2)
```

If we planned on obtaining the ATE with population standard error, we would type

```
. eprobit y x1 x2 x3, entreat(treated = x1 z1 z2) vce(robust)
. estat teffects
```

Using treatment effects with other features of ERMs

`extreat()` and `entreat()` can be used with `endogenous()` and `select()`. Said differently, treatment models can contain endogenous covariates and be adjusted to handle endogenous sample selection. Treatment models can also include random effects to account for within-panel or within-group correlation; we will discuss this in [ERM] **Intro 6**. Here we will focus on combining treatment with endogenous covariates and sample selection.

By now, you are familiar with the `endogenous()` option. Some examples of `eregress` used with `extreat()` and `entreat()` are

```
. eregress y x1 x2 w1, extreat(treated) endogenous(w1 = x1 z1 z2, nomain)
. eregress y x1 x2 w1, entreat(treated = z3 w1)  ///
     endogenous(w1 = x1 z1 z2, nomain)
. eregress y x1 x2, entreat(treated = z3 w1) endogenous(w1 = x1 z1 z2, nomain)
```

We used the `nomain` suboption and explicitly included `w1` in the main equation if we wanted it there. In those cases, we could have omitted the explicit mention and deleted option `nomain`. Equivalent to the first example is

```
. eregress y x1 x2, extreat(treated) endogenous(w1 = x1 z1 z2)
```

So far, we have not included the `povariance` and `pocorrelation` suboptions. We can add these to estimate potential-outcome specific variances of $e.y$ and potential-outcome specific correlations between $e.y$ and $e.w1$ and between $e.y$ and $e.\texttt{treated}$. For instance, we could extend the second example to include variances and correlations that vary across treatment groups by typing

```
. eregress y x1 x2 w1, ///
     entreat(treated = z3 w1, povariance pocorrelation) ///
     endogenous(w1 = x1 z1 z2, nomain)
```

Next, we consider use of `entreat()` and `extreat()` with `select()` to account for endogenous and exogenous sample selection.

We wish to fit a treatment-effect model but there is a problem. The treatment-effect model we want to fit is

```
. eregress y x1 x2, entreat(treated = x1 z3)
```

The problem is that the information on `y` was collected at the end of the study, and some patients never showed up—they dropped out along the way. To fit the desired model with the complication, we type

```
. generate selected = !missing(y)
. eregress y x1 x2, entreat(treated = x1 z3) select(selected = x1 z4 z5)
```

To obtain the ATE, we type

```
. estat teffects
```

The model reported by `eregress` and the ATE reported by `estat teffects` will be adjusted for both the endogenous treatment assignment and the endogenous selection effects. The latter adjusts for the censored observations in which the final outcome `y` was not observed.

Reported were sample statistics. If we had wanted population statistics, we would have typed

```
. eregress y x1 x2, entreat(treated = x1 z3) select(selected = x1 z4 z5)  ///
     vce(robust)
. estat teffects
```

If treatment assignment had been exogenous, we would have specified `extreat(treated)` instead of `entreat(treated = x1 z3)`.

If we wanted to allow the variance of $e.y$ to vary across potential outcomes, we could add the `povariance` suboption to the `entreat()` or `extreat()` option. If we wanted to allow the correlations between $e.y$ and other error terms in the model to vary across potential outcomes, we could add the `pocorrelation` suboption to `entreat()` or `extreat()`.

Note in the above example that treatment can have one arm as in the story we told or it can have multiple arms. In other words, the treatment can be binary or ordinal. Nothing we typed would need to change.

Using treat() and select() to handle lost to follow-up

The important feature of the above example is that the censored observations were lost to follow-up. By that, we mean that the patients did not report for the final meeting, and thus y was unobserved. Specifying the select() equation allowed the error in selection (that is, the unobserved reasons that subjects showed up or did not show up) to be correlated with the error in the main outcome equation (the error in the benefit of the treatment). It also allowed the error in selection to be correlated with the error in the treatment assignment. Said statistically, all endogeneity issues were handled.

This was all possible because the treatment arm was assigned even for the censored observations. Variable treated was not missing. It contained a treatment-arm value just as it does in all the other observations.

What if that is not true? What if censoring occurred before the treatment arm was assigned? Then we have an issue we need to discuss. First, here is how you determine whether your data have this issue. Type

```
. assert !missing(treat) if selected==0
```

If assert reports that the assertion is false, your data have this issue. You have censored observations for which the treatment arm is unassigned.

ERMs handle this issue differently for exogenous and endogenous treatment assignment. If you are fitting an exogenous treatment model,

```
. eregress y x1 x2, extreat(treated) select(selected = x1 z4 z5)
```

ERMs do not care that the treatment arm is missing. Endogenous selection will be fully handled just as if the treatment arm had been observed. That is, ERMs handle the issue as long as the treatment arm does not appear as an explanatory variable in your selection equation.

It would not be unreasonable to fit a model such as

```
. eregress y x1 x2, extreat(treated) select(selected = treated x1 z4 z5)
```

If you are fitting this model, it should be obvious that the treatment arm must be assigned to the censored observations. Your selection equation says that the treatment arm itself will affect whether observations are censored.

Let's put that case aside and return to the usual case of missing treatment with exogenous treatment assignment. The ERM commands fit the model without problem. estat teffects will report the ATE. estat teffects has options for reporting other statistics, all of which will be fine except ATET— the average treatment effect among the treated. ATET is defined to include all treated observations. Because treated is sometimes missing because of selection, the computed ATET will exclude those observations for which treatment assignment is missing.

Now, let's consider endogenous treatment assignment. We want to fit the model

```
. eregress y x1 x2, entreat(treated = x1 z3) select(selected = x1 z4 z5)
```

What makes us hesitate is that some of or all the censored observations have treated equal to missing, meaning that treatment was evidently not assigned for them. If we typed the command and fit the model, ERM would fit it omitting those observations. This is equivalent to assuming that the observations were censored completely at random. That could be reasonable. Perhaps most of the censored observations were lost to follow-up—for them, the treatment arm is observed—and only a few were lost before treatment was assigned because of misplaced paperwork.

On the other hand, if all the censored observations were censored before the treatment arm was assigned, the model cannot be fit. Omitting those with missing treatment omits the censored observations, and there is simply no selection equation left to fit. After dropping the observations containing missing values, everyone left in the estimation sample is not censored.

The bottom line is that ERMs cannot fit this model. ERMs place selection after treatment assignment because lost to follow-up is the common case.

You might be able to salvage the situation. Is it just that the treatment-arm values are not in your dataset because the data were not entered? If so, retrieve the data. If that is not the case but the experiment is still ongoing, run the censored observations through the treatment-assignment process.

Treatment statistics reported by estat teffects

estat teffects reports ATEs, which are the average effects of the treatment if it had been applied to the entire sample or to the entire underlying population. estat teffects reports its value and standard error. The standard error is for the sample if vce(robust) was not specified when the model was fit and for the population if vce(robust) was specified when the model was fit.

Sometimes ERMs assume vce(robust) even when you do not type it. This happens when you specify features that themselves require vce(robust). Option vce(cluster) requires vce(robust). Actually, it is a variation on vce(robust), but that is not important for this problem. If you specify pweights, then vce(robust) is used too.

The types of standard errors reported will be clearly labeled on the output eregress, eintreg, eprobit, or eoprobit produces. estat teffects indicates clearly in its output, too, whether output is for the sample or the population.

estat teffects reports

ATE, the average treatment effect for each treatment for the entire sample/population.

estat teffects, atet reports

ATET, the average treatment effect for each treatment for the treated sample or population.

estat teffects, pomeans reports

POMEANS, the potential-outcome means for each treatment arm, meaning means for the untreated and means for each of the treated.

Outcomes here are the values of the dependent variable in the main equation, or y.

Potential outcomes are the values, observation by observation, of y_i that would be observed if each was treated and untreated.

Potential-outcome means are the means of treated and, separately, of untreated. The difference between them is the ATE.

All of these statistics can also be reported for subsamples or subpopulations.

Video example

Extended regression models: Nonrandom treatment assignment

Also see

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Intro 6 — Panel data and grouped data model features

Description   Remarks and examples   Also see

Description

In panel data and in other grouped data, observations within the same panel or group are not independent. The `xt` versions of the ERM commands fit models with random effects to account for the within-panel or within-group correlation. `xteregress`, `xteintreg`, `xteprobit`, and `xteoprobit` are explained below.

Remarks and examples

Remarks are presented under the following headings:

> *Random-effects models that ERMs handle*
> *Random effects can be used with other features of ERMs*

Random-effects models that ERMs handle

In [ERM] **Intro 2** through [ERM] **Intro 5**, we discussed models with observation-level errors that are assumed independent and identically distributed.

When we typed

```
. eregress y x1 x2
```

the model fit was

$$\mathbf{y}_j = \beta_0 + \beta_1 \mathbf{x1}_j + \beta_2 \mathbf{x2}_j + e_j.\mathbf{y}$$

For this model, we could tell a story about how $\mathbf{y}_j$ is the college grade point average (GPA) of student j and the error $e_j.\mathbf{y}$ represents the unobserved factors that influence $\mathbf{y}_j$. Each observation in these data could be a randomly drawn student.

What if we record the semester GPA for each student for eight semesters? In addition to the observation-level error that is represented in $e_j.\mathbf{y}$, the unobserved characteristics of the student (such as student ability) may have an effect on $\mathbf{y}_j$.

We can adjust our model to include effect of student,

$$\mathbf{y}_{ij} = \beta_0 + \beta_1 \mathbf{x1}_{ij} + \beta_2 \mathbf{x2}_{ij} + u_i.\mathbf{y} + e_{ij}.\mathbf{y}$$

Now the unobserved effect of semester j for student i is $e_{ij}.\mathbf{y}$. The unobserved effect of the student is $u_i.\mathbf{y}$. We treat the effect of student as random.

Some disciplines would refer to this model as a panel-data random-effects model where students are the panels. Others would refer to it as a multilevel model with two levels—the student level and the semester-within-student level—with random intercepts at the student level. Regardless of terminology, the model is the same. We estimate variance parameters for the student-level random effects and for the observation-level errors.

The ERM commands for random-effects models follow the style of Stata's panel-data (`xt`) commands.

If the variable `studentid` identifies the students in our sample and the variable `semester` identifies the semester, we can fit the model above by typing

```
. xtset studentid semester
. xteregress y x1 x2
```

The `xtset` command left information so that `xteregress` recognized `studentid` as the panel identifier.

Although we use the panel-data syntax, these ERM commands do not require a traditional panel dataset where you have repeated time periods within panels.

Let's change our story so that each observation is a different student. We observe GPA only once for each student. However, we still have grouped data because the students are randomly chosen from multiple colleges. Students from the same college may have more in common than students from different colleges. Now our two levels are colleges and students nested within college. College can be specified in `xtset` as the panel identifier. To fit the model, we type

```
. xtset college
. xteregress y x1 x2
```

If we have a different type of outcome, we can use one of the other panel-data ERM commands— `xteintreg`, `xteprobit`, or `xteoprobit`. For instance, if `y` is a binary indicator of whether the student graduated, we would type

```
. xtset college
. xteprobit y x1 x2
```

Ignoring the panel or group structure of your data can lead to inefficient estimates in the linear model and inconsistent estimates for nonlinear models. The `xteregress`, `xteintreg`, `xteprobit`, and `xteoprobit` commands model random effects and provide efficient and consistent estimates.

Random effects can be used with other features of ERMs

We can combine random effects with other features of ERMs, that is, with endogenous covariates, sample selection, and treatment effects.

By typing

```
. xtset college
. xteregress y x1 x2, endogenous(w1 = x1 z1 z2)
```

we would fit a linear regression model for `y` with random effects and an endogenous covariate `w1`. We would have a college-level random effect for `y`. This would be correlated with the college-level random effect for `w1`. The observation-level errors for `y` and `w1` would be correlated as well.

We can similarly include the `select()`, `extreat()`, and `entreat()` options described in [ERM] **Intro 4** and [ERM] **Intro 5** to account for sample selection and nonrandom treatment when fitting random-effects models with any of the panel-data ERM commands.

Random effects are included in the main outcome equation as well as in equations for endogenous covariates, sample selection, and endogenous treatment. If we do not believe that the other features should have random effects, we specify the `nore` suboption with `select()`, `extreat()`, or `entreat()`.

Suppose `w1` was high school GPA. The college-level random effect for high school GPA may be negligible. To fit the model with a random effect in the equation for `y` but not in the equation for the endogenous covariate `w1`, we would type

```
. xtset schoolid
. xteregress y x1 x2, endogenous(w1 = x1 z1 z2, nore)
```

Also see

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Intro 7 — Model interpretation

Description   Remarks and examples   References   Also see

Description

After you fit a model using one of the ERM commands, you can generally interpret the coefficients in the usual way. You can also use `margins` to produce counterfactuals, derivatives, contrasts, potential outcomes, treatment effects, and effects for any type of change in the covariates.

In this entry, we discuss how to interpret coefficients, how to use `margins`, and how to use `predict`. We demonstrate how this works for a simple linear model, and we discuss how the same `margins` and `predict` commands work for nonlinear and random-effects models.

Remarks and examples

Remarks are presented under the following headings:

> Use margins
> Endogenous covariates
> How to interpret coefficients
> How to use and interpret margins
> How to use margins in models without endogenous covariates
> How to use margins with endogenous covariates
> margins with predict(asf)
> margins with predict(fixedasf)
> When to use which
> Using margins with nonlinear and random-effects models
> Advanced options: Using margins predict(base()) and predict(fix())

Use margins

In many models, including many that can be fit by the ERM commands, the coefficients have a structural interpretation. An example of a structural interpretation is, "What would we expect to happen to our dependent variable if we increased the value of a covariate by one unit for everyone in the population?" So long as there are no endogenous covariates in the main equation and your model is correctly specified, the coefficients from all models fit by ERM commands have a structural interpretation. For linear models you fit with `eregress`, `eintreg`, `xteregress`, and `xteintreg`, the coefficients almost always still have a structural interpretation, even if the model has endogenous covariates.

What can we do if our main equation does have endogenous covariates and we are not using a linear estimator? Use `margins`, which automatically produces estimates of margins, derivatives, and effects that fully account for any endogenous covariates. Even if you have a linear model where the coefficients have a structural interpretation, we recommend you still use `margins`. The default results from `margins` will always have a structural interpretation, and you do not need to worry whether your model meets the criteria that allow the coefficients to be interpreted.

Here is a simple example. If you fit a model,

```
. eregress y x1 x2
```

36

and then type

```
. margins, dydx(x1)
```

`margins` will return an estimate of the expected change in y given an instantaneous unit change in x1. And, because the model is linear, this estimate, its standard error, and confidence intervals will all match those from the coefficient on x1 in the results of `eregress`.

With linear models, this result holds even if we believe x1 is endogenous. If we type

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

followed by

```
. margins, dydx(x1)
```

the results from `margins` will again match those for the coefficient x1 from `eregress`.

Things change if we are interested in a result that is not linear in the coefficient estimates. Consider the following:

```
. eprobit y x1 x2, endogenous(x1 = z1, nomain)
```

If we are interested in the average change in the probability of observing $y = 1$ in the population for an instantaneous unit change in x1, and we use only the coefficients on x1 and x2, we would not account for the contribution $e_i.y$. Prior to the instantaneous change, the probability is

$$\Pr(y_i = 1) = \Pr(\beta_0 + \beta_1 x1_i + \beta_2 x2_i + e_i.y > 0)$$

Say we computed the probability as $\Phi(\beta_0 + \beta_1 x1_i + \beta_2 x2_i)$. This will not be at the proper starting point on the nonlinear normal cumulative density curve, because we do not account for the correlation of the unobservables and the covariates. You do not need to worry about this when you use ERM postestimation commands. If you type

```
. margins, dydx(x1)
```

probability computations account for $e_i.y$ and its correlation with the covariates. The effects produced have a structural interpretation. That is to say, they have an interpretation as though we could change values in the population and observe their effect. So the bottom line is use `margins`. Assuming your model is correctly specified, the default results from `margins` will always have a structural interpretation. See Blundell and Powell (2003), Imbens and Newey (2009), and Wooldridge (2010) for a detailed description of structural functions with linear and nonlinear endogenous models.

If you want to understand more about why and how, keep reading.

Endogenous covariates

Sometimes, care must be taken interpreting the coefficients from models when your main equation contains endogenous covariates. Endogenous covariates in the main equation cause problems, which means that if your model has no endogenous covariates in the main equation, you have no problems. The following models have no endogenous covariates in the main equation:

```
. eregress y x1 x2
. eregress y x1 x2 c.x1#c.x2
. eregress y x1 x2, select(selected = x1 z1 z2)      ///
                 endogenous(z2 = z3 z4, nomain)

. xteprobit y x1 x2
```

```
. xteprobit y x1 x2 c.x1#c.x2
. xteprobit y x1 x2, select(selected = x1 z1 z2)      ///
                     endogenous(z2 = z3 z4, nomain)
```

We showed examples with `eregress` and `xteprobit`. We could just as well have shown examples with any of the other ERM commands. Note that the last model for each command we showed has an endogenous covariate, but it is *not* in the main equation.

In any case, if you have no endogenous covariates in the main equation, you interpret coefficients and use `margins` and `predict` just as you usually would. If you have endogenous covariates in `eprobit` or `eoprobit`, do not attempt to interpret your coefficients. Rather, use `margins`, and interpret the results as you normally would.

In the rest of the manual entry, when we write about models with or without endogenous covariates, we mean models with or without endogenous covariates in the main equation.

Models with endogenous covariates in the main equation require care in interpretation, even if you fit a model as simple as

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

There are four ways endogenous covariates can end up in the main equation:

1. You specify `endogenous(x1 = ...)` to add variable `x1` to the main equation.

2. You specify `endogenous(x1 = ..., nomain)`, *and* you include `x1` in the main equation.

3. You specify `entreat(treated = ...)` to handle endogenous treatment effects. `entreat()` itself adds endogenous covariate `treated` to the main equation.

4. You specify `select(selected = ...)` to handle endogenous selection, *and* you include `selected` in the main equation. `select()` makes variable `selected` endogenous, but it does not automatically add it to the main equation.

In what follows, we will show examples of endogenous covariates added to the main equation by option `endogenous()`, but we could have added them in any of the above ways.

In this manual entry, we depart from our usual practice of naming exogenous covariates `x1`, `x2`, ... and naming endogenous covariates `w1`, `w2`, ... We depart from this practice because we will introduce a situation and then say, "if `x1` is exogenous, do this; if it is endogenous, do something else".

How to interpret coefficients

For `eregress`, `eintreg`, `xteregress`, and `xteintreg`, you can almost always interpret your coefficients in the usual way. This is true even if your model has endogenous covariates in the main equation. What do we mean by "the usual way"?

Say you are interested in the effect of covariate `x1`. Whether you have typed

```
. eregress y1 x1 x2
```

or

```
. eregress y1 x1 x2, endogenous(x1 = z1, nomain)
```

or even

```
. eintreg y1 y2 x1 x2, endogenous(x1 = x2 z1, nomain)    ///
                       endogenous(z1 = x2 z2, nomain)    ///
                       select(selected = x2 z3 z4)
```

you will have fit a model where

$$y1_i = \cdots + \beta_1 x1_i + \cdots$$

You interpret the fitted coefficient β_1 as the change in y1 for a one-unit change in x1. That is true whether x1 is an exogenous or an endogenous covariate. That interpretation sounds obvious, but we will see cases later where we must be more specific about the questions we ask regarding changes to x1.

Even if x1 is interacted with another covariate, you still interpret the coefficients in the usual way. Say you have the model

```
. eregress y1 x1 c.x1#c.x2 x2
```

you will have fit a model where

$$y1_i = \cdots + \beta_1 x1_i + \beta_2 x1_i \times x2_i + \cdots$$

So a one-unit change in x1 leads to a $\beta_1 + \beta_2 x2$ change in y1. Again, this is true whether x1 is exogenous or endogenous.

We said you can "almost always interpret your coefficients in the usual way". When can you not? You cannot interpret them in the usual way when all the following are true:

1. The covariate you are trying to interpret is endogenous or is an endogenous treatment.

2. If the covariate is endogenous, it is either binary or ordinal and is so declared in the endogenous() option using suboption probit or oprobit.

3. That covariate is in the main equation.

4. There is a second endogenous covariate in the main equation.

5. You have designated that each level (category) of the covariate you are interpreting has a different outcome error variance. Or you have designated that the correlation of the outcome error with the other endogenous errors varies by the levels of the covariate you are interpreting. You specify these cases by adding suboption povariance or suboption pocorrelation to the equation for the endogenous covariate of interest.

Whew! We did say that you could "almost always interpret your coefficients in the usual way".

Here is one way to specify such a model,

```
. eregress y1 y2 x1 x2, endogenous(x1 = x2 z1, probit povariance nomain) ///
                        endogenous(x2 = z2, nomain)
```

The coefficient on x2 can be interpreted in the usual way. The coefficient on x1 cannot. Why not? The conditional-on-x2 expectation for y1 depends on the conditional-on-x2 expectation of the error for y1. Because there is a different error variance when x1 = 0 and when x1 = 1, their expectation no longer cancels out when we take the expected value of the effect. That's the "intuitive" answer. Were we conditioning on the observed value of x1 in the effect (evaluating the treatment effect on the treated), we would have the same situation. The expectation of the errors would not cancel out. See *Treatment* in `eregress` for the full mathematical explanation.

For all other models, the best approach is to use `margins`. That should give you comfort, not concern—`margins` is a clear and safe way to form inferences and to measure and test effects. In fact, feel free to use `margins` rather than the coefficients even in regressions where you can "interpret your coefficients in the usual way". `margins` will give you exactly the same answers that you will get by looking at the coefficients. `margins` also makes it easy to ask what happens if you increase

x1 by 100, rather than by 1. Or to ask what happens if you give each person an additional 100 units of x1 beyond his or her current endowment. In models with interactions or models with treatments, such questions can be tedious to answer from the coefficients.

To be completely honest, the coefficients from `eprobit` and `eoprobit` models without endogenous covariates can be interpreted in the same way as the coefficients from `probit` and `oprobit` models. The coefficients are in standard-deviation-of-the-latent-dependent-variable units. If you understood that, great, go ahead. If you did not, use `margins` for all post hoc inferences after `probit`, `oprobit`, `eprobit`, `xteprobit`, `eoprobit`, and `xteoprobit` models. With `margins`, you can easily make and test statements about how your covariates determine the levels of the probability of a positive outcome and how changes in your covariate change that probability.

How to use and interpret margins

You can always interpret the results from `margins` as being structural results. That means we can interpret them as though we were able to manipulate values in the population and compute the effect of those manipulations on the dependent variable. Obviously, this assumes the model is correctly specified.

When you have endogenous covariates, `margins` fully accounts for the correlation that gave rise to the endogeneity. For an in-depth discussion of interpretation of results of models with endogenous covariates, see Blundell and Powell (2003), Imbens and Newey (2009), and Wooldridge (2010).

How to use margins in models without endogenous covariates

If your models include no endogenous covariates in the main equation, you can use `margins` in the ordinary way. Here is how you would ordinarily use `margins`. The following model has no endogenous covariates:

```
. use https://www.stata-press.com/data/r18/ermexample
(Artificial ERM example data)
. eregress y x1 x2 c.x1#c.x2
  (output omitted )
```

The model fit is

$$y_i = \beta_0 + \beta_1 x1_i + \beta_2 x2_i + \beta_3 x1_i x2_i + e_i.y$$

Assume that our interest is in the effect of x1. One way to interpret the effect is to interpret the coefficients: a one-unit increase in x1 increases y by $\beta_1 + \beta_3 x2$. Another way to interpret the effect is by using counterfactuals. In these data, what would be the average change in y if x1 were increased by 1? `margins` will tell us if we type

```
. margins, at(x1=generate(x1)) at(x1=generate(x1+1)) contrast(at(r) nowald)

Contrasts of predictive margins                           Number of obs = 200
Model VCE: OIM

Expression: Average structural function mean, predict()
1._at: x1 =    x1
2._at: x1 = x1+1
```

	Contrast	Delta-method std. err.	[95% conf. interval]	
_at				
(2 vs 1)	1.109641	.1750625	.7665246	1.452757

You can learn about `margins`, its features, and its syntax in [R] **margins**. We will tell you enough, however, so that everything we say will make sense.

Assume that the data comprise three subgroups in which we have a special interest. For instance, we want to know how an increase in x1 would affect each subgroup. `margins` can tell us that too.

```
. margins, at(x1=generate(x1)) at(x1=generate(x1+1)) contrast(at(r) nowald)
> over(group)

Contrasts of predictive margins                           Number of obs = 200
Model VCE: OIM

Expression: Average structural function mean, predict()
Over:       group
1._at: 0.group
            x1 =    x1
       1.group
            x1 =    x1
       2.group
            x1 =    x1
2._at: 0.group
            x1 = x1+1
       1.group
            x1 = x1+1
       2.group
            x1 = x1+1
```

	Contrast	Delta-method std. err.	[95% conf. interval]	
_at@group				
(2 vs 1) 0	.5561469	.1960937	.1718102	.9404835
(2 vs 1) 1	1.123401	.1754062	.7796108	1.46719
(2 vs 1) 2	1.641114	.2153742	1.218988	2.063239

`margins` helps us to understand changes that are different in each observation. If we had the simple model `eregress y x1 x2`, we know the effect of incrementing x1 is to increase y by $\widehat{\beta}_1$, which might be 3. The change would be 3 in every observation. In the model we have, however, the effect of incrementing x1 is to increase y by $\beta_1 + \beta_3 x2$. The average effect depends on the distribution of x2.

`margins` helps us to understand how a change affects the average in our data and subgroups of our data. We are using our sample as a proxy for the population and subpopulations, but that is what we usually do in statistics. We assume that our sample is representative. The issues are the same as we discussed in [ERM] **Intro 5**.

If our sample is representative but we want `margins` to report population-based standard errors, we need to specify `vce(robust)` when we fit the model:

```
. eregress y x1 x2 c.x1#c.x2, vce(robust)
```

If our sample is not representative, we can weight it with the inverse probability that its observations were sampled from the underlying population. If we want `margins` to report population-based standard errors, we can type

```
. eregress y x1 x2 c.x1#c.x2 [pw = weight], vce(robust)
```

or type

```
. eregress y x1 x2 c.x1#c.x2 [pw = weight]
```

We can type either because specifying `[pw=weight]` implies `vce(robust)`.

Even when we do specify or imply `vce(robust)`, `margins` will report sample standard errors by default. To obtain population-based standard errors, we must specify or imply `vce(robust)` when we fit the model, and when we use `margins`, we must specify its `vce(unconditional)` option:

```
. margins, at(x1=generate(x1)) at(x1=generate(x1+1)) contrast(at(r))  ///
           vce(unconditional)
```

In the linear regression example we have been discussing, we included an interaction in the model and used `margins` to report averages. We used `margins` because the interaction caused changes to vary observation by observation. Probit and ordered probit models produce predictions that vary observation by observation even in models with no interactions. Consider the following probit model, which is almost the simplest one possible:

```
. eprobit y_p x1
```

The model is

$$\Pr(\text{positive outcome}) = \Pr(\beta_0 + \beta_1 \mathtt{x1}_i + e_i.\mathtt{y\_p} > 0) = \text{normal}(\beta_0 + \beta_1 \mathtt{x1})$$

Assume that our interest is in `x1` just as it was previously. The effect of a one-unit increase in `x1` is to increase the normal index by $\widehat{\beta}_1$. Simple, right? No, it is not. The effect in probabilities of that change varies observation by observation. Here is how the results vary if $\widehat{\beta}_1$ were 0.5 and we incremented `x1` by 1. The effect depends on each subject's initial probability of a positive outcome:

Subject's original Pr(pos. outcome)	Increment by	Subject's new Pr(pos. outcome)	Difference
0.01	0.5 s.d.	0.03	0.02
0.10	0.5 s.d.	0.22	0.12
0.20	0.5 s.d.	0.37	0.17
0.40	0.5 s.d.	0.60	0.20
0.50	0.5 s.d.	0.69	0.19
0.60	0.5 s.d.	0.77	0.17
0.90	0.5 s.d.	0.96	0.06
0.99	0.5 s.d.	1.00	0.01

A subject whose original probability was 0.40 experiences an increase of 0.20 when `x1` is incremented by 1. Meanwhile, a subject whose probability was 0.90 experiences a mere 0.06 increase.

Using `margins`, we can obtain the average changes in probabilities in the data due to incrementing `x1` by 1. We type

```
. margins, at(x1=generate(x1)) at(x1=generate(x1+1)) contrast(at(r) nowald)
Contrasts of adjusted predictions                        Number of obs = 200
Model VCE: OIM

Expression: Average structural function probability, predict()
1._at: x1 =   x1
2._at: x1 = x1+1
```

		Delta-method		
	Contrast	std. err.	[95% conf. interval]	
_at				
(2 vs 1)	.2961685	.0287644	.2397912	.3525458

We can obtain the changes for each of the three subgroups too:

```
. margins, at(x1=generate(x1)) at(x1=generate(x1+1)) contrast(at(r) nowald)
> over(group)
Contrasts of adjusted predictions                        Number of obs = 200
Model VCE: OIM

Expression: Average structural function probability, predict()
Over:        group
1._at: 0.group
            x1 =    x1
       1.group
            x1 =    x1
       2.group
            x1 =    x1
2._at: 0.group
            x1 = x1+1
       1.group
            x1 = x1+1
       2.group
            x1 = x1+1
```

		Delta-method		
	Contrast	std. err.	[95% conf. interval]	
_at@group				
(2 vs 1) 0	.3857775	.051078	.2856664	.4858885
(2 vs 1) 1	.2944176	.0294406	.2367152	.3521201
(2 vs 1) 2	.2096478	.0202614	.1699363	.2493594

Counterfactuals are useful in complicated linear models—we had an interaction in ours—and in nonlinear models whether simple or complicated.

How to use margins with endogenous covariates

Let's start with a reasonably simple model, namely,

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

The model is

$$y_i = \beta_0 + \beta_1 x1_i + \beta_2 x2_i + e_i.y$$
$$x1_i = \gamma_0 + \gamma_1 z1_i + e_i.x1$$

where $\rho = \text{corr}(e.x1, e.y)$ and is nonzero.

Let's imagine that y is a health outcome and x1 is a 0/1 variable indicating whether a treatment was administered that is expected to improve the outcome. Observations are people, and people choose for themselves whether to have the treatment. Given the story, we *should* fit the model by typing

```
. eregress y i.x1 x2, endogenous(x1 = z1, probit nomain)
```

Nonetheless, we are going to fit the model without the `probit` specification and factor-variable notation for endogenous covariate x1:

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

We omit `probit` only because it will be easier for us to explain. We need to show you some math, and the math will be simpler in the linear model case.

What is important is that ρ is likely to be nonzero, no matter how the model is fit. ρ is the correlation between $e.y$ and $e.x1$. $e.y$ includes all the unobserved things that affect how well the treatment works. $e.x1$ includes all the unobserved things that affect whether individuals choose the treatment. ρ is likely to be nonzero and positive because people who believe that they are more likely to benefit from the treatment ($e.y > 0$) should be more likely to choose the treatment ($e.x1 > 0$).

Thus, the best prediction of y that we can make for people like person 1 in our data—people who have the same value of x1, x2, and z1—includes the effect of $\widehat{\rho}$, albeit indirectly. The best prediction of y we can make for people like person 1 is that their expected value of y will be

$$\widehat{y}_1 = \widehat{\beta}_0 + \widehat{\beta}_1 x1_1 + \widehat{\beta}_2 x2_1 + \widehat{e}_1.y$$

$\widehat{e}_1.y$ is our estimate of the expected value of $e.y$ in the first observation. Expected values of errors are often 0, but not in this case. This one depends on ρ. Given that we know the values $x1_1$ and $z1_1$, we have an estimate of $e_1.x1$, namely,

$$\widehat{e}_1.x1 = x1_1 - \widehat{\gamma}_0 - \widehat{\gamma}_1 z1_1$$

Because $e.x1$ and $e.y$ are correlated, we can produce an estimate of $e_1.y$ given $\widehat{e}_1.x1$ and $\widehat{\rho}$. It is a detail, but the formula is

$$\widehat{e}_1.y = \frac{\rho \times \texttt{s.d.}(e.y)}{\texttt{s.d.}(e.x1)} \times \widehat{e}_1.x1$$

The value of $\widehat{e}_1.y$ can be calculated, and the best prediction we can make for people like person 1 includes it and is

$$\widehat{y}_1 = \widehat{\beta}_0 + \widehat{\beta}_1 x1_1 + \widehat{\beta}_2 x2_1 + \widehat{e}_1.y$$

margins with predict(asf)

Let us temporarily consider x1 to be continuous. We do this to consider what happens if we add 1 to x1.

What is the best prediction we can make for people like person 1 if x1 was incremented by 1? It is

$$\widehat{y}_1 = \widehat{\beta}_0 + \widehat{\beta}_1 (x1_1 + 1) + \widehat{\beta}_2 x2_1 + \widehat{e}_1.y$$

The above is how `margins` with the default method, `predict(asf)`, makes the calculation for each observation in the data. Observation by observation, it calculates

$$\widehat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1(\mathtt{x1}_i + 1) + \widehat{\beta}_2\mathtt{x2}_i + \widehat{e}_i.\mathtt{y} \tag{1}$$

`margins` includes $\widehat{e}_i.\mathtt{y}$ in the calculations. This is the best prediction for people like the people in our population conditioned on everything we know about them. Using the jargon in the literature, we say that it computes a structural function that averaged over your population gives you an average structural function.

But there is something more happening. You may have noticed that $\widehat{e}_i.\mathtt{y}$ is a function of the endogenous covariates, in this case $\mathtt{x1}$. But we did not add 1 to $\mathtt{x1}$ in $\widehat{e}_i.\mathtt{y}$. We are asking `margins` to compute derivatives and effects taking the level of endogeneity as given by the data. Effects that condition on the level of endogeneity have a causal interpretation. To be precise, they have a structural function interpretation. For an in-depth discussion of structural functions, see Blundell and Powell (2003), Imbens and Newey (2009), and Wooldridge (2010).

Now, we return to considering $\mathtt{x1}$ to be binary.

margins with predict(fixedasf)

`predict(asf)` uses (1) and makes its predictions given how the world currently operates. People choose their values of $\mathtt{x1}$, and the choice they make is correlated with the outcomes they expect.

`predict(fixedasf)` makes predictions for a world that operates differently. In the alternative world, $\mathtt{x1}$ is fixed at a value such as 1. This means that the population of people like person 1 is expanded from being all people like person 1 who made the same treatment choice to being all people like person 1 regardless of the treatment choice they made. In the expanded definition of people like person 1, the correlation between $e.\mathtt{y}$ and $e.\mathtt{x1}$ is broken. The correlation is now 0, and the best prediction for people like person 1 sets $\widehat{e}_1.\mathtt{y}$ to 0:

$$\widehat{\mathtt{y}}_1 = \widehat{\beta}_0 + \widehat{\beta}_1\mathtt{x1}_1 + \widehat{\beta}_2\mathtt{x2}_1 \tag{2}$$

In the jargon of statistics, $\mathtt{x1}$ is no longer endogenous—it is fixed, and the entire equation for $\mathtt{x1}$ becomes irrelevant.

When you specify `margins` with the `predict(fixedasf)` option, it makes the calculation for each person by using the approach used for person 1 in (2). It uses

$$\widehat{\mathtt{y}}_i = \widehat{\beta}_0 + \widehat{\beta}_1\mathtt{x1}_i + \widehat{\beta}_2\mathtt{x2}_i$$

These observation-by-observation predictions are called potential outcomes when applied to treatment models. The averages based on them that `margins` reports are called potential-outcome means (POMs). These averages correspond to what would be observed in a world in which $\mathtt{x1}$ is fixed at a particular value.

We refer to them as fixed average structural functions. They too have a structural interpretation when we average over the whole population and when the covariates can be considered to be fixed exogenously, by fiat.

Note that in the case described in the prior paragraph, the results from `fixedasf` and `asf` are asymptotically equivalent. So you can continue to use the default predictions for `margins` if you wish. `fixedasf` simply provides an alternative computation.

When to use which

margins can produce counterfactuals in two ways.

With the default predict(asf) method, margins uses

$$\widehat{\mathbf{y}}_i = \widehat{\beta}_0 + \widehat{\beta}_1 \mathbf{x1}_i + \widehat{\beta}_2 \mathbf{x2}_i + \widehat{e}_i.\mathbf{y}$$

for the values of x1 and x2 specified. The predictions are a function of x1 and x2 and the covariates appearing in the x1 equation. Those covariates along with $\widehat{\rho}$ go into the calculation of $\widehat{e}.\mathbf{y}$. These predictions correspond to how the current world operates.

When you specify predict(fixedasf), margins uses

$$\widehat{\mathbf{y}}_i = \widehat{\beta}_0 + \widehat{\beta}_1 \mathbf{x1}_i + \widehat{\beta}_2 \mathbf{x2}_i$$

where x1 is fixed at the value specified. These predictions are based on the exogenous covariates in the main equation (x2 in this case) and the value to which the fixed variable (x1) is set. These predictions correspond to a different world in which x1 is no longer endogenous but is fixed to a particular value.

Using margins with nonlinear and random-effects models

Above, we showed you results for one-level (cross-sectional) linear models that are fit with eregress. That discussion extends naturally when fitting any of the other ERM models.

The formulas are more complicated when models are nonlinear, but the assumptions and their implications are the same.

What if we fit a random-effects model for panel data or grouped data? If we type

```
. xteregress y x1 x2, endogenous(x1 = z1, nomain)
```

the model is

$$\mathbf{y}_{ij} = \beta_0 + \beta_1 \mathbf{x1}_{ij} + \beta_2 \mathbf{x2}_{ij} + u_i.\mathbf{y} + v_{ij}.\mathbf{y}$$
$$\mathbf{x1}_{ij} = \gamma_0 + \gamma_1 \mathbf{z1}_{ij} + u_i.\mathbf{x1} + v_{ij}.\mathbf{x1}$$

We can rewrite this in terms of the combined errors $e_{ij}.\mathbf{y} = u_i.\mathbf{y} + v_{ij}.\mathbf{y}$ and $e_{ij}.\mathbf{x1} = u_i.\mathbf{x1} + v_{ij}.\mathbf{x1}$. Then, we have

$$\mathbf{y}_{ij} = \beta_0 + \beta_1 \mathbf{x1}_{ij} + \beta_2 \mathbf{x2}_{ij} + e_{ij}.\mathbf{y}$$
$$\mathbf{x1}_{ij} = \gamma_0 + \gamma_1 \mathbf{z1}_{ij} + e_{ij}.\mathbf{x1}$$

This produces an estimate of $e_{ij}.\mathbf{y}$ that depends on estimates of $e_{ij}.\mathbf{x1}$ and $\rho = \mathrm{corr}(e_{ij}.\mathbf{x1}, e_{ij}.\mathbf{y})$.

Everything we said above about using default predictions (predict(asf)) with margins is true when we fit a random-effects model. To see this, we just replace $\widehat{e}_i.\mathbf{y}$ with $\widehat{e}_{ij}.\mathbf{y}$ in each of the formulas in the previous sections.

Advanced options: Using margins predict(base()) and predict(fix())

We have simplified our lives by creating predictions that `margins` can consume and that provide a structural function (causal) interpretation (see Blundell and Powell [2003], Imbens and Newey [2009], and Wooldridge [2010]), namely, with `predict(asf)` and `predict(fixedasf)`. Behind the scenes, to compute these structural predictions, we use two advanced prediction options, `predict(base())` and `predict(fix())`. In most cases, the effects we want are given by `margins` with the default `predict(asf)` option. But perhaps, you may find exceptions to the rule. If this is the case, this section will be helpful; otherwise, you may skip this section without any harm.

Let's think of the model given by

```
. eregress y1 x1 x2, endogenous(x1 = z1, nomain)
```

and described by

$$\widehat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1(\mathtt{x1}_i) + \widehat{\beta}_2\mathtt{x2}_i + \widehat{e}_i.\mathtt{y}$$

`margins` with the default prediction will compute inferences for a given level of endogeneity, and it will assume $\widehat{e}_i.\mathtt{y}$ is fixed. In other words, `margins` allows x1 to change in $\widehat{\beta}_1(\mathtt{x1}_i)$ but not in $\widehat{e}_i.\mathtt{y}$. This is also true for z1. `margins` does this by creating copies of the variables x1 and z1 (let us call them x1orig and z1orig) and then computing the following:

```
. margins ..., ... predict(base(x1=x1orig z1=z1orig))
```

`predict(base())` allows us to tell `margins` which endogenous equations and elements within them to fix at any value. The default `predict(asf)` method fixes the elements of the endogenous equation at its original values.

`predict(fixedasf)` proceeds similarly. Because we are fixing the correlation to be zero, we need to specify only the endogenous equations that we wish to be treated as fixed. In our example, we would type

```
. margins ..., ... predict(fix(x1))
```

You can appreciate how the bookkeeping becomes more involved as we increase the number of endogenous equations and exclude instruments such as z1.

But the utility of `predict(base())` and `predict(fix())` is not to manually compute what `predict(asf)` and `predict(fixedasf)` provide us. One case where this framework becomes handy is when we have an endogenous treatment equation and multiple endogenous equations. Say we fit

```
. eregress y1 x1, entreat(x1 = z1) endogenous(x2 = z1 z2)
```

We want to compute an average treatment effect for x1. We would type

```
. margins r.x1, predict(fix(x1))
```

When we type this, we ignore the correlation for the endogenous equation of x1 because the values of x1 here are exogenously fixed. Yet we are not ignoring the correlation of the endogenous equation for x2. This cannot be done with `predict(fixedasf)` because it will fix all endogenous equations. It cannot be done with `predict(asf)` either because it would incorporate the correlation of the endogenous treatment equation for x1.

This is a case where understanding `predict(fix())` and `predict(base())` matters. In fact, it is such an important case that we again have made it easy to obtain. Whenever you have an endogenous treatment equation, you can just type

```
. estat teffects
```

and get the average treatment effect.

We could have obtained a consistent estimate of the average treatment effect typing `margins r.x1` and using the default `predict(asf)` method. However, using `predict(fix())` is more efficient in this case.

References

Blundell, R. W., and J. L. Powell. 2003. Endogeneity in nonparametric and semiparametric regression models. In *Advances in Economics and Econometrics: Theory and Applications, Eighth World Congress*, ed. M. Dewatripont, L. P. Hansen, and S. J. Turnovsky, vol. 2, 312–357. Cambridge: Cambridge University Press.

Imbens, G. W., and W. K. Newey. 2009. Identification and estimation of triangular simultaneous equations models without additivity. *Econometrica* 77: 1481–1512. https://doi.org/10.3982/ECTA7108.

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

Also see

[ERM] **Intro 9** — Conceptual introduction via worked example

[ERM] **Example 1a** — Linear regression with continuous endogenous covariate

Title

Intro 8 — A Rosetta stone for extended regression commands

Description   Remarks and examples   Also see

Description

If you already are familiar with some or all of Stata's other commands that fit models with endogenous covariates, sample selection, random effects, and treatment effects, this entry shows you how to use that knowledge to fit equivalent models using ERMs.

Remarks and examples

Aside from providing a single coherent framework that allows complications to be combined, ERMs use similar syntax and the resulting models have the same interpretation.

In most cases, the estimation method used by the ERM commands and that used by other estimators to fit the same model produce results that are the same. Typically, there are small numerical differences because the optimization is different. Also, ancillary parameters, such as variances of errors, are sometimes parameterized differently. In some cases, a different estimation method is used. In this case, results will be asymptotically equivalent, but in finite samples, results will differ.

The table below provides a basic guide for the correspondence between Stata commands you may already be familiar with and the ERM commands.

Command you know	Equivalent extended regression command
Linear regression with endogenous covariate	
`ivregress liml y1 x (y2 = z)`	`eregress y1 x, endogenous(y2 = z x)`
Probit model with endogenous covariate	
`ivprobit y1 x (y2 = z)`	`eprobit y1 x, endogenous(y2 = z x)`
Tobit model with endogenous covariate	
`ivtobit y1 x (y2 = z), ll(0) ul(20)`	`generate y1_ll = y1`
	`replace y1_ll = . if y1<=0`
	`generate y1_ul = y1`
	`replace y1_ul = . if y1>=20 & y1<.`
	`eintreg y1_ll y1_ul x, endogenous(y2 = z x)`
Linear regression with exogenous treatment	
`teffects ra (y x1 x2) (t1)`	`eregress y x1 x2, extreat(t1) vce(robust)`
	`estat teffects`
Probit model with exogenous treatment	
`teffects ra (y x1 x2, probit) (t1)`	`eprobit y x1 x2, extreat(t1) vce(robust)`
	`estat teffects`

49

Linear regression with endogenous treatment

```
etregress y x, treat(t1 = x z)                    eregress y x, entreat(t1 = x z, nointeract)
```

Linear regression with sample selection

```
heckman y x, select(s1 = x z)                     eregress y x, select(s1 = x z)
```

Probit model with sample selection

```
heckprobit y x, select(s1 = x z)                  eprobit y x, select(s1 = x z)
```

Ordered probit model with sample selection

```
heckoprobit y x, select(s1 = x z)                 eoprobit y x, select(s1 = x z)
```

Linear regression with random effects

```
xtreg y x                                         xteregress y x
```

Linear regression with random effects and endogenous covariate

```
xtivreg y x (y2 = z)                              xteregress y x, endogenous(y2 = x z)
```

Tobit model with random effects

```
xttobit y1 x, ll(0) ul(20)                        generate y1_ll = y1
                                                  replace y1_ll = . if y1<=0
                                                  generate y1_ul = y1
                                                  replace y1_ul = . if y1>=20 & y1<.
                                                  xteintreg y1_ll y1_ul x
```

Probit model with random effects

```
xtprobit y x                                      xteprobit y x
```

Ordered probit model with random effects

```
xtoprobit y x                                     xteoprobit y x
```

___

You can build on the basic syntax of the ERM commands by combining options and suboptions, giving you the flexibility to fit a myriad of models. Here is a short list of what you might try.

Linear regression with a continuous endogenous covariate but where the exogenous variable is not included as an instrument

```
. eregress y1 x, endogenous(y2 = z1)
```

Linear regression with two continuous endogenous covariates

```
. eregress y1 x, endogenous(y2 y3 = z1 x)
```

Same as above, but with different instruments for different endogenous covariates

```
. eregress y1 x, endogenous(y2 = z1 x) endogenous(y3 = z2 x)
```

Same as above, but with one endogenous covariate being binary

```
. eregress y1 x, endogenous(y2 = z1 x) endogenous(y4 = z3 x, probit)
```

Linear regression with a continuous endogenous covariate and an endogenous treatment

```
. eregress y1 x, endogenous(y2 = z1 x) entreat(t1 = z3 x)
```

Same as above, but instead include a multivalued treatment

```
. eregress y1 x, endogenous(y2 = z1 x) entreat(t2 = z3 x, oprobit)
```

Same as above, and also allow for endogenous selection

```
. eregress y1 x, endogenous(y2 = z1 x) entreat(t2 = z3 x, oprobit)  ///
             select(s1 = w x)
```

Same as above, but where censoring of variable s2 indicates selection status

```
. eregress y1 x, endogenous(y2 = z1 x) entreat(t2 = z3 x, oprobit)  ///
             tobitselect(s2 = w x)
```

eprobit or eoprobit may be directly substituted for any eregress command above to fit a probit or an ordered probit regression when y1 is binary or ordinal. xteregress, xteprobit, or xteoprobit may be substituted for eregress to fit a random-effects linear, probit, or ordered probit regression. To fit a tobit or interval regression, you can use eintreg and specify two dependent variables containing the upper and lower bounds of the interval in place of y1. You can use xteintreg with two dependent variables to fit a random-effects tobit or interval regression.

Also see

[ERM] **Intro 1** — An introduction to the ERM commands

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Intro 9 — Conceptual introduction via worked example

Description

This entry introduces the concepts of endogenous covariates, nonrandom treatment assignment, and endogenous sample selection through a series of examples. It also provides an overview of how to interpret the results of ERMs.

Remarks and examples

Remarks are presented under the following headings:

Introduction

In a perfect research world, several assumptions we conventionally make about our data and the data-collection process would be true. For example, we could gather data about all the variables that influence the outcome we want to study. These data would be collected on a random sample of the population of interest. Any inferences we made about a relationship between the dependent variable and an independent variable when studying one group would be just as valid if we studied this group again at a different time or even if we conducted the study for a different group.

Often, applied research is complicated when one or more of the classical assumptions are not true. For example, data on key variables of interest may be unavailable. Our interest may lie in a treatment that cannot be randomly assigned or may be endogenous. Or the subjects we have available to study are not representative of the population we want to study.

When any of these things is true, we cannot make accurate inferences using standard regression methods. Stata provides many commands that can be used when one of these complications occurs. The ERM commands allow you to address these problems in isolation and, more importantly, in combination—as they often occur.

Imagine that a large company is considering offering a workplace wellness program to its employees to help them lose weight. They have conducted a pilot study at one location, and all other locations are expected to be similar. In our dataset, the `wellpgm` variable records whether a given employee participated. After one year, the company wants to know whether the program was effective. Our outcome of interest is weight lost in kilograms. We have called this `weightloss0` to distinguish it from the observed `weightloss` later.

In our fictional data, the number of kilograms lost is also determined by the employee's age in years (`age`), the employee's sex (`sex`), and the employee's starting weight in kilograms (`weight`). Because this is an entirely fictitious example, we have a true measure of willingness to engage in healthy behaviors (`health`).

More formally, in our simulated data, the process that determines weight lost is

$$\mathtt{weightloss0}_i = -4 - 0.1 \times \mathtt{age}_i - 1.5 \times \mathtt{sex}_i + 0.14 \times \mathtt{weight}_i + 1.2 \times \mathtt{wellpgm}_i$$
$$+ 0.5 \times \mathtt{health}_i + u_i$$

Suppose that we are in the situation described above. We observed complete information for all variables for all employees, and participation in the wellness program was unrelated to any employee attributes that we could not observe. In this case, we could fit our model by typing

```
. use https://www.stata-press.com/data/r18/wellness
(Fictional workplace wellness data)

. regress weightloss0 age i.sex weight i.wellpgm health
```

Source	SS	df	MS		Number of obs	=	545
					F(5, 539)	=	589.61
Model	2417.76071	5	483.552141		Prob > F	=	0.0000
Residual	442.044242	539	.820119187		R-squared	=	0.8454
					Adj R-squared	=	0.8440
Total	2859.80495	544	5.25699439		Root MSE	=	.9056

weightloss0	Coefficient	Std. err.	t	P>|t|	[95% conf. interval]	
age	-.0991644	.0038045	-26.06	0.000	-.1066378	-.0916909
sex						
Male	-1.481883	.0937504	-15.81	0.000	-1.666044	-1.297722
weight	.1359547	.0054405	24.99	0.000	.1252676	.1466419
wellpgm						
Yes	1.254928	.1076792	11.65	0.000	1.043406	1.46645
health	.4814308	.0255931	18.81	0.000	.4311564	.5317053
_cons	-3.754726	.4432054	-8.47	0.000	-4.625348	-2.884105

```
. estimates store true
```

From this model, we can estimate the average treatment effect (ATE) of the wellness program by using the coefficient on wellpgm. We estimate that the ATE is 1.25 kg. In other words, the average weight lost over the course of the year would be 1.25 kg greater if all the company's employees participated in the program versus if no employees participated.

Because we simulated these data, we can confirm that all the confidence intervals contain the true values. If we continued to add more observations, our point estimates would become closer and closer to the real values. This is true because the coefficient estimates shown above are consistent. Because they are consistent, we can make inferences about the effects of each variable on the outcome. We estimates store these values as true for comparison with later models.

Complications

As discussed in [ERM] Intro 3, a covariate is endogenous if it is correlated with the error term. Practically, this correlation arises for many reasons. For example, we may have omitted an important variable from our model that is correlated with a variable that we included, as we did here. Or we may not have accurately measured one of the covariates in our model. We could also have the case where a variable in the model and the outcome of interest are partially determined by the same unobserved factors. For concreteness, we focus on the role of a single omitted variable in this conceptual introduction.

Often in observational research, the treatment (participation in the wellness program) was not randomly assigned. As discussed in [ERM] **Intro 5**, we might be able to ignore this issue if we do not suspect that unobserved factors that affect participation also affect the amount of weight loss. However, in this case, we believe participation in the wellness program is also likely to be determined by factors we cannot observe, such as the now-omitted `health` variable.

Further, suppose that the pilot study was structured such that baseline information about all employees was collected at a mandatory benefits meeting at the start of the year. At the end of the year, all employees were asked to go to the company gym during business hours to have their year-end weight recorded, regardless of program participation. Because employees were not required to have their final weight recorded, we observe only the weight of employees who voluntarily went to the gym. We have a selected sample in this case.

Whether an employee is observed in the study could be correlated with unobserved factors that also determine how much weight he or she lost. For example, employees with high values of the now-omitted `health` variable may have generally better diet and exercise habits (independent of the wellness program), leading to higher weight loss. Let's say that for bragging rights, they want to have their superior weight loss recorded, so they are more likely to show up at the end of the year. As discussed in [ERM] **Intro 4**, if selection is related to unobserved factors that are correlated with the outcome, it cannot be ignored.

If we ignore all of these potential complications, we might erroneously fit the model below. In this model, we omit `health`, and `weightloss` records the observed weight loss only for employees who went to the gym at the end of the year.

$$\texttt{weightloss}_i = \beta_1 \times \texttt{age}_i + \beta_2 \times \texttt{sex}_i + \beta_3 \times \texttt{weight}_i + \beta_4 \times \texttt{wellpgm}_i + u_i$$

As before, we could fit the model using `regress`.

```
. regress weightloss age i.sex weight i.wellpgm
```

Source	SS	df	MS		Number of obs	=	337
					F(4, 332)	=	219.74
Model	1239.17345	4	309.793362		Prob > F	=	0.0000
Residual	468.060374	332	1.4098204		R-squared	=	0.7258
					Adj R-squared	=	0.7225
Total	1707.23382	336	5.08105304		Root MSE	=	1.1874

weightloss	Coefficient	Std. err.	t	P>|t|	[95% conf. interval]	
age	−.0800928	.0063184	−12.68	0.000	−.0925219	−.0676637
sex						
Male	−1.023886	.146934	−6.97	0.000	−1.312925	−.734847
weight	.0803689	.0074025	10.86	0.000	.0658072	.0949305
wellpgm						
Yes	1.913531	.1596906	11.98	0.000	1.599398	2.227664
_cons	−.3699558	.6741312	−0.55	0.584	−1.696063	.9561513

None of the confidence intervals for our coefficient estimates contain the true values. We store the estimates so that we can compare them with estimates from other models later.

```
. estimates store base
```

Endogenous covariates

Continuing with our example, we suspect that `weight` is endogenous now that we cannot observe `health`. Employees who are predisposed to healthy behaviors will likely have a lower starting weight, and this could influence how much weight they lose over the course of the year-long study. If we have a suitable model for how `weight` relates to the unobserved `health`, we can still estimate the parameters consistently.

Let's suppose we believe that the employee's starting weight is a function of the employee's sex and the number of times the employee visits the company gym. We measure gym use as the employee's average number of visits per month to the company gym before the program (`gym`). This will be an instrumental variable for `weight`. Instrumental variables are exogenous covariates that are correlated with the endogenous covariate, not directly related to the outcome, and not correlated with the unobserved error. Because we are using preprogram gym use, we do not expect it to be related to weight loss during the year of the program.

We fit the model using `eregress`, storing the estimates for later comparison.

```
. eregress weightloss age i.sex i.wellpgm, endogenous(weight = i.sex gym)
(output omitted)
. estimates store endog
```

Now, we view and compare the results from each of the commands. We focus on the coefficients here because our interest lies in illustrating how the point estimates change as we address different complications. At the end of the introduction, we show the full output of `eregress` and discuss its interpretation.

```
. estimates table true base endog, stats(N) equations(1) keep(#1:)
```

Variable	true	base	endog
age	-.09916437	-.08009282	-.07964086
sex Male	-1.481883	-1.023886	-1.6411717
weight	.13595472	.08036889	.14701973
wellpgm Yes	1.2549281	1.9135311	1.9008534
health	.48143082		
_cons	-3.7547263	-.36995584	-5.5172377
N	545	337	337

Once we account for the endogeneity of `weight`, the coefficients for `sex` and `weight` are close to those of the `true` model and have the correct signs. The estimates for `age` and `wellpgm`, however, are close to each other in the `base` and `endog` models but not close to the `true` values. Our estimates remain inconsistent because we have not yet addressed the endogeneity of the `wellpgm` program indicator.

Endogenous covariates in ERMs need not be continuous. We could instead have an endogenous binary or ordinal covariate. To address the endogeneity of `wellpgm`, we could include an additional model by adding another `endogenous()` option; see [ERM] **Intro 3** for more on specifying models with different types of endogenous covariates. Another way to approach the analysis of binary and ordinal endogenous covariates is in the potential-outcomes framework.

Nonrandom treatment assignment

Treatment-effect regressions model the effect of a discrete treatment or intervention on the outcome. In observational data, we cannot randomly assign a treatment of interest to individuals. Treatment status may be related to other covariates that we measure. It may even be related to the unobserved factors that affect the outcome and be endogenous. We cannot just take the sample means of the treated and untreated to estimate the ATE. Instead, we can use the potential-outcomes framework to estimate a treatment effect.

In the potential-outcomes framework, the treatment effect is the difference between the outcome that would occur when a given subject receives the treatment and the outcome that would occur when the subject receives the control instead. We only observe the potential outcome associated with that subject's observed treatment value (either treated or control). However, we can estimate both potential outcomes, conditional on covariates, by using information from the model. For more information about the potential-outcomes framework, see [CAUSAL] **teffects intro advanced**.

The ERM commands may be used with an exogenous or endogenous treatment where the treatment variable is binary or ordinal.

To address the endogenous selection of participation in the wellness program, we need a model for `wellpgm`. Whether the employee was a smoker at the beginning of the year (`smoke`) is an additional covariate in our treatment model. Because smoking signals a lower willingness to engage in healthy behaviors, it should be correlated with participation in the program, but smoking status measured before the program was offered should not be independently associated with weight loss during the program.

```
. eregress weightloss age i.sex, endogenous(weight = i.sex gym)
> entreat(wellpgm = age i.smoke, nointeract)
  (output omitted )
. estimates store entrt
```

By specifying `nointeract`, we keep the same coefficients for both treatment groups in the main equation. This is not the most common approach. However, we simulated the data this way to keep the `estimates table` results compact and easy to compare across models. We will show you a more interesting model later.

Now, we view and compare the results for the main equation for each of the models.

```
. estimates table true base endog entrt, stats(N) equations(1) keep(#1:)
```

Variable	true	base	endog	entrt
age	-.09916437	-.08009282	-.07964086	-.10430319
sex				
Male	-1.481883	-1.023886	-1.6411717	-1.5995888
weight	.13595472	.08036889	.14701973	.14151952
wellpgm				
Yes	1.2549281	1.9135311	1.9008534	.83556752
health	.48143082			
_cons	-3.7547263	-.36995584	-5.5172377	-3.488841
N	545	337	337	337

In the `entrt` model, where we have accounted for the endogeneity of starting weight and the endogenous treatment assignment to the wellness program, we estimate that the effect of participating

in the program is 0.84 kg lost. This is closer to the 1.25 kg we estimated in the `true` model than the 1.90 kg we estimated in the `endog` model that did not account for treatment assignment.

Endogenous sample selection

Sample selection is an ambiguous term because different authors have used it to mean different things. To add more ambiguity, sample selection has been equated with nonresponse bias and selection bias in some disciplines. Much of the ambiguity arises from authors not being precise about when sample selection is ignorable.

Sample selection is like treatment assignment: a process maps each individual into or out of the sample. This process depends on observable covariates and unobservable factors. When unobservable factors that affect who is in the sample are independent of unobservable factors that affect the outcome, then the sample selection is not endogenous. In this case, the sample selection is ignorable—our estimator that ignores sample selection is still consistent.

In contrast, when the unobservable factors that affect who is included in the sample are correlated with the unobservable factors that affect the outcome, the sample selection is endogenous and it is not ignorable, because estimators that ignore endogenous sample selection are not consistent.

The ERM commands may be used with endogenous sample selection with a probit or tobit selection model. A probit selection model is used when we have a binary indicator of selection. A tobit selection model is used when we have a continuous indicator for selection.

We suspect that unobserved factors that influence whether employees came to the gym for the year-end weigh-in also influence the amount of weight lost. In other words, we believe we may have endogenous sample selection. Our `true` model included all information on all 545 employees. In reality, only 337 completed the final weigh-in for our study. However, we still want to know what the potential effect of the program was for all employees. The 0.84 kg that we estimated in Nonrandom treatment assignment is not a consistent estimate of the program's ATE in the company if the 337 employees in our study are not representative of the population.

By modeling the sample-selection process, we can include all 545 employees in our estimation sample. The variable `completed` indicates whether the employee completed the final weigh-in. Employees with `completed = 0` have missing values for `weightloss`. However, because all other data were gathered at a mandatory meeting at the start of the year (such as starting weight) or collected from administrative records (such as prior-year visits to the company gym), we have complete information for all other variables.

We include the employee's job classification (`salaried`) and years employed at the company (`experience`) as additional covariates in our selection model that are excluded from the main equation. `salaried` is 1 if the employee is salaried and is 0 if the employee is paid hourly. We anticipate that salaried employees will have more opportunity to visit the gym during the day and that employees who have been with the company longer will be more motivated to help complete the study. Aside from their effect on completing the weigh-in, we do not believe that `salaried` or `experience` have any direct effect on `weightloss`.

We fit our model, accounting for the potentially endogenous selection.

```
. eregress weightloss age i.sex, endogenous(weight = i.sex gym)
> entreat(wellpgm = age i.smoke, nointeract)
> select(completed = i.wellpgm experience i.salaried)
  (output omitted )
. estimates store endsel
```

Then, we compare these estimates with those from our previous models.

```
. estimates table true base endog entrt endsel, stats(N) equations(1) keep(#1:)
```

Variable	true	base	endog	entrt	endsel
age	-.09916437	-.08009282	-.07964086	-.10430319	-.11149981
sex Male	-1.481883	-1.023886	-1.6411717	-1.5995888	-1.5607651
weight	.13595472	.08036889	.14701973	.14151952	.14353999
wellpgm Yes	1.2549281	1.9135311	1.9008534	.83556752	.92462755
health	.48143082				
_cons	-3.7547263	-.36995584	-5.5172377	-3.488841	-3.6798876
N	545	337	337	337	545

After accounting for the potentially endogenous selection that occurs because some employees chose not to complete the final weigh-in, we see that our estimated ATE is 0.925, which is closer to its true value than in the models that did not address selection.

Interpreting effects

In the previous sections, we showed only the coefficient estimates from the main outcome equation. The full output for `eregress` and the other ERM commands includes estimates of coefficients of covariates in the auxiliary models, error variances, and error correlation terms.

For many models, the coefficient estimates themselves are not directly useful. You will need to use margins or estat teffects to obtain interpretable effects. However, the correlation estimates always provide relevant information.

The full results for the last `eregress` command that we estimated are as follows:

```
. eregress weightloss age i.sex, endogenous(weight = i.sex gym)
> entreat(wellpgm = age i.smoke, nointeract)
> select(completed = i.wellpgm experience i.salaried)
```
  (*iteration log omitted*)

```
Extended linear regression                        Number of obs   =      545
                                                       Selected  =      337
                                                    Nonselected  =      208

                                                  Wald chi2(4)    = 749.04
Log likelihood = -2800.8318                       Prob > chi2     = 0.0000
```

	Coefficient	Std. err.	z	P>|z|	[95% conf. interval]	
weightloss						
age	-.1114998	.0083531	-13.35	0.000	-.1278715	-.0951281
sex						
Male	-1.560765	.2062746	-7.57	0.000	-1.965056	-1.156474
weight	.14354	.0175073	8.20	0.000	.1092263	.1778537
wellpgm						
Yes	.9246275	.2750269	3.36	0.001	.3855848	1.46367
_cons	-3.679888	1.464123	-2.51	0.012	-6.549515	-.81026
completed						
wellpgm						
Yes	.6553902	.2263862	2.90	0.004	.2116814	1.099099
experience	-.8153984	.0617977	-13.19	0.000	-.9365196	-.6942772
salaried						
Yes	.4709859	.1419878	3.32	0.001	.192695	.7492768
_cons	4.902936	.3973849	12.34	0.000	4.124076	5.681796
wellpgm						
age	-.0938617	.0072734	-12.90	0.000	-.1081173	-.079606
smoke						
Yes	-1.477078	.1772103	-8.34	0.000	-1.824404	-1.129752
_cons	4.228481	.337379	12.53	0.000	3.56723	4.889732
weight						
sex						
Male	9.506396	.6960864	13.66	0.000	8.142091	10.8707
gym	-.8184902	.0779351	-10.50	0.000	-.9712401	-.6657402
_cons	80.10245	.5407952	148.12	0.000	79.04251	81.16239
var(e.weig~s)	2.015328	.263477			1.559777	2.603927
var(e.weight)	65.98395	3.997213			58.59678	74.30241
corr(e.com~d, e.weightloss)	.5434105	.0824836	6.59	0.000	.362338	.6849556
corr(e.wel~m, e.weightloss)	.5878321	.1054098	5.58	0.000	.3440372	.7573749
corr(e.wei~t, e.weightloss)	-.4801763	.089175	-5.38	0.000	-.6353685	-.2877017
corr(e.wel~m, e.completed)	.3753168	.1523364	2.46	0.014	.0470351	.6304273
corr(e.wei~t, e.completed)	-.0643813	.0718768	-0.90	0.370	-.2030702	.0768401
corr(e.wei~t, e.wellpgm)	-.096324	.0691411	-1.39	0.164	-.2292586	.0401382

The `completed`, `wellpgm`, and `weight` equations provide the coefficient estimates for the auxiliary endogenous selection, treatment assignment, and endogenous covariate models.

The correlation estimates tell us about the endogeneity in our model. For example, we speculated that we might have endogenous selection. The error correlation `corr(e.completed,e.weightloss)` is an estimate of the correlation between the error from the selection equation and the error from the outcome equation. The estimate is significant, so we reject the hypothesis that there is no endogenous selection. It is positive, so we conclude that unobserved factors that increase the likelihood of being in the sample tend to occur with unobserved factors that increase the amount of weight lost. Looking at the other correlations, we find that our suspicions of endogenous treatment choice and the endogeneity of initial weight are likewise confirmed.

We estimated an ATE in our running example. In our simple illustration, we were able to use the coefficient on `wellpgm`. If `wellpgm` had been interacted with other covariates in the model, we would have needed to use `estat teffects`. We also could have estimated the effect of the wellness program on just those employees who participated, the average treatment effect on the treated (ATET).

Using this regression, if we ask questions about how participating in `wellpgm` affects the expected change in `weightloss`, we will almost always get the same answer: 0.92 kg greater weight loss with the program than without. That is the coefficient on `wellpgm` in the main outcome equation. This model is linear and contains no interactions between the treatment and other covariates. So, whether we ask about the ATE or the ATET, the answer is 0.92. Whether we ask about the expected additional `weightloss` for a person who chose to participate or about all the women who chose to participate, the answer is the same. No matter what, the expected change is always 0.92.

To make this interesting, we will need a more complex model. We could take the `nointeract` suboption off the `entreat()` option. If we did that and refit the model, all the questions above would produce different answers. But, as we said, our data are simulated with no interaction. So let's use another artifice.

Let's assume that the clerk in charge of the final weigh-in overheard management discussing the new program. The managers seemed particularly interested in participants losing at least 4 kg (8.8 pounds). Thinking he was being helpful, our clerk decided to save everyone some effort and did not record actual weights. Instead, he recorded only whether employees were at least 4 kg lighter than they had been at the initial weigh-in.

We can no longer analyze weight loss, but we can analyze the probability of losing at least 4 kg. We fit the same full model but this time use `eprobit`, and our dependent variable becomes `lost4`, which is 0 if the employee lost less than 4 kg and is 1 if the employee lost 4 kg or more.

```
. eprobit lost4 age i.sex, endogenous(weight = i.sex gym)
> entreat(wellpgm = age i.smoke, nointeract)
> select(completed = i.wellpgm experience i.salaried) vce(robust)
```
  (*iteration log omitted*)

Extended probit regression

Number of obs =	545
Selected =	337
Nonselected =	208

Log pseudolikelihood = -2392.5364

Wald chi2(4) =	184.27
Prob > chi2 =	0.0000

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
lost4						
age	-.0461113	.0129744	-3.55	0.000	-.0715406	-.020682
sex						
Male	-1.192968	.1806428	-6.60	0.000	-1.547022	-.8389148
weight	.1131467	.0108868	10.39	0.000	.0918089	.1344844
wellpgm						
Yes	1.370215	.4048158	3.38	0.001	.5767905	2.163639
_cons	-8.034426	1.199574	-6.70	0.000	-10.38555	-5.683305
completed						
wellpgm						
Yes	.6534203	.2310957	2.83	0.005	.200481	1.10636
experience	-.801973	.0676059	-11.86	0.000	-.9344781	-.6694679
salaried						
Yes	.3955088	.1549943	2.55	0.011	.0917255	.6992921
_cons	4.862419	.4186367	11.61	0.000	4.041906	5.682932
wellpgm						
age	-.0958611	.0071251	-13.45	0.000	-.109826	-.0818963
smoke						
Yes	-1.515911	.1754356	-8.64	0.000	-1.859758	-1.172063
_cons	4.310847	.338842	12.72	0.000	3.646728	4.974965
weight						
sex						
Male	9.501602	.6983151	13.61	0.000	8.13293	10.87028
gym	-.8162669	.0765488	-10.66	0.000	-.9662998	-.666234
_cons	80.09771	.5302486	151.06	0.000	79.05844	81.13697
var(e.weight)	65.98399	3.805168			58.93203	73.8798
corr(e.com~d, e.lost4)	.5236573	.1297834	4.03	0.000	.2268709	.7314522
corr(e.wel~m, e.lost4)	.249717	.2438804	1.02	0.306	-.2493086	.6439525
corr(e.wei~t, e.lost4)	-.6846067	.096236	-7.11	0.000	-.8314263	-.448426
corr(e.wel~m, e.completed)	.3678357	.1636913	2.25	0.025	.014886	.6392761
corr(e.wei~t, e.completed)	-.0821217	.074566	-1.10	0.271	-.2255026	.0647412
corr(e.wei~t, e.wellpgm)	-.0888819	.0671873	-1.32	0.186	-.218281	.0435887

These parameter estimates are pretty close to those from running `eregress` on `weightloss`. But unless you like thinking in terms of shifts along a standardized normal distribution, the coefficient of 1.37 on `wellpgm` is difficult to interpret. We still know that the effect of the program is statistically significant, but little more.

Note that we added `vce(robust)`. This will allow us to treat our sample as a draw from a population when using `estat teffects` and `margins` and thus make inferences about the population. Otherwise, we would be taking the sample as fixed and not as a draw from a population.

If management is thinking about expanding the program, they will want to evaluate its effectiveness. What proportion of employees across all facilities would lose 4 kg or more naturally, either through all employees not participating or through the program simply not being offered? What proportion would lose 4 kg or more if all employees participated? `estat teffects` will estimate these proportions when we request potential-outcome means.

```
. estat teffects, pomean
```

Predictive margins   Number of obs = 545

		Unconditional				
	Margin	std. err.	z	P>|z|	[95% conf. interval]
POmean						
wellpgm						
No	.0844851	.0431001	1.96	0.050	.0000106	.1689597
Yes	.4702298	.086969	5.41	0.000	.2997736	.640686

About 47% are expected to lose 4 kg if everyone participates compared with only 8% when no one participates. More to the point, what is the difference in those averages? We type

```
. estat teffects
```

Predictive margins   Number of obs = 545

		Unconditional				
	Margin	std. err.	z	P>|z|	[95% conf. interval]
ATE						
wellpgm						
(Yes vs No)	.3857447	.1226037	3.15	0.002	.1454458	.6260436

The proportion of employees who would be expected to lose 4 kg increases by 0.39 if everyone participates in the program versus if no one participates; that is the ATE.

What if we consider only program participants? What is the expected average increase in the proportion losing 4 kg? Let's estimate the expected effect of the wellness program on just those employees who choose to participate, the ATET.

```
. estat teffects, atet
```

Predictive margins   Number of obs   = 545
                                                            Subpop. no. obs = 208

		Unconditional				
	Margin	std. err.	z	P>|z|	[95% conf. interval]
ATET						
wellpgm						
(Yes vs No)	.5335926	.1098873	4.86	0.000	.3182174	.7489678

The ATET of 0.53 implies that the program is expected to increase the proportion of employees who lose 4 kg by 0.53 among those who choose to participate across all facilities. Recall that we believed success in the program would be positively correlated with employees' decision to participate. That is what made the decision endogenous. It is not surprising that we expect better results for participants than we do for all the employees as a whole.

We are going to need `margins` to answer some other questions, so let's introduce it by reestimating the ATET.

```
. margins r(0 1).wellpgm if wellpgm, contrast(effects nowald)

Contrasts of predictive margins                         Number of obs = 208
Model VCE: Robust

Expression: Average structural function probability, predict()
```

	Contrast	Delta-method std. err.	z	P>|z|	[95% conf. interval]
wellpgm (Yes vs No)	.5335926	.109483	4.87	0.000	.3190099	.7481752

We have reproduced the estimate.

There is a lot happening in that `margins` command.

- `r(0 1).wellpgm` tells `margins` to form two counterfactuals—one at `wellpgm=0` and another at `wellpgm=1`—and to then take the reference (r) contrast (difference) of those two counterfactuals.

- `if wellpgm` restricts the sample to those who participated in the wellness program.

- `contrast(effects nowald)` tells `margins` to report the z statistic and probability $> z$, which are not shown by default. It also tells `margins` to suppress the overall Wald statistic.

The standard errors are slightly smaller than those from `estat teffects`. If we wanted them to match exactly, we would use the `vce(unconditional)` option with `margins`. That option creates standard errors appropriate to make inferences about the population. The standard errors are so close that we will dispense with `vce(unconditional)` in this section.

Now, let's ask a series of different questions from a different perspective.

The physical trainer for our fictional company is having lunch with a new employee, Betty. The trainer mentions the wellness program, and Betty asks if it is likely to do her much good. Betty looks to be mid thirties and average weight. She says she goes to the gym a couple of times a month. The trainer recalls people with those characteristics doing well with the program. Betty's data are already in the company's database, so the trainer opens Stata on her laptop and types

```
. margins r(0 1).wellpgm if name=="Betty", contrast(effects nowald) noesample
warning: prediction constant over observations.
Contrasts of predictive margins                        Number of obs = 1
Model VCE: Robust

Expression: Average structural function probability, predict()
```

	Contrast	Delta-method std. err.	z	P>|z|	[95% conf. interval]
wellpgm					
(Yes vs No)	.5894013	.222732	2.65	0.008	.1528545 1.025948

The trainer tells Betty that employees with her characteristics increase their chances of losing 4 kg by about 59 percentage points when they are in the program.

Later, another new employee, Fred, asks whether the program is likely to help him lose that last few kilograms. He is thin, in his upper fifties, and he already goes to the gym about twice a week. Our trainer types

```
. margins r(0 1).wellpgm if name=="Fred", contrast(effects nowald) noesample
warning: prediction constant over observations.
Contrasts of predictive margins                        Number of obs = 1
Model VCE: Robust

Expression: Average structural function probability, predict()
```

	Contrast	Delta-method std. err.	z	P>|z|	[95% conf. interval]
wellpgm					
(Yes vs No)	.0147468	.0226186	0.65	0.514	−.0295849 .0590785

She tells Fred that the program might be good for him but not to expect it to create much weight loss. Fred says he would like to sign up, just so he can meet some other employees.

When Fred leaves, our trainer calls her office mate and makes a wager that Fred will not lose 4 kg on the program. The trainer then realizes that she placed a bet on overall weight loss, not just the loss attributable to the wellness program. To be certain, she checks the potential outcomes of weight loss for Fred being in the program and for Fred being out of the program.

```
. margins i(0 1).wellpgm if name=="Fred", noesample
warning: prediction constant over observations.
Predictive margins                                     Number of obs = 1
Model VCE: Robust

Expression: Average structural function probability, predict()
```

	Margin	Delta-method std. err.	z	P>|z|	[95% conf. interval]
wellpgm					
No	.0000218	.0000382	0.57	0.569	−.0000532 .0000967
Yes	.0147686	.022641	0.65	0.514	−.029607 .0591441

With a negligible chance of losing 4 kg if Fred chooses not to participate and a slim 1.5% chance if Fred does participate, our trainer feels pretty good about her wager. Even the upper bound of the confidence intervals makes the trainer confident. Of course, these are the expected results for all employees with Fred's characteristics; Fred might be an overachiever.

The counterfactuals and contrasts that we computed for Betty and Fred are the expected values from our model conditioned on the exogenous covariates in the main equation, `age` and `sex`, on the level of endogeneity for the endogenous equations for `weight` and `wellpgm`, and on fixing the values of `wellpgm` first to 0 and then to 1. The estimates for Betty are what we would expect if we averaged over hundreds of employees who match Betty's `age`, `sex`, and level of endogeneity. The same applies to Fred.

Also note that we typed `r(0 1).`, rather than just `r..`. That is because we are operating on a single observation, and `margins` cannot determine the appropriate levels of `wellpgm` for which to form counterfactuals. We had to tell `margins` to use 0 and 1.

It is unlikely that our trainer has Stata on her laptop or has the inclination to type `margins` commands. As analysts, however, we might create a table for her that she can use to assess candidates and help employees form realistic expectations.

Our dataset already has grouping variables for `age`, `gym`, `weight`, and `sex`. We can estimate the expected additional probability of losing more than 4 kg for each combination of these groups by using an `over()` option.

```
. margins r.wellpgm, contrast(effects nowald) over(agegrp gymgrp wtgrp sex)

Contrasts of predictive margins                         Number of obs = 545
Model VCE: Robust

Expression: Average structural function probability, predict()
Over:       agegrp gymgrp wtgrp sex
```

	Contrast	Delta-method std. err.	z	P>|z|	[95% conf. interval]
wellpgm@ agegrp# gymgrp# wtgrp#sex (Yes vs No) 20–29 0 < 60 Female	0	(omitted)			
(Yes vs No) 20–29 0 < 60 Male	0	(omitted)			
(Yes vs No) 20–29 0 60–69 Female	.6437056	.1621665	3.97	0.000	.325865	.9615462
(Yes vs No) 20–29 0 60–69 Male	0	(omitted)			

(output omitted)

Those rows marked (`omitted`) represent combinations of characteristics for which we do not have any employees in our sample. We could use our model to extrapolate to those groups, but we are not going to do that. What we do have for each combination of groups is an estimate of the expected increase in the probability of losing 4 kg, a test that the probability is greater than 0, and a 95% confidence interval.

Those results will take a lot of transcription to create something compact for the trainer. And while our hearts are warmed by the tests and confidence intervals, the trainer might not feel the same way. If we wanted to be exceptionally helpful, we could build a table manually showing ATEs for each group.

```
. predict te, te
. table (agegrp gymgrp) (sex wtgrp), statistic(mean te) nformat(%4.2f) nototals
```

		Female					Male		
		Weight groups					Weight groups		
	< 60	60–69	70–79	80–89	90 up	60–69	70–79	80–89	90 up
Age groups									
20–29									
Gym visit groups									
0		0.64	0.58	0.53	0.41		0.63	0.62	0.52
0–5		0.65	0.63	0.57			0.64	0.65	0.59
6–10		0.51	0.64	0.65			0.55	0.58	0.64
11 up		0.40					0.35		
30–39									
Gym visit groups									
0	0.48		0.65	0.62	0.61		0.61	0.65	0.62
0–5		0.48	0.64	0.64				0.56	0.62
6–10	0.21	0.39	0.54	0.60		0.27	0.31	0.51	0.48
11 up		0.34	0.43				0.24		0.53
40–49									
Gym visit groups									
0		0.45	0.57	0.62	0.65		0.33	0.50	0.62
0–5		0.39	0.48	0.59	0.61		0.27	0.38	0.55
6–10		0.22	0.33	0.38		0.09	0.14	0.25	0.42
11 up	0.07	0.10		0.28		0.09	0.08	0.19	
50–59									
Gym visit groups									
0		0.26	0.35	0.46	0.62		0.25	0.29	0.47
0–5	0.06		0.22	0.40	0.62		0.12	0.20	0.32
6–10	0.05	0.04	0.22	0.16	0.39		0.05	0.10	0.13
11 up	0.01	0.01	0.05			0.01	0.01		
60 up									
Gym visit groups									
0			0.17	0.25	0.37		0.07	0.12	0.26
0–5			0.07	0.33	0.28	0.02		0.03	0.11
6–10		0.02	0.04	0.10		0.01	0.02	0.03	0.08
11 up	0.00		0.02				0.00	0.02	0.03

We first predicted the expected treatment effects for each observation in our sample. Then, we let `table` average those values for each combination of groups. For any combination of groups, these estimates match those from `estat teffects` for each group.

Here we have demonstrated how you can use `estat teffects` and `margins` to answer a variety of interesting questions after fitting a model with an endogenous covariate, endogenous treatment, and endogenous sample selection. ERMs can address one additional complication—within-panel or within-group correlation—if we fit a random-effects model with `xteregress`, `xteintreg`, `xteprobit`, or `xteoprobit`. You can use `estat teffects` and `margins` to interpret results of random-effects models as well. See [ERM] **Example 7** and [ERM] **Example 9**.

Video examples

Extended regression models, part 4: Interpreting the model

Extended regression models, part 3: Endogenous sample selection

Extended regression models, part 2: Nonrandom treatment assignment

Extended regression models, part 1: Endogenous covariates

References

Cameron, A. C., and P. K. Trivedi. 2005. *Microeconometrics: Methods and Applications*. New York: Cambridge University Press.

———. 2022. *Microeconometrics Using Stata*. 2nd ed. College Station, TX: Stata Press.

Gould, W. W. 2018. Ermistatas and Stata's new ERMs commands. *The Stata Blog: Not Elsewhere Classified*. https://blog.stata.com/2018/03/27/ermistatas-and-statas-new-erms-commands/.

Roodman, D. 2011. Fitting fully observed recursive mixed-process models with cmp. *Stata Journal* 11: 159–206.

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

Also see

[ERM] **Intro 1** — An introduction to the ERM commands

[ERM] **Intro 7** — Model interpretation

[ERM] **Glossary**

Title

---

eintreg — Extended interval regression

Description

eintreg fits an interval regression model that accommodates any combination of endogenous co-variates, nonrandom treatment assignment, and endogenous sample selection. Continuous, binary, and ordinal endogenous covariates are allowed. Treatment assignment may be endogenous or exogenous. A probit or tobit model may be used to account for endogenous sample selection.

xteintreg fits a random-effects interval regression model that accommodates endogenous covariates, treatment, and sample selection in the same way as eintreg and also accounts for correlation of observations within panels or within groups.

Quick start

All quick start examples use an interval-measured dependent variable with the interval's lower bound recorded in variable y_l and its upper bound recorded in y_u.

Regression of $[y\_l, y\_u]$ on x with continuous endogenous covariate y2 modeled by x and z

    eintreg y_l y_u x, endogenous(y2 = x z)

Same as above, but adding continuous endogenous covariate y3 modeled by x and z2

    eintreg y_l y_u x, endogenous(y2 = x z) endogenous(y3 = x z2)

Regression of $[y\_l, y\_u]$ on x with binary endogenous covariate d modeled by x and z

    eintreg y_l y_u x, endogenous(d = x z, probit)

Regression of $[y\_l, y\_u]$ on x with endogenous treatment recorded in trtvar and modeled by x and z

    eintreg y_l y_u x, entreat(trtvar = x z)

Regression of $[y\_l, y\_u]$ on x with exogenous treatment recorded in trtvar

    eintreg y_l y_u x, extreat(trtvar)

Random-effects regression of $[y\_l, y\_u]$ on x using [xtset](#) data

    xteintreg y_l y_u x

Regression of $[y\_l, y\_u]$ on x with endogenous sample-selection indicator selvar modeled by x and z

    eintreg y_l y_u x, select(selvar = x z)

Same as above, but adding endogenous covariate y2 modeled by x and z2

    eintreg y_l y_u x, select(selvar = x z) endogenous(y2 = x z2)

Same as above, but adding endogenous treatment recorded in trtvar and modeled by x and z3

```
eintreg y_l y_u x, select(selvar = x z) endogenous(y2 = x z2)  ///
    entreat(trtvar = x z3)
```

Same as above, but with random effects and without endogenous treatment

```
xteintreg y_l y_u x, select(selvar = x z) endogenous(y2 = x z2)
```

Menu

eintreg

Statistics > Endogenous covariates > Models adding selection and treatment > Interval regression

xteintreg

Statistics > Longitudinal/panel data > Endogenous covariates > Models adding selection and treatment > Interval regression (RE)

Syntax

Basic interval regression with endogenous covariates

eintreg *depvar*$_1$ *depvar*$_2$ [*indepvars*] , <u>endog</u>enous(*depvars*$_{en}$ = *varlist*$_{en}$) [*options*]

Basic interval regression with endogenous treatment assignment

eintreg *depvar*$_1$ *depvar*$_2$ [*indepvars*] , <u>entr</u>eat(*depvar*$_{tr}$ [= *varlist*$_{tr}$]) [*options*]

Basic interval regression with exogenous treatment assignment

eintreg *depvar*$_1$ *depvar*$_2$ [*indepvars*] , <u>extr</u>eat(*tvar*) [*options*]

Basic interval regression with sample selection

eintreg *depvar*$_1$ *depvar*$_2$ [*indepvars*] , <u>sel</u>ect(*depvar*$_s$ = *varlist*$_s$) [*options*]

Basic interval regression with tobit sample selection

eintreg *depvar*$_1$ *depvar*$_2$ [*indepvars*] , <u>tobitsel</u>ect(*depvar*$_s$ = *varlist*$_s$) [*options*]

Basic interval regression with random effects

xteintreg *depvar*$_1$ *depvar*$_2$ [*indepvars*] [, *options*]

Interval regression combining endogenous covariates, treatment, and selection

eintreg *depvar*$_1$ *depvar*$_2$ [*indepvars*] [*if*] [*in*] [*weight*] [, *extensions options*]

Interval regression combining random effects, endogenous covariates, treatment, and selection

xteintreg *depvar*$_1$ *depvar*$_2$ [*indepvars*] [*if*] [*in*] [, *extensions options*]

depvar$_1$ and *depvar*$_2$ should have the following form:

Type of data		*depvar*$_1$	*depvar*$_2$
point data	$a = [a, a]$	a	a
interval data	$[a, b]$	a	b
left-censored data	$(-\infty, b]$	.	b
right-censored data	$[a, +\infty)$	a	.
missing		.	.

extensions	Description
Model	
endogenous(*enspec*)	model for endogenous covariates; may be repeated
entreat(*entrspec*)	model for endogenous treatment assignment
extreat(*extrspec*)	exogenous treatment
select(*selspec*)	probit model for selection
tobitselect(*tselspec*)	tobit model for selection

options	Description
Model	
noconstant	suppress constant term
offset(*varname_o*)	include *varname_o* in model with coefficient constrained to 1
constraints(*numlist*)	apply specified linear constraints
SE/Robust	
vce(*vcetype*)	*vcetype* may be oim, robust, cluster *clustvar*, opg, bootstrap, or jackknife
Reporting	
level(*#*)	set confidence level; default is level(95)
nocnsreport	do not display constraints
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Integration	
intpoints(*#*)	set the number of integration (quadrature) points for integration over four or more dimensions; default is intpoints(128)
triintpoints(*#*)	set the number of integration (quadrature) points for integration over three dimensions; default is triintpoints(10)
reintpoints(*#*)	set the number of integration (quadrature) points for random-effects integration; default is reintpoints(7)
reintmethod(*intmethod*)	integration method for random effects; *intmethod* may be mvaghermite (the default) or ghermite
Maximization	
maximize_options	control the maximization process; seldom used
collinear	keep collinear variables
coeflegend	display legend instead of statistics

enspec is *depvars*$_{\text{en}}$ = *varlist*$_{\text{en}}$ [, *enopts*]

  where *depvars*$_{\text{en}}$ is a list of endogenous covariates. Each variable in *depvars*$_{\text{en}}$ specifies an endogenous covariate model using the common *varlist*$_{\text{en}}$ and options.

entrspec is *depvar*$_{\text{tr}}$ [= *varlist*$_{\text{tr}}$] [, *entropts*]

  where *depvar*$_{\text{tr}}$ is a variable indicating treatment assignment. *varlist*$_{\text{tr}}$ is a list of covariates predicting treatment assignment.

extrspec is *tvar* [, *extropts*]

where *tvar* is a variable indicating treatment assignment.

selspec is *depvar$_s$* = *varlist$_s$* [, *selopts*]

where *depvar$_s$* is a variable indicating selection status. *depvar$_s$* must be coded as 0, indicating that the observation was not selected, or 1, indicating that the observation was selected. *varlist$_s$* is a list of covariates predicting selection.

tselspec is *depvar$_s$* = *varlist$_s$* [, *tselopts*]

where *depvar$_s$* is a continuous variable. *varlist$_s$* is a list of covariates predicting *depvar$_s$*. The censoring status of *depvar$_s$* indicates selection, where a censored *depvar$_s$* indicates that the observation was not selected and a noncensored *depvar$_s$* indicates that the observation was selected.

enopts	Description
Model	
<u>pro</u>bit	treat endogenous covariate as binary
<u>opro</u>bit	treat endogenous covariate as ordinal
<u>pov</u>ariance	estimate a different variance for each level of a binary or an ordinal endogenous covariate
<u>poc</u>orrelation	estimate different correlations for each level of a binary or an ordinal endogenous covariate
<u>noma</u>in	do not add endogenous covariate to main equation
<u>nore</u>	do not include random effects in model for endogenous covariate
<u>nocons</u>tant	suppress constant term

`nore` is available only with `xteintreg`.

entropts	Description
Model	
<u>pov</u>ariance	estimate a different variance for each potential outcome
<u>poc</u>orrelation	estimate different correlations for each potential outcome
<u>noma</u>in	do not add treatment indicator to main equation
<u>noint</u>eract	do not interact treatment with covariates in main equation
<u>nore</u>	do not include random effects in model for endogenous treatment
<u>nocons</u>tant	suppress constant term
<u>off</u>set(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1

`nore` is available only with `xteintreg`.

extropts	Description
Model	
<u>pov</u>ariance	estimate a different variance for each potential outcome
<u>poc</u>orrelation	estimate different correlations for each potential outcome
<u>noma</u>in	do not add treatment indicator to main equation
<u>noint</u>eract	do not interact treatment with covariates in main equation

selopts	Description
Model	
<u>nore</u>	do not include random effects in selection model
<u>nocon</u>stant	suppress constant term
<u>off</u>set(*varname_o*)	include *varname_o* in model with coefficient constrained to 1

nore is available only with xteintreg.

tselopts	Description
Model	
*ll(*varname* │ #)	left-censoring variable or limit
*ul(*varname* │ #)	right-censoring variable or limit
main	add censored selection variable to main equation
<u>nore</u>	do not include random effects in tobit selection model
<u>nocon</u>stant	suppress constant term
<u>off</u>set(*varname_o*)	include *varname_o* in model with coefficient constrained to 1

* You must specify either ll() or ul().

nore is available only with xteintreg.

indepvars, *varlist_en*, *varlist_tr*, and *varlist_s* may contain factor variables; see [U] **11.4.3 Factor variables**.

depvar_1, *depvar_2*, *indepvars*, *depvars_en*, *varlist_en*, *depvar_tr*, *varlist_tr*, *tvar*, *depvar_s*, and *varlist_s* may contain time-series operators; see [U] **11.4.4 Time-series varlists**.

bootstrap, by, collect, jackknife, and statsby are allowed with eintreg and xteintreg. rolling and svy are allowed with eintreg. See [U] **11.1.10 Prefix commands**.

Weights are not allowed with the bootstrap prefix; see [R] **bootstrap**.

vce() and weights are not allowed with the svy prefix; see [SVY] **svy**.

fweights, iweights, and pweights are allowed with eintreg; see [U] **11.1.6 weight**.

reintpoints() and reintmethod() are available only with xteintreg.

collinear and coeflegend do not appear in the dialog box.

See [U] **20 Estimation and postestimation commands** for more capabilities of estimation commands.

Options

> **Model**

endogenous(*enspec*), entreat(*entrspec*), extreat(*extrspec*), select(*selspec*),
    tobitselect(*tselspec*); see [ERM] **ERM options**.

noconstant, offset(*varname_o*), constraints(*numlist*); see [R] **Estimation options**.

> **SE/Robust**

vce(*vcetype*); see [ERM] **ERM options**.

> **Reporting**

level(#), nocnsreport; see [R] **Estimation options**.

display_options: noci, <u>nop</u>values, <u>noo</u>mitted, vsquish, noemptycells, <u>base</u>levels,
    <u>allbase</u>levels, <u>nofvlab</u>el, fvwrap(#), fvwrapon(*style*), cformat(*%fmt*), pformat(*%fmt*),
    sformat(*%fmt*), and nolstretch; see [R] **Estimation options**.

    Integration

intpoints(*#*), triintpoints(*#*), reintpoints(*#*), reintmethod(*intmethod*); see [ERM] **ERM options**.

    Maximization

maximize_options: <u>d</u>ifficult, <u>tech</u>nique(*algorithm_spec*), <u>iter</u>ate(*#*), [no]log, <u>trace</u>, gradient, showstep, <u>hess</u>ian, <u>showtol</u>erance, <u>tol</u>erance(*#*), <u>ltol</u>erance(*#*), <u>nrtol</u>erance(*#*), <u>nonrtol</u>erance, and from(*init_specs*); see [R] **Maximize**.

The default technique for eintreg is technique(nr). The default technique for xteintreg is technique(bhhh 10 nr 2).

Setting the optimization type to technique(bhhh) resets the default *vcetype* to vce(opg).

The following options are available with eintreg and xteintreg but are not shown in the dialog box:

collinear, coeflegend; see [R] **Estimation options**.

Remarks and examples

    eintreg and xteintreg fit models that we refer to as "extended interval regression models", meaning that they accommodate endogenous covariates, nonrandom treatment assignment, endogenous sample selection, and panel data or other grouped data.

    eintreg fits models for cross-sectional data (one-level models). eintreg can account for endogenous covariates, treatment, and sample selection, whether these complications arise individually or in combination.

    xteintreg fits random-effects models (two-level models) for panel data or grouped data. xteintreg accounts for endogenous covariates, treatment, and sample selection in the same way as eintreg and also accounts for within-panel or within-group correlation among observations.

    In this entry, you will find information on the syntax for the eintreg and xteintreg commands. You can see *Methods and formulas* for a full description of the models that can be fit with these commands and for details about how those models are fit.

    More information on extended interval regression models is found in the separate introductions and example entries. We recommend reading those entries to learn how to use eintreg and xteintreg. Below, we provide a guide to help you locate the ones that will be helpful to you.

    For an introduction to eintreg, xteintreg, and the other extended regression commands for continuous, binary, and ordinal outcomes, see [ERM] **Intro 1**–[ERM] **Intro 9**.

    [ERM] **Intro 1** introduces the ERM commands, the problems they address, and their syntax.

    [ERM] **Intro 2** provides background on the four types of models—linear regression, interval regression, probit regression, and ordered probit regression—that can be fit using ERM commands. This intro also demonstrates how to fit tobit models using eintreg by transforming your dependent variable into the required format. This same transformation can be used to fit random-effects tobit models with xteintreg.

    [ERM] **Intro 3** considers the problem of endogenous covariates and how to solve it using ERM commands.

    [ERM] **Intro 4** gives an overview of endogenous sample selection and using ERM commands to account for it.

[ERM] **Intro 5** covers nonrandom treatment assignment and how to account for it using `eintreg` or any of the other ERM commands.

[ERM] **Intro 6** covers random-effects models for panel data and other grouped data. It discusses `xteintreg` and the other ERM commands for panel data.

[ERM] **Intro 7** discusses interpretation of results. You can interpret coefficients from `eintreg` and `xteintreg` in the usual way, but this introduction goes beyond the interpretation of coefficients. We demonstrate how to find answers to interesting questions by using `margins`. If your model includes an endogenous covariate or an endogenous treatment, the use of `margins` differs from its use after other estimation commands, so we strongly recommend reading this intro if you are fitting these types of models.

[ERM] **Intro 8** will be helpful if you are familiar with `ivtobit`, `xtintreg`, `xttobit`, and other commands that address endogenous covariates, sample selection, nonrandom treatment assignment, or panel data. This introduction is a Rosetta stone that maps the syntax of those commands to the syntax of `eintreg` and `xteintreg`. If your outcome is stored in a single variable, which is necessary with `tobit`, `ivtobit`, and `xttobit`, the Rosetta stone also demonstrates how to create the lower- and upper-bound variables necessary for `eintreg` and `xteintreg`.

[ERM] **Intro 9** walks you through an example that gives insight into the concepts of endogenous covariates, treatment assignment, and sample selection while fitting models with `eregress` that address these complications. Although the example uses `eregress`, the discussion applies equally to `eintreg`. This intro also demonstrates how to interpret results by using `margins` and `estat teffects`.

Additional examples are presented in [ERM] **Example 1a**–[ERM] **Example 9**. For examples using `eintreg`, see

      [ERM] **Example 1b**   Interval regression with continuous endogenous covariate
      [ERM] **Example 1c**   Interval regression with endogenous covariate and sample selection

See *Examples* in [ERM] **Intro** for an overview of all the examples. All examples may be interesting because they handle complications in the same way. Examples using `eregress` and `xteregress` will be of particular interest because results of models fit by `eintreg` and `xteintreg` are interpreted in the same way.

`eintreg` and `xteintreg` fit many models discussed in the literature. For instance, the tobit model was originally conceived in Tobin (1958) as a model of consumption of consumer durables, where purchases were left-censored at 0. Wooldridge (2020, sec. 17.4) introduces censored and truncated regression models. Cameron and Trivedi (2022, chap. 19) discuss the tobit model using Stata examples. `eintreg` can also fit models like the tobit regression model with continuous endogenous regressors (Newey 1987) and the censored regression model with binary endogenous regressors (Angrist 2001). `xteintreg` can fit the random-effects tobit model discussed in (Wooldridge 2010, sec. 17.8). Roodman (2011) investigated interval regression models with endogenous covariates and endogenous sample selection and demonstrated how multiple observational data complications could be addressed with a triangular model structure. He and Tamás Bartus showed how random effects could be used in the triangular model structure in Bartus and Roodman (2014). Roodman's work has been used to model processes like the effect of innovation on labor productivity (Mairesse and Robin 2009) and the effect of insect-resistant crops on pesticide demand (Fernandez-Cornejo and Wechsler 2012).

Stored results

eintreg stores the following in e():

Scalars
e(N)	number of observations
e(N_selected)	number of selected observations
e(N_nonselected)	number of nonselected observations
e(N_unc)	number of uncensored observations
e(N_lc)	number of left-censored observations
e(N_rc)	number of right-censored observations
e(N_int)	number of interval-censored observations
e(k)	number of parameters
e(k_cat#)	number of categories for the #th *depvar*, ordinal
e(k_eq)	number of equations in e(b)
e(k_eq_model)	number of equations in overall model test
e(k_dv)	number of dependent variables
e(k_aux)	number of auxiliary parameters
e(df_m)	model degrees of freedom
e(ll)	log likelihood
e(N_clust)	number of clusters
e(chi2)	χ^2
e(p)	p-value for model test
e(n_quad)	number of integration points for multivariate normal
e(n_quad3)	number of integration points for trivariate normal
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise

Macros
e(cmd)	eintreg
e(cmdline)	command as typed
e(depvar)	names of dependent variables
e(tsel_ll)	left-censoring limit for tobit selection
e(tsel_ul)	right-censoring limit for tobit selection
e(wtype)	weight type
e(wexp)	weight expression
e(title)	title in estimation output
e(clustvar)	name of cluster variable
e(offset#)	offset for the #th *depvar*, where # is determined by equation order in output
e(chi2type)	Wald; type of model χ^2 test
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. err.
e(opt)	type of optimization
e(which)	max or min; whether optimizer is to perform maximization or minimization
e(ml_method)	type of ml method
e(user)	name of likelihood-evaluator program
e(technique)	maximization technique
e(properties)	b V
e(estat_cmd)	program used to implement estat
e(predict)	program used to implement predict
e(marginsok)	predictions allowed by margins
e(marginsnotok)	predictions disallowed by margins
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices
e(b)	coefficient vector
e(cat#)	categories for the #th *depvar*, ordinal
e(Cns)	constraints matrix
e(ilog)	iteration log (up to 20 iterations)
e(gradient)	gradient vector

| e(V) | variance–covariance matrix of the estimators |
| e(V_modelbased) | model-based variance |

Functions
| e(sample) | marks estimation sample |

In addition to the above, the following is stored in r():

Matrices
| r(table) | matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals |

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

xteintreg stores the following in e():

Scalars
e(N)	number of observations
e(N_g)	number of groups
e(N_selected)	number of selected observations
e(N_nonselected)	number of nonselected observations
e(N_unc)	number of uncensored observations
e(N_lc)	number of left-censored observations
e(N_rc)	number of right-censored observations
e(N_int)	number of interval-censored observations
e(k)	number of parameters
e(k_cat#)	number of categories for the #th *depvar*, ordinal
e(k_eq)	number of equations in e(b)
e(k_eq_model)	number of equations in overall model test
e(k_dv)	number of dependent variables
e(k_aux)	number of auxiliary parameters
e(df_m)	model degrees of freedom
e(ll)	log likelihood
e(N_clust)	number of clusters
e(chi2)	χ^2
e(p)	p-value for model test
e(n_quad)	number of integration points for multivariate normal
e(n_quad3)	number of integration points for trivariate normal
e(n_requad)	number of integration points for random effects
e(g_min)	smallest group size
e(g_avg)	average group size
e(g_max)	largest group size
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise

Macros
e(cmd)	xteintreg
e(cmdline)	command as typed
e(depvar)	names of dependent variables
e(tsel_ll)	left-censoring limit for tobit selection
e(tsel_ul)	right-censoring limit for tobit selection
e(ivar)	variable denoting groups
e(title)	title in estimation output
e(clustvar)	name of cluster variable
e(offset#)	offset for the #th *depvar*, where # is determined by equation order in output
e(chi2type)	Wald; type of model χ^2 test
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. err.
e(reintmethod)	integration method for random effects
e(opt)	type of optimization
e(which)	max or min; whether optimizer is to perform maximization or minimization

e(ml_method)	type of ml method
e(user)	name of likelihood-evaluator program
e(technique)	maximization technique
e(properties)	b V
e(estat_cmd)	program used to implement estat
e(predict)	program used to implement predict
e(marginsok)	predictions allowed by margins
e(marginsnotok)	predictions disallowed by margins
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices
e(b)	coefficient vector
e(cat#)	categories for the #th *depvar*, ordinal
e(Cns)	constraints matrix
e(ilog)	iteration log (up to 20 iterations)
e(gradient)	gradient vector
e(V)	variance–covariance matrix of the estimators
e(V_modelbased)	model-based variance

Functions
e(sample)	marks estimation sample

In addition to the above, the following is stored in r():

Matrices
r(table)	matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Methods and formulas

The methods and formulas presented here are for the interval model. The estimators implemented in eintreg and xteintreg are maximum likelihood estimators covered by the results in chapter 13 of Wooldridge (2010) and White (1996).

The log-likelihood functions maximized by eintreg and xteintreg are implied by the triangular structure of the model. Specifically, the joint distribution of the endogenous variables is a product of conditional and marginal distributions because the model is triangular. For a few of the many relevant applications of this result in literature, see chapter 10 of Amemiya (1985); Heckman (1976, 1979); chapter 5 of Maddala (1983); Maddala and Lee (1976); sections 15.7.2, 15.7.3, 16.3.3, 17.5.2, and 19.7.1 in Wooldridge (2010); and Wooldridge (2014). Roodman (2011) and Bartus and Roodman (2014) used this result to derive the formulas discussed below.

Methods and formulas are presented under the following headings:

Introduction

A regression model of outcome y_i on covariates $\mathbf{x}_i$ may be written as

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \epsilon_i$$

where ϵ_i is normal with mean 0 and variance σ^2. Instead of observing y_i, we observe the endpoints y_{li} and y_{ui}.

If y_i is left-censored, the lower endpoint $y_{li} = -\infty$ and we know that $y_i \leq y_{ui}$. If y_i is right-censored, the upper endpoint $y_{ui} = +\infty$ and we know that $y_i \geq y_{li}$. If there is no censoring, $y_{li} = y_{ui} = y_i$. When y_{li} and y_{ui} are real valued and not equal, we know that $y_{li} \leq y_i \leq y_{ui}$.

The log likelihood is

$$
\begin{aligned}
\ln L = &\sum_{i \in U} w_i \ln \phi\big(y_i - \mathbf{x}_i\boldsymbol{\beta}, \sigma^2\big) \\
&+ \sum_{i \in L} w_i \ln \Phi\left(\frac{y_{ui} - \mathbf{x}_i\boldsymbol{\beta}}{\sigma}\right) \\
&+ \sum_{i \in R} w_i \ln \Phi\left(\frac{-y_{li} + \mathbf{x}_i\boldsymbol{\beta}}{\sigma}\right) \\
&+ \sum_{i \in I} w_i \ln \left\{ \Phi\left(\frac{y_{ui} - \mathbf{x}_i\boldsymbol{\beta}}{\sigma}\right) - \Phi\left(\frac{y_{li} - \mathbf{x}_i\boldsymbol{\beta}}{\sigma}\right) \right\}
\end{aligned}
$$

where U is the set of observations where y_i is not censored, L is the set of observations where y_i is left-censored, R is the set of observations where y_i is right-censored, I is the set of observations where y_i is interval-censored, and w_i are the weights.

The conditional mean of y_i is

$$E(y_i|\mathbf{x}_i) = \mathbf{x}_i\boldsymbol{\beta}$$

If we wished to condition on the censoring, we could calculate an expectation on $y_i^\star = \max\{y_{li}, \min(y_{ij}, y_{ui})\}$ or a constrained mean $E(y_i|y_{li} < y_i < y_{ui})$. See *Predictions using the full model* in [ERM] **eprobit postestimation** for details on how this is done.

If you are willing to take our word for some derivations and notation, the following is complete. Longer explanations and derivations for some terms and functions are provided in *Methods and formulas* of [ERM] **eprobit**. For example, we need the two-sided probability function $\Phi_d^\star$ that is discussed in *Introduction* in [ERM] **eprobit**.

If you are interested in all the details, we suggest you read *Methods and formulas* of [ERM] **eprobit** in its entirety before reading this section. Here we mainly show how the complications that arise in ERMs are handled in an interval regression framework.

Endogenous covariates

Continuous endogenous covariates

An interval regression of y_i on exogenous covariates $\mathbf{x}_i$ and C continuous endogenous covariates $\mathbf{w}_{ci}$ has the form

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{w}_{ci}\boldsymbol{\beta}_c + \epsilon_i$$

$$\mathbf{w}_{ci} = \mathbf{z}_{ci}\mathbf{A}_c + \boldsymbol{\epsilon}_{ci}$$

As in *Introduction*, we do not observe y_i but instead observe the endpoints y_{li} and y_{ui}. The vector $\mathbf{z}_{ci}$ contains variables from $\mathbf{x}_i$ and other covariates that affect $\mathbf{w}_{ci}$. For the model to be identified, $\mathbf{z}_{ci}$ must contain one extra exogenous covariate not in $\mathbf{x}_i$ for each of the endogenous regressors in $\mathbf{w}_{ci}$. The unobserved errors ϵ_i and $\boldsymbol{\epsilon}_{ci}$ are multivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma^2 & \boldsymbol{\sigma}'_{1c} \\ \boldsymbol{\sigma}_{1c} & \boldsymbol{\Sigma}_c \end{bmatrix}$$

Conditional on the endogenous and exogenous covariates, ϵ_i has mean and variance

$$\mu_{1|c,i} = E\left(\epsilon_i | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{ci}\right) = \boldsymbol{\sigma}'_{1c}\boldsymbol{\Sigma}_c^{-1}\left(\mathbf{w}_{ci} - \mathbf{z}_{ci}\mathbf{A}_c\right)'$$

$$\sigma^2_{1|c} = \text{Var}\left(\epsilon_i | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{ci}\right) = \sigma^2 - \boldsymbol{\sigma}'_{1c}\boldsymbol{\Sigma}_c^{-1}\boldsymbol{\sigma}_{1c}$$

Let

$$r_{li} = y_{li} - \mathbf{x}_i\boldsymbol{\beta} - \mathbf{w}_{ci}\boldsymbol{\beta}_c - \mu_{1|c,i}$$

$$r_{ui} = y_{ui} - \mathbf{x}_i\boldsymbol{\beta} - \mathbf{w}_{ci}\boldsymbol{\beta}_c - \mu_{1|c,i}$$

The log likelihood is

$$\begin{aligned} \ln L = &\sum_{i \in U} w_i \ln \phi\left(r_{li}, \sigma^2_{1|c}\right) \\ &+ \sum_{i \in L} w_i \ln \Phi_1^*\left(-\infty, r_{ui}, \sigma^2_{1|c}\right) \\ &+ \sum_{i \in R} w_i \ln \Phi_1^*\left(r_{li}, \infty, \sigma^2_{1|c}\right) \\ &+ \sum_{i \in I} w_i \ln \Phi_1^*\left(r_{li}, r_{ui}, \sigma^2_{1|c}\right) \\ &+ \sum_{i=1}^{N} w_i \ln \phi_C(\mathbf{w}_{ci} - \mathbf{z}_{ci}\mathbf{A}_c, \boldsymbol{\Sigma}_c) \end{aligned}$$

where U is the set of observations where y_i is not censored, L is the set of observations where y_i is left-censored, R is the set of observations where y_i is right-censored, and I is the set of observations where y_i is interval-censored.

The conditional mean of y_i is

$$E(y_i | \mathbf{x}_i, \mathbf{w}_{ci}, \mathbf{z}_{ci}) = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{w}_{ci}\boldsymbol{\beta}_c + \boldsymbol{\sigma}'_{1c}\boldsymbol{\Sigma}_c^{-1}(\mathbf{w}_{ci} - \mathbf{z}_{ci}\mathbf{A}_c)'$$

Binary and ordinal endogenous covariates

Here we begin by formulating the interval regression of y_i on exogenous covariates $\mathbf{x}_i$ and B binary and ordinal endogenous covariates $\mathbf{w}_{bi} = [w_{b1i}, \ldots, w_{bBi}]$. Indicator (dummy) variables for the levels of each binary and ordinal covariate are used in the model. You can also interact other covariates with the binary and ordinal endogenous covariates, as in treatment-effect models.

The binary and ordinal endogenous covariates $\mathbf{w}_{bi}$ are formulated as in *Binary and ordinal endogenous covariates* in [ERM] **eprobit**.

The model for the outcome can be formulated with or without different variance and correlation parameters for each level of $\mathbf{w}_{bi}$. Level-specific parameters are obtained by specifying povariance or pocorrelation in the endogenous() option.

If the variance and correlation parameters are not level specific, we have

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB} + \epsilon_i$$

The $\mathbf{wind}_{bji}$ vectors are defined in *Binary and ordinal endogenous covariates* in [ERM] **eprobit**. As in *Introduction*, we do not observe y_i but instead observe the endpoints y_{li} and y_{ui}. The binary and ordinal endogenous errors $\epsilon_{b1i}, \ldots, \epsilon_{bBi}$ and outcome error ϵ_i are multivariate normal with 0 mean and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_b & \boldsymbol{\sigma}_{1b} \\ \boldsymbol{\sigma}'_{1b} & \sigma^2 \end{bmatrix}$$

From here, we discuss the model with ordinal endogenous covariates. The results for binary endogenous covariates are similar.

As in *Binary and ordinal endogenous covariates* in [ERM] **eregress**, for the uncensored observations, we write the joint density of y_i and $\mathbf{w}_{bi}$ using the conditional density of $\epsilon_{b1i}, \ldots, \epsilon_{bBi}$ on ϵ_i. For the censored observations, we use tools discussed in *Likelihood for multiequation models* in [ERM] **eprobit** to formulate the joint density directly.

For $i \in U$, the uncensored observations, define

$$r_i = y_i - (\mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB})$$

For the censored observations, define

$$r_{li} = y_{li} - (\mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB})$$
$$r_{ui} = y_{ui} - (\mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB})$$

Let

$$\boldsymbol{\Sigma}_{b|1} = \boldsymbol{\Sigma} - \frac{\boldsymbol{\sigma}_{1b}\boldsymbol{\sigma}'_{1b}}{\sigma^2}$$

Now the log likelihood is

$$
\ln L = \sum_{i \in U} w_i \ln \left\{ \Phi_B^*(\mathbf{l}_i, \mathbf{u}_i, \mathbf{\Sigma}_{b|1}) \, \phi(r_i, \sigma^2) \right\}
$$

$$
+ \sum_{i \in L} w_i \ln \Phi_{B+1}^*([\, \mathbf{l}_{bi} \quad -\infty \,], [\, \mathbf{u}_{bi} \quad r_{ui} \,], \mathbf{\Sigma})
$$

$$
+ \sum_{i \in R} w_i \ln \Phi_{B+1}^*([\, \mathbf{l}_{bi} \quad r_{li} \,], [\, \mathbf{u}_{bi} \quad \infty \,], \mathbf{\Sigma})
$$

$$
+ \sum_{i \in I} w_i \ln \Phi_{B+1}^*([\, \mathbf{l}_{bi} \quad r_{li} \,], [\, \mathbf{u}_{bi} \quad r_{ui} \,], \mathbf{\Sigma})
$$

where U is the set of observations where y_i is not censored, L is the set of observations where y_i is left-censored, R is the set of observations where y_i is right-censored, and I is the set of observations where y_i is interval-censored. The vectors $\mathbf{l}_{bi}$ and $\mathbf{u}_{bi}$ are the upper and lower limits for the binary and ordinal endogenous regressors defined in *Binary and ordinal endogenous covariates* in [ERM] **eprobit**. The vectors $\mathbf{l}_i$ and $\mathbf{u}_i$ are the upper and lower limits for the binary and ordinal endogenous regressors defined in *Binary and ordinal endogenous covariates* in [ERM] **eregress**.

The expected value of y_i conditional on $\mathbf{w}_{bi}$ can be calculated using the techniques discussed in *Predictions using the full model* in [ERM] **eprobit postestimation**.

When the endogenous ordinal variables are different treatments, holding the variance and correlation parameters constant over the treatment levels is a constrained form of the potential-outcome model. In an unconstrained potential-outcome model, the variance of the outcome and the correlations between the outcome and the treatments—the endogenous ordinal regressors $\mathbf{w}_{bi}$—vary over the levels of each treatment.

In this unconstrained model, there is a different potential-outcome error for each level of each treatment. For example, when the endogenous treatment variable w_1 has three levels (0, 1, and 2) and the endogenous treatment variable w_2 has four levels (0, 1, 2, and 3), the unconstrained model has $12 = 3 \times 4$ outcome errors. So there are 12 outcome error variance parameters. Because there is a different correlation between each potential outcome and each endogenous treatment, there are 2×12 correlation parameters between the potential outcomes and the treatments in this example model.

We denote the number of different combinations of values for the endogenous treatments $\mathbf{w}_{bi}$ by M, and we denote the vector of values in each combination by $\mathbf{v}_j$ ($j \in \{1, 2, \dots, M\}$). Letting k_{wp} be the number of levels of endogenous ordinal treatment variable $p \in \{1, 2, \dots, B\}$ implies that $M = k_{w1} \times k_{w2} \times \cdots \times k_{wB}$.

Denoting the outcome errors ϵ_{1i}, ..., ϵ_{Mi}, we have

$$
y_{1i} = \mathbf{x}_i \boldsymbol{\beta} + \mathbf{wind}_{b1i} \boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi} \boldsymbol{\beta}_{bB} + \epsilon_{1i}
$$

$$
\vdots
$$

$$
y_{Mi} = \mathbf{x}_i \boldsymbol{\beta} + \mathbf{wind}_{b1i} \boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi} \boldsymbol{\beta}_{bB} + \epsilon_{Mi}
$$

$$
y_i = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j) y_{ji}
$$

For $j = 1$, ..., M, the endogenous errors $\epsilon_{b1i}, \dots, \epsilon_{bBi}$ and outcome error ϵ_{ji} are multivariate normal with 0 mean and covariance

$$
\mathbf{\Sigma}_j = \begin{bmatrix} \mathbf{\Sigma}_b & \boldsymbol{\sigma}_{j1b} \\ \boldsymbol{\sigma}'_{j1b} & \sigma_j^2 \end{bmatrix}
$$

Now let

$$
\sigma_{i,b} = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j)\sigma_j
$$

$$
\boldsymbol{\Sigma}_{i,b} = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j)\boldsymbol{\Sigma}_j
$$

$$
\boldsymbol{\Sigma}_{i,b|1} = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j)\left(\boldsymbol{\Sigma}_b - \frac{\boldsymbol{\sigma}_{j1b}\boldsymbol{\sigma}'_{j1b}}{\sigma_j^2}\right)
$$

Now the log likelihood for this model is

$$
\begin{aligned}
\ln\!L = &\sum_{i\in U} w_i \ln\left\{\Phi_B^*(\mathbf{l}_i, \mathbf{u}_i, \boldsymbol{\Sigma}_{i,b|1})\,\phi\!\left(r_i, \sigma_{i,b}^2\right)\right\} \\
&+ \sum_{i\in L} w_i \ln\Phi_{B+1}^*([\,\mathbf{l}_{bi} \quad -\infty\,], [\,\mathbf{u}_{bi} \quad r_{ui}\,], \boldsymbol{\Sigma}_{i,b}) \\
&+ \sum_{i\in R} w_i \ln\Phi_{B+1}^*([\,\mathbf{l}_{bi} \quad r_{li}\,], [\,\mathbf{u}_{bi} \quad \infty\,], \boldsymbol{\Sigma}_{i,b}) \\
&+ \sum_{i\in I} w_i \ln\Phi_{B+1}^*([\,\mathbf{l}_{bi} \quad r_{li}\,], [\,\mathbf{u}_{bi} \quad r_{ui}\,], \boldsymbol{\Sigma}_{i,b})
\end{aligned}
$$

As in the other case, the expected value of y_i conditional on $\mathbf{w}_{bi}$ can be calculated using the techniques discussed in *Predictions using the full model* in [ERM] **eprobit postestimation**.

Treatment

In the potential-outcomes framework, the treatment t_i is a discrete variable taking T values, indexing the T potential outcomes of the outcome y_i: $y_{1i}, \ldots, y_{Ti}$.

When we observe treatment t_i with levels $v_1, \ldots, v_T$, we have

$$
y_i = \sum_{j=1}^{T} 1(t_i = v_j)y_{ji}
$$

So for each observation, we observe only the potential outcome associated with that observation's treatment value.

For exogenous treatments, our approach is equivalent to the regression adjustment treatment-effect estimation method. See [CAUSAL] **teffects intro advanced**. We do not model the treatment assignment process. The formulas for the treatment effects and potential-outcome means (POMs) are equivalent to what we provide here for endogenous treatments. The treatment effect on the treated for $\mathbf{x}_i$ for an exogenous treatment is equivalent to what we provide here for the endogenous treatment when the correlation parameter between the outcome and treatment errors is set to 0. The average treatment effects (ATEs) and POMs for exogenous treatments are estimated as predictive margins in an analogous manner to what we describe here for endogenous treatments. We can also obtain different variance parameters for the different exogenous treatment groups by specifying povariance in extreat().

From here, we assume an endogenous treatment t_i. As in *Treatment* in [ERM] **eprobit**, we model the treatment assignment process with a probit or an ordered probit model, and we call the treatment assignment error ϵ_{ti}. An interval regression of y_i on exogenous covariates $\mathbf{x}_i$ and endogenous treatment t_i taking values $v_1, \ldots, v_T$ has the form

$$y_{1i} = \mathbf{x}_i \boldsymbol{\beta}_1 + \epsilon_{1i}$$

$$\vdots$$

$$y_{Ti} = \mathbf{x}_i \boldsymbol{\beta}_T + \epsilon_{Ti}$$

$$y_i = \sum_{j=1}^{T} 1(t_i = v_j) y_{ji}$$

As in *Introduction*, we do not observe y_i but instead observe the endpoints y_{li} and y_{ui}.

This model can be formulated with or without different variance and correlation parameters for each potential outcome. Potential-outcome specific parameters are obtained by specifying `povariance` or `pocorrelation` in the `entreat()` option.

If the variance and correlation parameters are not potential-outcome specific, for $j = 1, \ldots, T$, ϵ_{ji} and ϵ_{ti} are bivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma^2 & \sigma\rho_{1t} \\ \sigma\rho_{1t} & 1 \end{bmatrix}$$

The treatment is exogenous if $\rho_{1t} = 0$. Note that we did not specify the structure of the correlations between the potential-outcome errors. We do not need information about these correlations to estimate POMs and treatment effects because all covariates and the outcome are observed in observations from each group.

From here, we discuss a model with an ordinal endogenous treatment. The results for binary treatment models are similar. The likelihood is derived in a similar manner to *Binary and ordinal endogenous covariates*.

For $i \in U$, the uncensored observations, define

$$r_i = y_i - \mathbf{x}_i \boldsymbol{\beta}_j \quad \text{if} \quad t_i = v_j$$

For the censored observations, define

$$r_{li} = y_{li} - \mathbf{x}_i \boldsymbol{\beta}_j \quad \text{if} \quad t_i = v_j$$
$$r_{ui} = y_{ui} - \mathbf{x}_i \boldsymbol{\beta}_j \quad \text{if} \quad t_i = v_j$$

Now the log likelihood is

$$
\begin{aligned}
\ln L = \sum_{i \in U} w_i \ln \left\{ \Phi_1^* \left(l_{ti} - \frac{\rho_{1t}}{\sigma} r_i, u_{ti} - \frac{\rho_{1t}}{\sigma} r_i, 1 - \rho_{1t}^2 \right) \phi \left(r_i, \sigma^2 \right) \right\} \\
+ \sum_{i \in L} w_i \ln \Phi_2^* ([\, l_{ti} \quad -\infty \,], [\, u_{ti} \quad r_{ui} \,], \boldsymbol{\Sigma}) \\
+ \sum_{i \in R} w_i \ln \Phi_2^* ([\, l_{ti} \quad r_{li} \,], [\, u_{ti} \quad \infty \,], \boldsymbol{\Sigma}) \\
+ \sum_{i \in I} w_i \ln \Phi_2^* ([\, l_{ti} \quad r_{li} \,], [\, u_{ti} \quad r_{ui} \,], \boldsymbol{\Sigma})
\end{aligned}
$$

where U is the set of observations where y_i is not censored, L is the set of observations where y_i is left-censored, R is the set of observations where y_i is right-censored, and I is the set of observations where y_i is interval-censored. l_{ti} and u_{ti} are the limits for the treatment probability given in *Treatment* in [ERM] **eprobit**.

The treatment effect $y_{ji} - y_{1i}$ is the difference in the outcome for individual i if the individual receives the treatment $t_i = v_j$ and what the difference would have been if the individual received the control treatment $t_i = v_1$ instead.

The conditional POM for treatment group j is

$$
\text{POM}_j(\mathbf{x}_i) = E\left(y_{ji} | \mathbf{x}_i\right) = \mathbf{x}_i \boldsymbol{\beta}_j
$$

For treatment group j, the treatment effect (TE) conditioned on $\mathbf{x}_i$ is

$$
\text{TE}_j(\mathbf{x}_i) = E\left(y_{ji} - y_{1i} | \mathbf{x}_i\right) = \text{POM}_j(\mathbf{x}_i) - \text{POM}_1(\mathbf{x}_i)
$$

For treatment group j, the treatment effect on the treated (TET) in group h is

$$
\begin{aligned}
\text{TET}_j(\mathbf{x}_i, t_i = v_h) &= E\left(y_{ji} - y_{1i} | \mathbf{x}_i, t_i = v_h\right) \\
&= \mathbf{x}_i \boldsymbol{\beta}_j - \mathbf{x}_i \boldsymbol{\beta}_1 + E\left(\epsilon_{ji} | \mathbf{x}_i, t_i = v_h\right) - E\left(\epsilon_{1i} | \mathbf{x}_i, t_i = v_h\right)
\end{aligned}
$$

Remembering that the outcome errors and the treatment error ϵ_{ti} are multivariate normal, for $j = 1, \ldots, T$, we can decompose ϵ_{ji} such that

$$
\epsilon_{ji} = \sigma \rho_{1t} \epsilon_{ti} + \psi_{ji}
$$

where ψ_{ji} has mean 0.

It follows that

$$
\text{TET}_j(\mathbf{x}_i, t_i = v_h) = \mathbf{x}_i \boldsymbol{\beta}_j - \mathbf{x}_i \boldsymbol{\beta}_1
$$

We can take the expectation of these conditional predictions over the covariates to get population average parameters. The `estat teffects` or `margins` command is used to estimate the expectations as predictive margins once the model is estimated with `eintreg`. The POM for treatment group j is

$$
\text{POM}_j = E\left(y_{ji}\right) = E\left\{\text{POM}_j(\mathbf{x}_i)\right\}
$$

The ATE for treatment group j is

$$\text{ATE}_j = E\left(y_{ji} - y_{1i}\right) = E\left\{\text{TE}_j(\mathbf{x}_i)\right\}$$

For treatment group j, the average treatment effect on the treated (ATET) in treatment group h is

$$\text{ATET}_{jh} = E\left(y_{ji} - y_{1i}|t_i = v_h\right) = E\left\{\text{TET}_j(\mathbf{x}_i, t_i = v_h)|t_i = v_h\right\}$$

The conditional mean of y_i at treatment level v_j is

$$E(y_i|\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_j) = \mathbf{x}_i\boldsymbol{\beta}_j + E(\epsilon_i|\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_j)$$

In *Predictions using the full model* in [ERM] **eprobit postestimation**, we discuss how the conditional mean of ϵ_i is calculated.

If the variance and correlation parameters are potential-outcome specific, for $j = 1, \ldots, T$, ϵ_{ji} and ϵ_{ti} are bivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma}_j = \begin{bmatrix} \sigma_j^2 & \sigma_j\rho_{jt} \\ \sigma_j\rho_{jt} & 1 \end{bmatrix}$$

Define

$$\rho_i = \sum_{j=1}^{T} 1(t_i = v_j)\rho_{jt}$$

$$\sigma_i = \sum_{j=1}^{T} 1(t_i = v_j)\sigma_j$$

$$\boldsymbol{\Sigma}_i = \sum_{j=1}^{T} 1(t_i = v_j)\boldsymbol{\Sigma}_j$$

Now the log likelihood for the model is

$$
\begin{aligned}
\ln L = &\sum_{i \in U} w_i \ln\left\{ \Phi_1^*\left(l_{ti} - \frac{\rho_i}{\sigma_i}r_i, u_{ti} - \frac{\rho_i}{\sigma_i}r_i, 1 - \rho_i^2\right) \phi\left(r_i, \sigma_i^2\right)\right\} \\
&+ \sum_{i \in L} w_i \ln\Phi_2^*([\, l_{ti} \quad -\infty\,], [\, u_{ti} \quad r_{ui}\,], \boldsymbol{\Sigma}_i) \\
&+ \sum_{i \in R} w_i \ln\Phi_2^*([\, l_{ti} \quad r_{li}\,], [\, u_{ti} \quad \infty\,], \boldsymbol{\Sigma}_i) \\
&+ \sum_{i \in I} w_i \ln\Phi_2^*([\, l_{ti} \quad r_{li}\,], [\, u_{ti} \quad r_{ui}\,], \boldsymbol{\Sigma}_i)
\end{aligned}
$$

The definitions for the potential-outcome means and treatment effects are the same as in the case where the variance and correlation parameters did not vary by potential outcome. For the treatment effect on the treated (TET) of group j in group h, we have

$$
\begin{aligned}
\text{TET}_j(\mathbf{x}_i, t_i = v_h) &= E\left(y_{ji} - y_{1i}|\mathbf{x}_i, t_i = v_h\right) \\
&= \mathbf{x}_i\boldsymbol{\beta}_j - \mathbf{x}_i\boldsymbol{\beta}_1 + E\left(\epsilon_{ji}|\mathbf{x}_i, t_i = v_h\right) - E\left(\epsilon_{1i}|\mathbf{x}_i, t_i = v_h\right)
\end{aligned}
$$

The outcome errors and the treatment error ϵ_{ti} are multivariate normal, so for $j = 1, \ldots, T$, we can decompose ϵ_{ji} such that

$$\epsilon_{ji} = \sigma_j \rho_j \epsilon_{ti} + \psi_{ji}$$

where ψ_{ji} has mean 0 and is independent of t_i.

It follows that

$$
\begin{aligned}
\text{TET}_j(\mathbf{x}_i, t_i = v_h) &= E\left(y_{ji} - y_{1i} | \mathbf{x}_i, t_i = v_h\right) \\
&= \mathbf{x}_i \boldsymbol{\beta}_j - \mathbf{x}_i \boldsymbol{\beta}_1 + (\sigma_j \rho_j - \sigma_1 \rho_1) E\left(\epsilon_{ti} | \mathbf{x}_i, t_i = v_h\right)
\end{aligned}
$$

The mean of ϵ_{ti} conditioned on t_i and the exogenous covariates $\mathbf{x}_i$ can be determined using the formulas discussed in *Predictions using the full model* in [ERM] **eprobit postestimation**. It is nonzero. So the treatment effect on the treated will be equal only to the treatment effect under an exogenous treatment or when the correlation and variance parameters are identical between the potential outcomes.

As in the other case, we can take the expectation of these conditional predictions over the covariates to get population-averaged parameters. The estat teffects or margins command is used to estimate the expectations as predictive margins once the model is fit with eintreg.

Endogenous sample selection

Probit endogenous sample selection

The regression for outcome y_i with selection on s_i has the form

$$
\begin{aligned}
y_i &= \mathbf{x}_i \boldsymbol{\beta} + \epsilon_i \\
s_i &= 1\left(\mathbf{z}_{si} \boldsymbol{\alpha}_s + \epsilon_{si} > 0\right)
\end{aligned}
$$

where $\mathbf{x}_i$ are covariates that affect the outcome and $\mathbf{z}_{si}$ are covariates that affect selection. As in the *Introduction* above, we do not observe y_i but instead observe the endpoints y_{li} and y_{ui}. If $s_i = 1$, then the observation is selected, and there is an interval regression contribution to the likelihood. If $s_i = 0$, then the observation is not selected, and there is no interval regression contribution to the likelihood.

The unobserved errors ϵ_i and ϵ_{si} are normal with mean 0 and covariance

$$
\boldsymbol{\Sigma} = \begin{bmatrix} \sigma^2 & \sigma \rho_{1s} \\ \sigma \rho_{1s} & 1 \end{bmatrix}
$$

The likelihood is derived in a similar manner to that in *Treatment.*

For $i \in U$, the uncensored and selected observations, define

$$r_i = y_i - \mathbf{x}_i \boldsymbol{\beta}$$

Let

$$
\begin{aligned}
\mu_{s|1,i} &= \frac{\rho_{1s}}{\sigma} r_i \\
\sigma_{s|1} &= 1 - \rho_{1s}^2
\end{aligned}
$$

For the selection indicator s_i, the lower and upper limits on ϵ_{si} are

$$
l_{si} = \begin{cases} -\infty & s_i = 0 \\ \\ -\mathbf{z}_{si}\boldsymbol{\alpha}_s & s_i = 1 \end{cases}
\qquad
u_{si} = \begin{cases} -\mathbf{z}_{si}\boldsymbol{\alpha}_s & s_i = 0 \\ \\ \infty & s_i = 1 \end{cases}
$$

For the censored but selected observations, $i \notin U$, define

$$
r_{li} = y_{li} - \mathbf{x}_i\boldsymbol{\beta}_j
$$
$$
r_{ui} = y_{ui} - \mathbf{x}_i\boldsymbol{\beta}_j
$$

Now the log likelihood is

$$
\begin{aligned}
\ln L = \sum_{i \in U} w_i \ln \Big\{ \Phi_1^*(l_{si} - \mu_{s|1,i}, u_{si} - \mu_{s|1,i}, \sigma_{s|1}^2)\, \phi(r_i, \sigma^2) \Big\} \\
+ \sum_{i \in L} w_i \ln \Phi_2^*([\,l_{si} \quad -\infty\,], [\,u_{si} \quad r_{ui}\,], \boldsymbol{\Sigma}) \\
+ \sum_{i \in R} w_i \ln \Phi_2^*([\,l_{si} \quad r_{li}\,], [\,u_{si} \quad \infty\,], \boldsymbol{\Sigma}) \\
+ \sum_{i \in I} w_i \ln \Phi_2^*([\,l_{si} \quad r_{li}\,], [\,u_{si} \quad r_{ui}\,], \boldsymbol{\Sigma}) \\
\sum_{i \notin S} w_i \ln \Phi_1^*(l_{si}, u_{si}, 1)
\end{aligned}
$$

where U is the set of observations where y_i is not censored, L is the set of observations where y_i is left-censored, R is the set of observations where y_i is right-censored, I is the set of observations where y_i is interval-censored, and S is the set of selected observations.

The conditional mean of y_i is

$$
E(y_i|\mathbf{x}_i) = \mathbf{x}_i\boldsymbol{\beta}
$$

Tobit endogenous sample selection

Instead of constraining the selection indicator to be binary, tobit endogenous sample selection uses a censored continuous endogenous sample-selection indicator. We allow the selection variable to be left-censored or right-censored.

The underlying regression model for y_i with tobit selection on s_i has the form

$$
y_i = \mathbf{x}_i\boldsymbol{\beta} + \epsilon_i
$$

We observe the selection indicator s_i, which indicates the censoring status of the latent selection variable $s_i^\star$,

$$
s_i^\star = \mathbf{z}_{si}\boldsymbol{\alpha}_s + \epsilon_{si}
$$

$$
s_i = \begin{cases} l_i & s_i^\star \le l_i \\ \\ s_i^\star & l_i < s_i^\star < u_i \\ \\ u_i & s_i^\star \ge u_i \end{cases}
$$

where $\mathbf{z}_{si}$ are covariates that affect selection and l_i and u_i are fixed lower and upper limits.

As in *Introduction*, y_i is observed via the endpoints y_{li} and y_{ui}. If $s_i^\star$ is not censored ($l_i < s_i^\star < u_i$), then the observation is selected, and there is an interval regression contribution to the likelihood. Otherwise, if $s_i^\star$ is left-censored ($s_i^\star < l_i$) or right-censored ($s_i^\star > l_i$), then the observation is not selected, and there is no interval regression contribution to the likelihood. The unobserved errors ϵ_i and ϵ_{si} are normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma^2 & \sigma_{1s} \\ \sigma_{1s} & \sigma_s^2 \end{bmatrix}$$

For the selected observations, we can treat s_i as a continuous endogenous regressor, as in *Continuous endogenous covariates*. In fact, s_i may even be used as a regressor for y_i in `eintreg` (specify `tobitselect(... main)`). On the nonselected observations, we treat s_i like the probit endogenous sample-selection indicator in *Probit endogenous sample selection*.

Conditional on $s_i^\star$ and the exogenous covariates, ϵ_i has mean and variance

$$\mu_{1|s,i} = E\left(\epsilon_i | s_i^\star, \mathbf{x}_i, \mathbf{z}_{si}\right) = \sigma_{1s}\sigma_s^{-2}\left(\mathbf{s}_i^\star - \mathbf{z}_{si}\boldsymbol{\alpha}_s\right)$$
$$\sigma_{1|s}^2 = \text{Var}\left(\epsilon_i | s_i^\star, \mathbf{x}_i, \mathbf{z}_{si}\right) = \sigma^2 - \sigma_{1s}\sigma_s^{-2}\sigma_{1s}$$

Let

$$r_{li} = y_{li} - \mathbf{x}_i\boldsymbol{\beta} - \mu_{1|s,i}$$
$$r_{ui} = y_{ui} - \mathbf{x}_i\boldsymbol{\beta} - \mu_{1|s,i}$$

The log likelihood is

$$\begin{aligned}
\ln L = &\sum_{i \in U} w_i \ln \phi\left(r_{li}, \sigma_{1|s}^2\right) \\
&+ \sum_{i \in L} w_i \ln \Phi_1^*\left(-\infty, r_{ui}, \sigma_{1|s}^2\right) \\
&+ \sum_{i \in R} w_i \ln \Phi_1^*\left(r_{li}, \infty, \sigma_{1|s}^2\right) \\
&+ \sum_{i \in I} w_i \ln \Phi_1^*\left(r_{li}, r_{ui}, \sigma_{1|s}^2\right) \\
&+ \sum_{i \in S} w_i \ln \phi(s_i - \mathbf{z}_{si}\boldsymbol{\alpha}_s, \sigma_s^2) \\
&+ \sum_{i \in L_n} w_i \ln\Phi_1^*(l_{li}, u_{li}, 1) \\
&+ \sum_{i \in R_n} w_i \ln\Phi_1^*(l_{ui}, u_{ui}, 1)
\end{aligned}$$

where S is the set of observations for which y_{li} and y_{ui} are observed, $U \subset S$ is the set of observations where y_i is not censored, $L \subset S$ is the set of observations where y_i is left-censored, $R \subset S$ is the set of observations where y_i is right-censored, $I \subset S$ is the set of observations where y_i is interval-censored, L_n is the set of observations for which $s_i^\star$ is left-censored, and R_n is the set of observations for which $s_i^\star$ is right-censored. The lower and upper limits for selection—l_{li}, u_{li}, l_{ui}, and u_{ui}—are defined in *Tobit endogenous sample selection* in [ERM] **eprobit**.

When s_i is not a covariate in $\mathbf{x}_i$, we use the standard conditional mean formula,

$$E(y_i|\mathbf{x}_i) = \mathbf{x}_i\boldsymbol{\beta}$$

Otherwise, we use

$$E(y_i|\mathbf{x}_i, s_i, z_{si}) = \mathbf{x}_i\boldsymbol{\beta} + \frac{\sigma_{1s}}{\sigma_s^2}(s_i - z_{si}\boldsymbol{\alpha}_s)$$

Random effects

For an interval regression with random effects, we observe panel data. For panel $i = 1, \ldots, N$ and observation $j = 1, \ldots, N_i$, an interval regression of y_{ij} on exogenous covariates $\mathbf{x}_{ij}$ with random effect u_i has the form

$$y_{ij} = \mathbf{x}_{ij}\boldsymbol{\beta} + \epsilon_{ij} + u_i$$

As in *Introduction*, we do not observe y_{ij} but instead observe endpoints y_{lij} and y_{uij}. The random effect u_i is normal with mean 0 and variance σ_u^2. It is independent of the observation-level error ϵ_{ij}, which is normal with mean 0 and variance σ^2.

We derive the likelihood by using the conditional density of y_{lij} and y_{uij} on the random effect u_i and the marginal density of u_i. Multiplying them together we have the joint density, which is integrated over u_i.

Let

$$\begin{aligned}
l_{ij}(u) = &\sum_{j \in U_i} \phi\big(y_{ij} - \mathbf{x}_{ij}\boldsymbol{\beta} - u, \sigma^2\big) \\
&+ \sum_{j \in L_i} \Phi\left(\frac{y_{uij} - \mathbf{x}_{ij}\boldsymbol{\beta} - u}{\sigma}\right) \\
&+ \sum_{i \in R_i} \Phi\left(\frac{-y_{lij} + \mathbf{x}_{ij}\boldsymbol{\beta} - u}{\sigma}\right) \\
&+ \sum_{i \in I_i} \left\{ \Phi\left(\frac{y_{uij} - \mathbf{x}_{ij}\boldsymbol{\beta} - u}{\sigma}\right) - \Phi\left(\frac{y_{lij} - \mathbf{x}_{ij}\boldsymbol{\beta} - u}{\sigma}\right) \right\}
\end{aligned}$$

where U_i is the set of observations where y_{ij} is not censored, L_i is the set of observations where y_{ij} is left-censored, R_i is the set of observations where y_{ij} is right-censored, and I_i is the set of observations where y_{ij} is interval-censored.

The likelihood for panel i is

$$L_i = \int_{-\infty}^{\infty} \phi\left(\frac{u_i}{\sigma_u}\right) \prod_{j=1}^{N_i} l_{ij}(u_i) du_i$$

We can approximate this integral using Gauss–Hermite quadrature. For q-point Gauss–Hermite quadrature, let the abscissa and weight pairs be denoted by (a_{ki}, w_{ki}), $k = 1, \ldots, q$. The Gauss–Hermite quadrature approximation is then

$$\int_{-\infty}^{\infty} f(x) \exp(-x^2)\, dx \approx \sum_{k=1}^{q} w_{ki} f(a_{ki})$$

The default approximation used by xteintreg is mean–variance adaptive Gauss–Hermite quadrature. This chooses optimal abscissa and weights for each panel. See *Likelihood for multiequation models* in [ERM] **eprobit** for more information on the use of mean–variance adaptive Gauss–Hermite quadrature.

Using the quadrature approximation, the log likelihood is

$$\ln L = \sum_{i=1}^{N} \ln \left\{ \sum_{k=1}^{q} w_{ki} \prod_{j=1}^{N_i} l_{ij}(\sigma_u a_{ki}) \right\}$$

The conditional mean of y_{ij} is

$$E(y_{ij}|\mathbf{x}_{ij}) = \mathbf{x}_{ij}\boldsymbol{\beta}$$

Combinations of features

Extended interval regression models that involve multiple features can be formulated using the techniques discussed in *Likelihood for multiequation models* in [ERM] **eprobit**. Essentially, the density of the observed endogenous covariates can be written in terms of the unobserved normal errors. The observed endogenous and exogenous covariates determine the range of the errors, and the joint density can be evaluated as multivariate normal probabilities and densities.

Confidence intervals

The estimated variances will always be nonnegative, and the estimated correlations will always fall in $(-1, 1)$. To obtain confidence intervals that accommodate these ranges, we must use transformations.

We use the log transformation to obtain the confidence intervals for variance parameters and the atanh transformation to obtain confidence intervals for correlation parameters. For details, see *Confidence intervals* in [ERM] **eprobit**.

References

Amemiya, T. 1985. *Advanced Econometrics*. Cambridge, MA: Harvard University Press.

Angrist, J. D. 2001. Estimation of limited dependent variable models with dummy endogenous regressors: Simple strategies for empirical practice. *Journal of Business and Economic Statistics* 19: 2–28. https://doi.org/10.1198/07350010152472571.

Bartus, T., and D. Roodman. 2014. Estimation of multiprocess survival models with cmp. *Stata Journal* 14: 756–777.

Cameron, A. C., and P. K. Trivedi. 2022. *Microeconometrics Using Stata*. 2nd ed. College Station, TX: Stata Press.

Fernandez-Cornejo, J., and S. Wechsler. 2012. Revisiting the impact of Bt corn adoption by U.S. farmers. *Agricultural and Resource Economics Review* 41: 377–390. https://doi.org/10.1017/S1068280500001337.

Heckman, J. J. 1976. The common structure of statistical models of truncation, sample selection and limited dependent variables and a simple estimator for such models. *Annals of Economic and Social Measurement* 5: 475–492.

———. 1979. Sample selection bias as a specification error. *Econometrica* 47: 153–161. https://doi.org/10.2307/1912352.

Maddala, G. S. 1983. *Limited-Dependent and Qualitative Variables in Econometrics*. Cambridge: Cambridge University Press.

Maddala, G. S., and L.-F. Lee. 1976. Recursive models with qualitative endogenous variables. *Annals of Economic and Social Measurement* 5: 525–545.

Mairesse, J., and S. Robin. 2009. Innovation and productivity: A firm-level analysis for French manufacturing and services using CIS3 and CIS4 data (1998–2000 and 2002–2004). CREST–ENSAE.

Newey, W. K. 1987. Efficient estimation of limited dependent variable models with endogenous explanatory variables. *Journal of Econometrics* 36: 231–250. https://doi.org/10.1016/0304-4076(87)90001-7.

Roodman, D. 2011. Fitting fully observed recursive mixed-process models with cmp. *Stata Journal* 11: 159–206.

Tobin, J. 1958. Estimation of relationships for limited dependent variables. *Econometrica* 26: 24–36. https://doi.org/10.2307/1907382.

White, H. L., Jr. 1996. *Estimation, Inference and Specification Analysis*. Cambridge: Cambridge University Press.

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

——. 2014. Quasi-maximum likelihood estimation and testing for nonlinear models with endogenous explanatory variables. *Journal of Econometrics* 182: 226–234. https://doi.org/10.1016/j.jeconom.2014.04.020.

——. 2020. *Introductory Econometrics: A Modern Approach*. 7th ed. Boston: Cengage.

Also see

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

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

[ERM] **predict advanced** — predict's advanced features

[ERM] **predict treatment** — predict for treatment statistics

[ERM] **estat teffects** — Average treatment effects for extended regression models

[ERM] **Intro 9** — Conceptual introduction via worked example

[R] **intreg** — Interval regression

[R] **ivtobit** — Tobit model with continuous endogenous covariates

[R] **tobit** — Tobit regression

[SVY] **svy estimation** — Estimation commands for survey data

[XT] **xtintreg** — Random-effects interval-data regression models

[XT] **xttobit** — Random-effects tobit models

[U] **20 Estimation and postestimation commands**

Title

> **eintreg postestimation —** Postestimation tools for eintreg and xteintreg

Postestimation commands   predict   margins   Remarks and examples
Methods and formulas   Also see

Postestimation commands

The following postestimation command is of special interest after `eintreg` and `xteintreg`:

Command	Description
estat teffects	treatment effects and potential-outcome means

The following standard postestimation commands are also available after `eintreg` and `xteintreg`:

Command	Description
contrast	contrasts and ANOVA-style joint tests of estimates
estat ic	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC)
estat summarize	summary statistics for the estimation sample
estat vce	variance–covariance matrix of the estimators (VCE)
† estat (svy)	postestimation statistics for survey data
estimates	cataloging estimation results
etable	table of estimation results
* forecast	dynamic forecasts and simulations
* hausman	Hausman's specification test
lincom	point estimates, standard errors, testing, and inference for linear combinations of coefficients
* lrtest	likelihood-ratio test
margins	marginal means, predictive margins, marginal effects, and average marginal effects
marginsplot	graph the results from margins (profile plots, interaction plots, etc.)
nlcom	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
predict	means, probabilities, treatment effects, etc.
predictnl	point estimates, standard errors, testing, and inference for generalized predictions
pwcompare	pairwise comparisons of estimates
† suest	seemingly unrelated estimation
test	Wald tests of simple and composite linear hypotheses
testnl	Wald tests of nonlinear hypotheses

*`forecast`, `hausman`, and `lrtest` are not appropriate with `svy` estimation results.

†`suest` and the survey data `estat` commands are not available after `xteintreg`.

93

predict

Predictions after `eintreg` and `xteintreg` are described in

[ERM] **eintreg predict**	predict after eintreg and xteintreg
[ERM] **predict treatment**	predict for treatment statistics
[ERM] **predict advanced**	predict's advanced features

[ERM] **eintreg predict** describes the most commonly used predictions. If you fit a model with treatment effects, predictions specifically related to these models are detailed in [ERM] **predict treatment**. [ERM] **predict advanced** describes less commonly used predictions, such as predictions of outcomes in auxiliary equations.

margins

Description for margins

`margins` estimates statistics based on fitted models. These statistics include marginal means, marginal probabilities, potential-outcome means, average and conditional derivatives, average and conditional effects, and treatment effects.

Menu for margins

Statistics > Postestimation

Syntax for margins

> `margins` [*marginlist*] [, *options*]
>
> `margins` [*marginlist*] , <u>predict</u>(*statistic* ...) [<u>predict</u>(*statistic* ...) ...] [*options*]

statistic	Description
Main	
<u>mean</u>	mean; the default
pr	probability for binary or ordinal y_j
<u>pomean</u>	potential-outcome mean
te	treatment effect
tet	treatment effect on the treated
xb	linear prediction excluding all complications
pr(a,b)	$\Pr(a < y_j < b)$ for continuous y_j
e(a,b)	$E(y_j \mid a < y_j < b)$ for continuous y_j
<u>ystar</u>(a,b)	$E(y_j^*)$, $y_j^* = \max\{a, \min(y_j, b)\}$ for continuous y_j
<u>expmean</u>	calculate $E\{\exp(y_i)\}$

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

For the full syntax, see [R] **margins**.

Remarks and examples

See [ERM] **Intro 7** for an overview of using margins and predict after eintreg and xteintreg. For examples using margins, predict, and estat teffects, see *Interpreting effects* in [ERM] **Intro 9** and see [ERM] **Example 1a**.

Methods and formulas

Counterfactual predictions and inferences for the underlying model in interval regression can be evaluated as in a linear regression model. These predictions and effects are described in *Methods and formulas* of [ERM] **eregress postestimation**. Methods and formulas for all other predictions are given in *Methods and formulas* of [ERM] **eintreg**.

Also see

[ERM] **eintreg** — Extended interval regression

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

[ERM] **predict treatment** — predict for treatment statistics

[ERM] **predict advanced** — predict's advanced features

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

[U] **20 Estimation and postestimation commands**

Title

eintreg predict — predict after eintreg and xteintreg

Description

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

Syntax

You previously fit the model

        eintreg yl yu x1 ..., ...

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

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

where $\mathtt{yl}_i \leq y_i \leq \mathtt{yu}_i$.

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

        xteintreg yl yu x1 ..., ...

Then the main equation would be

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

where $\mathtt{yl}_{ij} \leq y_{ij} \leq \mathtt{yu}_{ij}$.

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, $e_{ij}.\mathtt{y} = u_i.\mathtt{y} + v_{ij}.\mathtt{y}$.

All predictions after `xteintreg` assume the panel-level random effects $(u_i.\mathtt{y})$ 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	
<u>mean</u>	linear prediction; the default
xb	linear prediction excluding all complications
<u>expmean</u>	expected value of exponential of the `mean`; $E\{\exp(\text{mean})\}$
<u>ystar</u>(*a*,*b*)	$E(y*_j), y*_j = \max\{a, \min(y_j, b)\}$

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

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

counterfactual	Description
Main	
<u>target</u>(*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 y 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.

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.

ystar(a, b) is often useful when calculating predictions or using [margins](#) to form inferences because it gives you predictions from the censored distribution rather than treating the censoring as a nuisance and calculating predictions from the uncensored distribution.

To obtain predictions using the censored distribution from your data, you can use the same censoring limits you used when fitting the model. For example, if you typed

```
. eintreg yl yu ...
```

you can then type

```
. predict ystar, ystar(yl, yu)
```

to obtain predictions based on the censored distribution corresponding with your observed data.

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 $\beta_0 + \beta_1 \mathtt{x1}_i + \cdots$. Use of target() is discussed in *Remarks and examples* of [ERM] **eregress predict**.

Remarks and examples

Predictions after fitting models with eintreg and xteintreg are handled the same as they are after fitting models with eregress or xteregress. The issues are the same. See [ERM] **eregress predict**.

Note that censoring is treated as a nuisance in eintreg and xteintreg by default. Default predicted values are from the uncensored distribution, not from the censored distribution.

Methods and formulas

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

Also see

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

[ERM] **eintreg** — Extended interval regression

Title

eoprobit — Extended ordered probit regression

Description

eoprobit fits an ordered probit regression model that accommodates any combination of endogenous covariates, nonrandom treatment assignment, and endogenous sample selection. Continuous, binary, and ordinal endogenous covariates are allowed. Treatment assignment may be endogenous or exogenous. A probit or tobit model may be used to account for endogenous sample selection.

xteoprobit fits a random-effects ordered probit regression model that accommodates endogenous covariates, treatment, and sample selection in the same way as eoprobit and also accounts for correlation of observations within panels or within groups.

Quick start

Ordered probit regression of y on x with continuous endogenous covariate y2 modeled by x and z

    eoprobit y x, endogenous(y2 = x z)

Same as above, but adding continuous endogenous covariate y3 modeled by x and z2

    eoprobit y x, endogenous(y2 = x z) endogenous(y3 = x z2)

Ordered probit regression of y on x with binary endogenous covariate d modeled by x and z

    eoprobit y x, endogenous(d = x z, probit)

Ordered probit regression of y on x with endogenous treatment recorded in trtvar and modeled by x and z

    eoprobit y x, entreat(trtvar = x z)

Ordered probit regression of y on x with exogenous treatment recorded in trtvar

    eoprobit y x, extreat(trtvar)

Random-effects ordered probit regression of y on x using xtset data

    xteoprobit y x

Ordered probit regression of y on x with endogenous sample-selection indicator selvar modeled by x and z

    eoprobit y x, select(selvar = x z)

Same as above, but adding endogenous covariate y2 modeled by x and z2

    eoprobit y x, select(selvar = x z) endogenous(y2 = x z2)

Same as above, but adding endogenous treatment recorded in trtvar and modeled by x and z3

    eoprobit y x, select(selvar = x z) endogenous(y2 = x z2)   ///
        entreat(trtvar = x z3)

Same as above, but with random effects and without endogenous treatment

```
xteoprobit y x, select(selvar = x z) endogenous(y2 = x z2)
```

Menu

eoprobit

Statistics > Endogenous covariates > Models adding selection and treatment > Ordered probit regression

xteoprobit

Statistics > Longitudinal/panel data > Endogenous covariates > Models adding selection and treatment > Ordered probit regression (RE)

Syntax

Basic ordered probit regression with endogenous covariates

   eoprobit *depvar* [*indepvars*] , <u>endog</u>enous(*depvars*_{en} = *varlist*_{en}) [*options*]

Basic ordered probit regression with endogenous treatment assignment

   eoprobit *depvar* [*indepvars*] , <u>entr</u>eat(*depvar*_{tr} [= *varlist*_{tr}]) [*options*]

Basic ordered probit regression with exogenous treatment assignment

   eoprobit *depvar* [*indepvars*] , <u>extr</u>eat(*tvar*) [*options*]

Basic ordered probit regression with sample selection

   eoprobit *depvar* [*indepvars*] , <u>sel</u>ect(*depvar*_s = *varlist*_s) [*options*]

Basic ordered probit regression with tobit sample selection

   eoprobit *depvar* [*indepvars*] , <u>tobitsel</u>ect(*depvar*_s = *varlist*_s) [*options*]

Basic ordered probit regression with random effects

   xteoprobit *depvar* [*indepvars*] [, *options*]

Ordered probit regression combining endogenous covariates, treatment, and selection

   eoprobit *depvar* [*indepvars*] [*if*] [*in*] [*weight*] [, *extensions options*]

Ordered probit regression combining random effects, endogenous covariates, treatment, and selection

   xteoprobit *depvar* [*indepvars*] [*if*] [*in*] [, *extensions options*]

extensions	Description
Model	
endogenous(*enspec*)	model for endogenous covariates; may be repeated
entreat(*entrspec*)	model for endogenous treatment assignment
extreat(*extrspec*)	exogenous treatment
select(*selspec*)	probit model for selection
tobitselect(*tselspec*)	tobit model for selection

options	Description
Model	
offset(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1
constraints(*numlist*)	apply specified linear constraints
SE/Robust	
vce(*vcetype*)	*vcetype* may be oim, robust, cluster *clustvar*, opg, bootstrap, or jackknife
Reporting	
level(*#*)	set confidence level; default is level(95)
nocnsreport	do not display constraints
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Integration	
intpoints(*#*)	set the number of integration (quadrature) points for integration over four or more dimensions; default is intpoints(128)
triintpoints(*#*)	set the number of integration (quadrature) points for integration over three dimensions; default is triintpoints(10)
reintpoints(*#*)	set the number of integration (quadrature) points for random-effects integration; default is reintpoints(7)
reintmethod(*intmethod*)	integration method for random effects; *intmethod* may be mvaghermite (the default) or ghermite
Maximization	
maximize_options	control the maximization process; seldom used
collinear	keep collinear variables
coeflegend	display legend instead of statistics

enspec is *depvars*$_{en}$ = *varlist*$_{en}$ [, *enopts*]

  where *depvars*$_{en}$ is a list of endogenous covariates. Each variable in *depvars*$_{en}$ specifies an endogenous covariate model using the common *varlist*$_{en}$ and options.

entrspec is *depvar*$_{tr}$ [= *varlist*$_{tr}$] [, *entropts*]

  where *depvar*$_{tr}$ is a variable indicating treatment assignment. *varlist*$_{tr}$ is a list of covariates predicting treatment assignment.

extrspec is *tvar* $\big[$, *extropts* $\big]$

   where *tvar* is a variable indicating treatment assignment.

selspec is *depvar$_s$* = *varlist$_s$* $\big[$, *selopts* $\big]$

   where *depvar$_s$* is a variable indicating selection status. *depvar$_s$* must be coded as 0, indicating
   that the observation was not selected, or 1, indicating that the observation was selected. *varlist$_s$*
   is a list of covariates predicting selection.

tselspec is *depvar$_s$* = *varlist$_s$* $\big[$, *tselopts* $\big]$

   where *depvar$_s$* is a continuous variable. *varlist$_s$* is a list of covariates predicting *depvar$_s$*. The
   censoring status of *depvar$_s$* indicates selection, where a censored *depvar$_s$* indicates that the
   observation was not selected and a noncensored *depvar$_s$* indicates that the observation was
   selected.

enopts	Description
Model	
probit	treat endogenous covariate as binary
oprobit	treat endogenous covariate as ordinal
pocorrelation	estimate different correlations for each level of a binary or an ordinal endogenous covariate
nomain	do not add endogenous covariate to main equation
nore	do not include random effects in model for endogenous covariate
noconstant	suppress constant term

nore is available only with xteoprobit.

entropts	Description
Model	
pocorrelation	estimate different correlations for each potential outcome
nomain	do not add treatment indicator to main equation
nocutsinteract	do not interact treatment with cutpoints
nointeract	do not interact treatment with covariates in main equation
nore	do not include random effects in model for endogenous treatment
noconstant	suppress constant term
offset(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1

nore is available only with xteoprobit.

extropts	Description
Model	
pocorrelation	estimate different correlations for each potential outcome
nomain	do not add treatment indicator to main equation
nocutsinteract	do not interact treatment with cutpoints
nointeract	do not interact treatment with covariates in main equation

selopts	Description
Model	
<u>nore</u>	do not include random effects in selection model
<u>noconstant</u>	suppress constant term
<u>off</u>set(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1

nore is available only with xteoprobit.

tselopts	Description
Model	
*ll(*varname* | #)	left-censoring variable or limit
*ul(*varname* | #)	right-censoring variable or limit
main	add censored selection variable to main equation
<u>nore</u>	do not include random effects in tobit selection model
<u>noconstant</u>	suppress constant term
<u>off</u>set(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1

* You must specify either ll() or ul().

nore is available only with xteoprobit.

indepvars, *varlist*$_{en}$, *varlist*$_{tr}$, and *varlist*$_s$ may contain factor variables; see [U] **11.4.3 Factor variables**.

depvar, *indepvars*, *depvars*$_{en}$, *varlist*$_{en}$, *depvar*$_{tr}$, *varlist*$_{tr}$, *tvar*, *depvar*$_s$, and *varlist*$_s$ may contain time-series operators; see [U] **11.4.4 Time-series varlists**.

bootstrap, by, collect, jackknife, and statsby are allowed with eoprobit and xteoprobit. rolling and svy are allowed with eoprobit. See [U] **11.1.10 Prefix commands**.

Weights are not allowed with the bootstrap prefix; see [R] **bootstrap**.

vce() and weights are not allowed with the svy prefix; see [SVY] **svy**.

fweights, iweights, and pweights are allowed with eoprobit; see [U] **11.1.6 weight**.

reintpoints() and reintmethod() are available only with xteoprobit.

collinear and coeflegend do not appear in the dialog box.

See [U] **20 Estimation and postestimation commands** for more capabilities of estimation commands.

Options

‎⎴ Model ⎴‎

endogenous(*enspec*), entreat(*entrspec*), extreat(*extrspec*), select(*selspec*),
   tobitselect(*tselspec*), re; see [ERM] **ERM options**.

offset(*varname$_o$*), constraints(*numlist*); see [R] **Estimation options**.

‎⎴ SE/Robust ⎴‎

vce(*vcetype*); see [ERM] **ERM options**.

‎⎴ Reporting ⎴‎

level(#), nocnsreport; see [R] **Estimation options**.

display_options: noci, <u>nopv</u>alues, <u>noomitted</u>, vsquish, noemptycells, <u>base</u>levels,
   <u>allbase</u>levels, <u>nofvlabel</u>, fvwrap(#), fvwrapon(*style*), cformat(*%fmt*), pformat(*%fmt*),
   sformat(*%fmt*), and nolstretch; see [R] **Estimation options**.

Integration
---

intpoints(*#*), triintpoints(*#*), reintpoints(*#*), reintmethod(*intmethod*); see [ERM] **ERM options**.

Maximization
---

maximize_options: <u>d</u>ifficult, <u>tech</u>nique(*algorithm_spec*), <u>iter</u>ate(*#*), [no]log, <u>trace</u>,
   gradient, showstep, <u>hess</u>ian, <u>showtol</u>erance, <u>tol</u>erance(*#*), <u>ltol</u>erance(*#*),
   <u>nrtol</u>erance(*#*), <u>nonrtol</u>erance, and from(*init_specs*); see [R] **Maximize**.

   The default technique for eoprobit is technique(nr). The default technique for xteoprobit
   is technique(bhhh 10 nr 2).

   Setting the optimization type to technique(bhhh) resets the default *vcetype* to vce(opg).

The following options are available with eoprobit and xteoprobit but are not shown in the dialog
box:

collinear, coeflegend; see [R] **Estimation options**.

Remarks and examples

   eoprobit and xteoprobit fit models that we refer to as "extended ordered probit regression
models", meaning that they accommodate endogenous covariates, nonrandom treatment assignment,
endogenous sample selection, and panel data or other grouped data.

   eoprobit fits models for cross-sectional data (one-level models). eoprobit can account for
endogenous covariates, treatment, and sample selection, whether these complications arise individually
or in combination.

   xteoprobit fits random-effects models (two-level models) for panel data or grouped data.
xteoprobit accounts for endogenous covariates, treatment, and sample selection in the same way
as eoprobit and also accounts for within-panel or within-group correlation among observations.

   In this entry, you will find information on the syntax for the eoprobit and xteoprobit commands.
You can see *Methods and formulas* for a full description of the models that can be fit with eoprobit
and xteoprobit and details about how those models are fit.

   More information on extended ordered probit models is found in the separate introductions and
example entries. We recommend reading those entries to learn how to use eoprobit and xteoprobit.
Below, we provide a guide to help you locate the ones that will be helpful to you.

   For an introduction to eoprobit and xteoprobit and the other extended regression commands
for continuous, interval, and binary outcomes, see [ERM] **Intro 1**–[ERM] **Intro 9**.

   [ERM] **Intro 1** introduces the ERM commands, the problems they address, and their syntax.

   [ERM] **Intro 2** provides background on the four types of models—linear regression, interval
regression, probit regression, and ordered probit regression—that can be fit using ERM commands.

   [ERM] **Intro 3** considers the problem of endogenous covariates and how to solve it using ERM
commands.

   [ERM] **Intro 4** gives an overview of endogenous sample selection and using ERM commands to
account for it when fitting a linear, interval, probit, or ordered probit model.

   [ERM] **Intro 5** covers nonrandom treatment assignment and how to account for it using eoprobit
or any of the other ERM commands.

[ERM] **Intro 6** covers random-effects models for panel data and other grouped data. It discusses xteoprobit and the other ERM commands for panel data.

[ERM] **Intro 7** discusses interpretation of results. You can interpret coefficients from eoprobit and xteoprobit in the usual way, but this introduction goes beyond the interpretation of coefficients. We demonstrate how to find answers to interesting questions by using margins. If your model includes an endogenous covariate or an endogenous treatment, the use of margins differs from its use after other estimation commands, so we strongly recommend reading this intro if you are fitting these types of models.

[ERM] **Intro 8** will be particularly helpful if you are familiar with heckoprobit or xtoprobit and other commands that address endogenous covariates, sample selection, nonrandom treatment assignment, or random effects. This introduction is a Rosetta stone that maps the syntax of those commands to the syntax of eoprobit and xteoprobit.

[ERM] **Intro 9** walks you through an example that gives insight into the concepts of endogenous covariates, treatment assignment, and sample selection while fitting models with eregress that address these complications. Although the example uses eregress, the discussion applies equally to eoprobit. This intro also demonstrates how to interpret results by using margins and estat teffects.

Additional examples are presented in [ERM] **Example 1a**–[ERM] **Example 9**. For examples using eoprobit, see

| [ERM] **Example 6a** | Ordered probit regression with endogenous treatment |
| [ERM] **Example 6b** | Ordered probit regression with endogenous treatment and sample selection |

For an example using xteoprobit, see

| [ERM] **Example 9** | Ordered probit regression with endogenous treatment and random effects |

See *Examples* in [ERM] **Intro** for an overview of all the examples. All examples may be interesting because they handle complications in the same way.

eoprobit and xteoprobit fit many models discussed in the literature. For instance, eoprobit can be used to fit models like the ordered probit model with endogenous sample selection discussed in De Luca and Perotti (2011) and the ordered probit models with continuous or binary endogenous covariates discussed in Wooldridge (2010, sec. 16.3.3). Roodman (2011) investigated ordered probit models with endogenous covariates and endogenous sample selection and demonstrated how multiple observational data complications could be addressed with a triangular model structure. He and Tamás Bartus showed how random effects could be used in the triangular model structure in Bartus and Roodman (2014). Roodman's work has been used to model processes like the effect of living with a child on the happiness of the elderly (Chyi and Mao 2012) and the effect of parental migration on child education (Botezat and Pfeiffer 2014).

Stored results

eoprobit stores the following in e():

Scalars
e(N)	number of observations
e(N_selected)	number of selected observations
e(N_nonselected)	number of nonselected observations
e(k)	number of parameters
e(k_cat#)	number of categories for the #th *depvar*, ordinal
e(k_eq)	number of equations in e(b)

e(k_eq_model)	number of equations in overall model test
e(k_dv)	number of dependent variables
e(k_aux)	number of auxiliary parameters
e(df_m)	model degrees of freedom
e(ll)	log likelihood
e(N_clust)	number of clusters
e(chi2)	χ^2
e(p)	p-value for model test
e(n_quad)	number of integration points for multivariate normal
e(n_quad3)	number of integration points for trivariate normal
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise

Macros
e(cmd)	eoprobit
e(cmdline)	command as typed
e(depvar)	names of dependent variables
e(tsel_ll)	left-censoring limit for tobit selection
e(tsel_ul)	right-censoring limit for tobit selection
e(wtype)	weight type
e(wexp)	weight expression
e(title)	title in estimation output
e(clustvar)	name of cluster variable
e(offset#)	offset for the #th *depvar*, where # is determined by equation order in output
e(chi2type)	Wald; type of model χ^2 test
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. err.
e(opt)	type of optimization
e(which)	max or min; whether optimizer is to perform maximization or minimization
e(ml_method)	type of ml method
e(user)	name of likelihood-evaluator program
e(technique)	maximization technique
e(properties)	b V
e(estat_cmd)	program used to implement estat
e(predict)	program used to implement predict
e(marginsok)	predictions allowed by margins
e(marginsnotok)	predictions disallowed by margins
e(marginsdefault)	default predict() specification for margins
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices
e(b)	coefficient vector
e(cat#)	categories for the #th *depvar*, ordinal
e(Cns)	constraints matrix
e(ilog)	iteration log (up to 20 iterations)
e(gradient)	gradient vector
e(V)	variance–covariance matrix of the estimators
e(V_modelbased)	model-based variance

Functions
e(sample)	marks estimation sample

In addition to the above, the following is stored in r():

Matrices
r(table)	matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

`xteoprobit` stores the following in `e()`:

Scalars
e(N)	number of observations
e(N_g)	number of groups
e(N_selected)	number of selected observations
e(N_nonselected)	number of nonselected observations
e(k)	number of parameters
e(k_cat#)	number of categories for the #th *depvar*, ordinal
e(k_eq)	number of equations in e(b)
e(k_eq_model)	number of equations in overall model test
e(k_dv)	number of dependent variables
e(k_aux)	number of auxiliary parameters
e(df_m)	model degrees of freedom
e(ll)	log likelihood
e(N_clust)	number of clusters
e(chi2)	χ^2
e(p)	*p*-value for model test
e(n_quad)	number of integration points for multivariate normal
e(n_quad3)	number of integration points for trivariate normal
e(n_requad)	number of integration points for random effects
e(g_min)	smallest group size
e(g_avg)	average group size
e(g_max)	largest group size
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise

Macros
e(cmd)	xteoprobit
e(cmdline)	command as typed
e(depvar)	names of dependent variables
e(tsel_ll)	left-censoring limit for tobit selection
e(tsel_ul)	right-censoring limit for tobit selection
e(ivar)	variable denoting groups
e(title)	title in estimation output
e(clustvar)	name of cluster variable
e(offset#)	offset for the #th *depvar*, where # is determined by equation order in output
e(chi2type)	Wald; type of model χ^2 test
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. err.
e(reintmethod)	integration method for random effects
e(opt)	type of optimization
e(which)	max or min; whether optimizer is to perform maximization or minimization
e(ml_method)	type of ml method
e(user)	name of likelihood-evaluator program
e(technique)	maximization technique
e(properties)	b V
e(estat_cmd)	program used to implement estat
e(predict)	program used to implement predict
e(marginsok)	predictions allowed by margins
e(marginsnotok)	predictions disallowed by margins
e(marginsdefault)	default predict() specification for margins
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices
e(b)	coefficient vector
e(cat#)	categories for the #th *depvar*, ordinal
e(Cns)	constraints matrix
e(ilog)	iteration log (up to 20 iterations)
e(gradient)	gradient vector

| e(V) | variance–covariance matrix of the estimators |
| e(V_modelbased) | model-based variance |

Functions
| e(sample) | marks estimation sample |

In addition to the above, the following is stored in `r()`:

Matrices
| r(table) | matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals |

Note that results stored in `r()` are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Methods and formulas

The methods and formulas presented here are for the ordered probit model. The estimators implemented in `eoprobit` and `xteoprobit` are maximum likelihood estimators covered by the results in chapter 13 of Wooldridge (2010) and White (1996).

The log-likelihood functions maximized by `eoprobit` and `xteoprobit` are implied by the triangular structure of the model. Specifically, the joint distribution of the endogenous variables is a product of conditional and marginal distributions because the model is triangular. For a few of the many relevant applications of this result in literature, see chapter 10 of Amemiya (1985); Heckman (1976, 1979); chapter 5 of Maddala (1983); Maddala and Lee (1976); sections 15.7.2, 15.7.3, 16.3.3, 17.5.2, and 19.7.1 in Wooldridge (2010); and Wooldridge (2014). Roodman (2011) and Bartus and Roodman (2014) used this result to derive the formulas discussed below.

Methods and formulas are presented under the following headings:

Introduction
Endogenous covariates
    Continuous endogenous covariates
    Binary and ordinal endogenous covariates
Treatment
Endogenous sample selection
    Probit endogenous sample selection
    Tobit endogenous sample selection
Random effects
Combinations of features
Confidence intervals

Introduction

An ordered probit regression of outcome y_i on covariates $\mathbf{x}_i$ may be written as

$$y_i = v_h \quad \text{iff} \quad \kappa_{h-1} < \mathbf{x}_i\boldsymbol{\beta} + \epsilon_i \leq \kappa_h$$

The values $v_1, \ldots, v_H$ are real numbers such that $v_h < v_m$ for $h < m$. κ_0 is taken as $-\infty$ and κ_H is taken as $+\infty$. The unobserved error ϵ_i is standard normal.

The log likelihood for this model is

$$
\ln L = \sum_{i=1}^{N} w_i \ln \left[\begin{array}{l} 1(y_i = v_1)\Phi\left(-\mathbf{x}_i\boldsymbol{\beta}\right) \\ + \displaystyle\sum_{h=2}^{H-1} 1(y_i = v_h)\left\{\Phi\left(\kappa_h - \mathbf{x}_i\boldsymbol{\beta}\right) - \Phi\left(\kappa_{h-1} - \mathbf{x}_i\boldsymbol{\beta}\right)\right\} \\ + 1(y_i = v_H)\Phi\left(\mathbf{x}_i\boldsymbol{\beta}\right) \end{array} \right]
$$

where w_i are the weights.

For $h = 0, \ldots, H$, define

$$
c_{ih} = \begin{cases} -\infty & h = 0 \\ \kappa_h - \mathbf{x}_i\boldsymbol{\beta} & h = 1, \ldots, H-1 \\ \infty & h = H \end{cases} \tag{1}
$$

This leads to the limits

$$
l_{1i} = c_{i(h-1)} \quad \text{if} \quad y_i = v_h \tag{2}
$$

and

$$
u_{1i} = c_{ih} \quad \text{if} \quad y_i = v_h \tag{3}
$$

These are limits on the unobserved ϵ_i based on the observed values of y_i and $\mathbf{x}_i$. They let us rewrite the log likelihood concisely as

$$
\ln L = \sum_{i=1}^{N} w_i \ln\Phi_1^*(l_{1i}, u_{1i}, 1)
$$

The conditional probabilities of success can be written using similar notation. For $h = 1, \ldots, H$,

$$
\Pr\left(y_i = v_h | \mathbf{x}_i\right) = \Phi_1^*(c_{i(h-1)}, c_{ih}, 1) \tag{4}
$$

If you are willing to take our word for some derivations and notation, the following is complete. Longer explanations and derivations for some terms and functions are provided in *Methods and formulas* of [ERM] **eprobit**. For example, we need the two-sided probability function Φ_d^* that is discussed in *Introduction* in [ERM] **eprobit**.

If you are interested in all the details, we suggest you read *Methods and formulas* of [ERM] **eprobit** in its entirety before reading this section. Here we mainly show how the complications that arise in ERMs are handled in an ordered probit framework.

Endogenous covariates

Continuous endogenous covariates

An ordered probit regression of y_i on exogenous covariates $\mathbf{x}_i$ and C continuous endogenous covariates $\mathbf{w}_{ci}$ has the form

$$y_i = v_h \quad \text{iff} \quad \kappa_{h-1} < \mathbf{x}_i\boldsymbol{\beta} + \mathbf{w}_{ci}\boldsymbol{\beta}_c + \epsilon_i \leq \kappa_h$$

$$\mathbf{w}_{ci} = \mathbf{z}_{ci}\mathbf{A}_c + \boldsymbol{\epsilon}_{ci}$$

The values $v_1, \ldots, v_H$ are real numbers such that $v_h < v_m$ for $h < m$. κ_0 is taken as $-\infty$ and κ_H is taken as $+\infty$. The vector $\mathbf{z}_{ci}$ contains variables from $\mathbf{x}_i$ and other covariates that affect $\mathbf{w}_{ci}$. The unobserved errors ϵ_i and $\boldsymbol{\epsilon}_{ci}$ are multivariate normal with mean 0 and covariance

$$\begin{bmatrix} 1 & \boldsymbol{\sigma}'_{1c} \\ \boldsymbol{\sigma}_{1c} & \boldsymbol{\Sigma}_c \end{bmatrix}$$

As in *Continuous endogenous covariates* in [ERM] **eprobit**, the likelihood can be written using the conditional density of ϵ_i on $\mathbf{w}_{ci}$.

Now, for $h = 0, \ldots, H$, define

$$c_{ih} = \begin{cases} -\infty & h = 0 \\ \kappa_h - \mathbf{x}_i\boldsymbol{\beta} - \boldsymbol{\sigma}'_{1c}\boldsymbol{\Sigma}_c^{-1}\left(\mathbf{w}_{ci} - \mathbf{z}_{ci}\mathbf{A}_c\right)' & h = 1, \ldots, H-1 \\ \infty & h = H \end{cases}$$

These expressions used the conditional mean of ϵ_i. The lower and upper limits for the y_i probability are

$$l_{1i} = c_{i(h-1)} \quad \text{if} \quad y_i = v_h$$

and

$$u_{1i} = c_{ih} \quad \text{if} \quad y_i = v_h$$

Using these limits, the conditional variance, and the conditional density of $\mathbf{w}_{ci}$, we obtain the log likelihood

$$\ln L = \sum_{i=1}^{N} w_i \left\{ \ln \Phi_1^*\left(l_{1i}, u_{1i}, 1 - \boldsymbol{\sigma}'_{1c}\boldsymbol{\Sigma}_c^{-1}\boldsymbol{\sigma}_{1c}\right) + \ln \phi_C(\mathbf{w}_{ci} - \mathbf{z}_{ci}\mathbf{A}_c, \boldsymbol{\Sigma}_c) \right\}$$

The conditional probabilities of success can be written using similar notation. For $h = 1, \ldots, H$,

$$\Pr\left(y_i = v_h | \mathbf{x}_i\right) = \Phi_1^*(c_{i(h-1)}, c_{ih}, 1 - \boldsymbol{\sigma}'_{1c}\boldsymbol{\Sigma}_c^{-1}\boldsymbol{\sigma}_{1c})$$

Binary and ordinal endogenous covariates

Here we begin by formulating the ordered probit regression of y_i on exogenous covariates $\mathbf{x}_i$ and B binary and ordinal endogenous covariates $\mathbf{w}_{bi} = [w_{b1i}, \ldots, w_{bBi}]$. Indicator (dummy) variables for the levels of each binary and ordinal covariate are used in the model. You can also interact other covariates with the binary and ordinal endogenous covariates, as in treatment-effect models.

The binary and ordinal endogenous covariates $\mathbf{w}_{bi}$ are formulated as in *Binary and ordinal endogenous covariates* in [ERM] **eprobit**.

The model for the outcome can be formulated with or without different correlation parameters for each level of $\mathbf{w}_{bi}$. Level-specific parameters are obtained by specifying `pocorrelation` in the `endogenous()` option.

If the correlation parameters are not level specific, we have

$$y_i = v_h \quad \text{iff} \quad \kappa_{h-1} < \mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB} + \epsilon_i \leq \kappa_h$$

where the values $v_1, \ldots, v_H$ are real numbers such that $v_h < v_m$ for $h < m$. κ_0 is taken as $-\infty$ and κ_H is taken as $+\infty$. The $\mathbf{wind}_{bji}$ vectors are defined in *Binary and ordinal endogenous covariates* in [ERM] **eprobit**. The outcome error ϵ_i and binary and ordinal endogenous errors $\epsilon_{b1i}, \ldots, \epsilon_{bBi}$ are multivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & \boldsymbol{\rho}'_{1b} \\ \boldsymbol{\rho}_{1b} & \boldsymbol{\Sigma}_b \end{bmatrix}$$

From here, we discuss the model with ordinal endogenous covariates. The results for binary endogenous covariates are similar.

Now, for $h = 0, \ldots, H$, define

$$c_{ih} = \begin{cases} -\infty & h = 0 \\ \kappa_h - \mathbf{x}_i\boldsymbol{\beta} - \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} - \cdots - \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB} & h = 1, \ldots, H-1 \\ \infty & h = H \end{cases}$$

The lower and upper limits for the y_i probability are

$$l_{1i} = c_{i(h-1)} \quad \text{if} \quad y_i = v_h$$

and

$$u_{1i} = c_{ih} \quad \text{if} \quad y_i = v_h$$

Let

$$\mathbf{l}_i = \begin{bmatrix} l_{1i} & l_{b1i} & \ldots & l_{bBi} \end{bmatrix}$$
$$\mathbf{u}_i = \begin{bmatrix} u_{1i} & u_{b1i} & \ldots & u_{bBi} \end{bmatrix}$$

where the l_{bji} and u_{bji} are the lower and upper limits for the binary and ordinal endogenous covariate probabilities. They are defined in *Binary and ordinal endogenous covariates* in [ERM] **eprobit**.

So the log likelihood for this model is

$$\ln L = \sum_{i=1}^{N} w_i \ln \Phi^*_{B+1}(\mathbf{l}_i, \mathbf{u}_i, \boldsymbol{\Sigma})$$

Now let

$$\mathbf{l}_{bi} = \begin{bmatrix} l_{b1i} & \dots & l_{bBi} \end{bmatrix}$$

$$\mathbf{u}_{bi} = \begin{bmatrix} u_{b1i} & \dots & u_{bBi} \end{bmatrix}$$

$$\mathbf{l}_{ih1} = \begin{bmatrix} c_{i(h-1)} & \mathbf{l}_{bi} \end{bmatrix}$$

$$\mathbf{u}_{ih1} = \begin{bmatrix} c_{ih} & \mathbf{u}_{bi} \end{bmatrix}$$

The conditional probabilities are

$$\Pr\left(y_i = v_h | \mathbf{x}_i, \mathbf{z}_{b1i}, \dots, \mathbf{z}_{bBi}, \mathbf{w}_{bi}\right) = \frac{\Phi^*_{B+1}(\mathbf{l}_{ih1}, \mathbf{u}_{ih1}, \mathbf{\Sigma})}{\Phi^*_B(\mathbf{l}_{bi}, \mathbf{u}_{bi}, \mathbf{\Sigma}_b)}$$

When the endogenous ordinal variables are different treatments, holding the correlation parameters constant over the treatment levels is a constrained form of the potential-outcome model. In an unconstrained potential-outcome model, the correlations between the outcome and the treatments—the endogenous ordinal regressors $\mathbf{w}_{bi}$—vary over the levels of each treatment.

In this unconstrained model, there is a different potential-outcome error for each level of each treatment. For example, when the endogenous treatment variable w_1 has three levels (0, 1, and 2) and the endogenous treatment variable w_2 has four levels (0, 1, 2, and 3), the unconstrained model has $12 = 3 \times 4$ outcome errors. Because there is a different correlation between each potential outcome and each endogenous treatment, there are 2×12 correlation parameters between the potential outcomes and the treatments in this example model.

We denote the number of different combinations of values for the endogenous treatments $\mathbf{w}_{bi}$ by M, and we denote the vector of values in each combination by $\mathbf{v}_j$ ($j \in \{1, 2, \dots, M\}$). Letting k_{wp} be the number of levels of endogenous ordinal treatment variable $p \in \{1, 2, \dots, B\}$ implies that $M = k_{w1} \times k_{w2} \times \cdots \times k_{wB}$.

In this case, we have

$$y_i = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j) y_{ji}$$

where for $j = 1, \dots, M$,

$$y_{ji} = v_h \quad \text{iff} \quad \kappa_{h-1} < \mathbf{x}_i \boldsymbol{\beta} + \mathbf{wind}_{b1i} \boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi} \boldsymbol{\beta}_{bB} + \epsilon_{ji} \leq \kappa_h$$

The outcome errors ϵ_{ji} and the endogenous errors $\epsilon_{b1i}, \dots, \epsilon_{bBi}$ are multivariate normal with 0 mean and covariance

$$\mathbf{\Sigma}_j = \begin{bmatrix} 1 & \boldsymbol{\rho}'_{j1b} \\ \boldsymbol{\rho}_{j1b} & \mathbf{\Sigma}_b \end{bmatrix}$$

Now let

$$\mathbf{\Sigma}_{i,b} = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j) \mathbf{\Sigma}_j$$

Now the log likelihood for this model is

$$\ln L = \sum_{i=1}^{N} w_i \ln \Phi^*_{B+1}(\mathbf{l}_i, \mathbf{u}_i, \mathbf{\Sigma}_{i,b})$$

The conditional probabilities are

$$\Pr\left(y_i = v_h | \mathbf{x}_i, \mathbf{z}_{b1i}, \dots, \mathbf{z}_{bBi}, \mathbf{w}_{bi}\right) = \frac{\Phi^*_{B+1}(\mathbf{l}_{ih1}, \mathbf{u}_{ih1}, \mathbf{\Sigma}_{i,b})}{\Phi^*_B(\mathbf{l}_{bi}, \mathbf{u}_{bi}, \mathbf{\Sigma}_b)}$$

Treatment

In the potential-outcomes framework, the treatment t_i is a discrete variable taking T values, indexing the T potential outcomes of the outcome y_i: $y_{1i}, \ldots, y_{Ti}$.

When we observe treatment t_i with levels $v_1, \ldots, v_T$, we have

$$y_i = \sum_{j=1}^{T} 1(t_i = v_{tj}) y_{ji}$$

So for each observation, we observe only the potential outcome associated with that observation's treatment value.

For exogenous treatments, our approach is equivalent to the regression adjustment treatment-effect estimation method. See [CAUSAL] **teffects intro advanced**. We do not model the treatment assignment process. The formulas for the treatment effects and potential-outcome means (POMs) are equivalent to what we provide here for endogenous treatments. The treatment effect on the treated for $\mathbf{x}_i$ for an exogenous treatment is equivalent to what we provide here for the endogenous treatment when the correlation parameter between the outcome and treatment errors is set to 0. The average treatment effects (ATEs) and POMs for exogenous treatments are estimated as predictive margins in an analogous manner to what we describe here for endogenous treatments.

From here, we assume an endogenous treatment t_i. As in *Treatment* in [ERM] **eprobit**, we model the treatment assignment process with a probit or an ordered probit model, and we call the treatment assignment error ϵ_{ti}. An ordered probit regression of y_i on treatment t_i with levels $v_{t1}, \ldots, v_{tT}$ has the form

$$y_i = \sum_{j=1}^{T} 1(t_i = v_{tj}) y_{ji}$$

where for $j = 1, \ldots, T$ and exogenous covariates $\mathbf{x}_i$

$$y_{ji} = v_h \quad \text{iff} \quad \kappa_{(h-1)j} < \mathbf{x}_i \boldsymbol{\beta}_j + \epsilon_{ji} \leq \kappa_{hj}$$

The values $v_1, \ldots, v_H$ are real numbers such that $v_h < v_m$ for $h < m$. For $j = 1, \ldots, T$, κ_{0j} is taken as $-\infty$ and κ_{Hj} is taken as $+\infty$.

This model can be formulated with or without different correlation parameters for each potential outcome. Potential-outcome specific parameters are obtained by specifying `pocorrelation` in the `entreat()` option.

If the correlation parameters are not potential-outcome specific, for $j = 1, \ldots, T$, ϵ_{ji} and ϵ_{ti} are bivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & \rho_{1t} \\ \rho_{1t} & 1 \end{bmatrix}$$

The treatment is exogenous if $\rho_{1t} = 0$. Note that we did not specify the structure of the correlations between the potential-outcome errors. We do not need information about these correlations to estimate POMs and treatment effects because all covariates and the outcome are observed in observations from each group.

From here, we discuss a model with an ordinal endogenous treatment. The results for binary treatment models are similar. Because the unobserved errors are bivariate normal, we can express the log likelihood in terms of the Φ_2^* function.

For $j = 1, \ldots, T$ and $h = 0, \ldots, H$, let

$$
c_{1ihj} = \begin{cases}
-\infty & h = 0 \\
\kappa_{hj} - \mathbf{x}_i \boldsymbol{\beta}_j & h = 1, \ldots, H - 1 \\
\infty & h = H
\end{cases}
$$

The lower and upper limits for the y_i probability are

$$
l_{1i} = c_{i(h-1)j} \quad \text{if} \quad y_i = v_h, t_i = v_{tj}
$$

and

$$
u_{1i} = c_{ihj} \quad \text{if} \quad y_i = v_h, t_i = v_{tj}
$$

The log likelihood for the model is

$$
\ln L = \sum_{i=1}^{N} w_i \ln \Phi_2^*([\, l_{1i} \quad l_{ti}\,], [\, u_{1i} \quad u_{ti}\,], \boldsymbol{\Sigma})
$$

where the lower and upper limits for the treatment probability, l_{ti} and u_{ti}, are defined in *Treatment* in [ERM] **eprobit**.

The conditional probability of obtaining treatment level v_{th} is

$$
\Pr(t_i = v_{th} | \mathbf{z}_{ti}) = \Phi_1^*(c_{ti(h-1)}, c_{tih}, 1)
$$

where the cutpoints for the treatment probabilities c_{tij} are defined in *Treatment* in [ERM] **eprobit**.

For $h = 1, \ldots, H$, the conditional probabilities for outcome level v_h at treatment level v_{tj} are

$$
\Pr(y_i = v_h | \mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_{tj}) = \frac{\Phi_2^*([\, c_{1i(h-1)j} \quad c_{ti(j-1)}\,], [\, c_{1ihj} \quad c_{tij}\,], \boldsymbol{\Sigma})}{\Phi_1^*(c_{ti(j-1)}, c_{tij}, 1)}
$$

The conditional POM for treatment group j and outcome category h is

$$
\text{POM}_{hj}(\mathbf{x}_i) = E\left\{1(y_{ji} = v_h) | \mathbf{x}_i\right\} = \Phi_1^*(c_{1i(h-1)j}, c_{1i(h-1)j}, 1)
$$

Conditional on the covariates $\mathbf{x}_i$ and $\mathbf{z}_{ti}$ and the treatment $t_i = v_m$, the POM for treatment group j and outcome category h is

$$
\begin{aligned}
\text{POM}_{hj}(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_m) &= E\left\{1(y_{ji} = v_h) | \mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_{tm}\right\} \\
&= \frac{\Phi_2^*([\, c_{1i(h-1)j} \quad c_{ti(m-1)}\,], [\, c_{1ihj} \quad c_{tim}\,], \boldsymbol{\Sigma})}{\Phi_1^*(c_{ti(m-1)}, c_{tim}, 1)}
\end{aligned}
$$

Without loss of generality, $t_i = v_{t1}$ corresponds to the control or base level of the treatment. Treatment effects are the differences between the potential outcomes $y_{2i}, \ldots, y_{Ti}$ and the control y_{1i}. When the potential outcomes are ordered probit, the treatment effect on a particular category is of interest.

The treatment effect of treatment group j on category h is $1(y_{ji} = v_h) - 1(y_{1i} = v_h)$, the difference in the outcome for individual i on being in category h if the individual receives the treatment $t_i = v_{tj}$ instead of the control $t_i = v_{t1}$. Evaluating this treatment effect lets us see how the treatment affects the probability of belonging to outcome category h.

For treatment group j, the treatment effect (TE) on category h conditioned on $\mathbf{x}_i$ is

$$
\begin{aligned}
\text{TE}_{hj}(\mathbf{x}_i) &= E\left\{1(y_{ji} = v_h) - 1(y_{1i} = v_h)|\mathbf{x}_i\right\} \\
&= \text{POM}_{hj}(\mathbf{x}_i) - \text{POM}_{h1}(\mathbf{x}_i)
\end{aligned}
$$

For treatment group j, the treatment effect on the treated (TET) on category h in treatment group m conditioned on $\mathbf{x}_i$ and $\mathbf{z}_{ti}$ is

$$
\begin{aligned}
\text{TET}_{hj}(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_m) &= E\left\{1(y_{ji} = v_h) - 1(y_{1i} = v_h)|\mathbf{x}_i, t_i = v_{t,m}\right\} \\
&= \text{POM}_{hj}(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_m) - \text{POM}_{h1}(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_m)
\end{aligned}
$$

We can take the expectation of these conditional predictions over the covariates to get population average parameters. The `estat teffects` or `margins` command is used to estimate the expectations as predictive margins once the model is fit with `eoprobit`. The POM for treatment group j and outcome category h is

$$
\text{POM}_{hj} = E\left\{1(y_{ji} = v_h)\right\} = E\left\{\text{POM}_{hj}(\mathbf{x}_i)\right\}
$$

The ATE for treatment group j and outcome category h is

$$
\text{ATE}_{hj} = E\left\{1(y_{ji} = v_h) - 1(y_{1i} = v_h)\right\} = E\left\{\text{TE}_{hj}(\mathbf{x}_i)\right\}
$$

For treatment group j, the average treatment effect on the treated (ATET) for outcome category h in treatment group m is

$$
\begin{aligned}
\text{ATET}_{hjm} &= E\left\{1(y_{ji} = v_h) - 1(y_{1i} = v_h)|t_i = v_m\right\} \\
&= E\left\{\text{TET}_{hj}(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_m)|t_i = v_m\right\}
\end{aligned}
$$

If the correlation parameters are potential-outcome specific, for $j = 1, \ldots, T$, ϵ_{ji} and ϵ_{ti} are bivariate normal with mean 0 and covariance

$$
\boldsymbol{\Sigma}_j = \begin{bmatrix} 1 & \rho_{j1t} \\ \rho_{j1t} & 1 \end{bmatrix}
$$

Now define

$$
\boldsymbol{\Sigma}_i = \sum_{j=1}^{T} 1(t_i = v_j)\boldsymbol{\Sigma}_j
$$

The log likelihood for the potential-outcome specification correlation model is

$$
\ln L = \sum_{i=1}^{N} w_i \ln \Phi_2^*([\, l_{1i} \quad l_{ti} \,], [\, u_{1i} \quad u_{ti} \,], \boldsymbol{\Sigma}_i)
$$

For $h = 1, \ldots, H$, the conditional probabilities for outcome level v_h at treatment level v_{tj} are now

$$\Pr(y_i = v_h | \mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_{tj}) = \frac{\Phi_2^*([\, c_{1i(h-1)j} \quad c_{ti(j-1)} \,], [\, c_{1ihj} \quad c_{tij} \,], \boldsymbol{\Sigma}_j)}{\Phi_1^*(c_{ti(j-1)}, c_{tij}, 1)}$$

The conditional POM for exogenous covariates $\mathbf{x}_i$, treatment group j, and outcome category h has the same definition as in the single correlation case. However, when we also condition on the treatment level $t_i = v_m$ and $\mathbf{z}_{ti}$, the POM for treatment group j and outcome category h is

$$
\begin{aligned}
\text{POM}_{hj}(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_m) &= E\left\{1(y_{ji} = v_h) | \mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_{tm}\right\} \\
&= \frac{\Phi_2^*([\, c_{1i(h-1)j} \quad c_{ti(m-1)} \,], [\, c_{1ihj} \quad c_{tim} \,], \boldsymbol{\Sigma}_j)}{\Phi_1^*(c_{ti(m-1)}, c_{tim}, 1)}
\end{aligned}
$$

Treatment effects are formulated as in the single correlation case but using these updated POM definitions. We can take the expectation of these conditional predictions over the covariates to get population-averaged parameters. The `estat teffects` or `margins` command is used to estimate the expectations as predictive margins once the model is fit with `eoprobit`.

Endogenous sample selection

Probit endogenous sample selection

An ordered probit model for outcome y_i with selection on s_i has the form

$$y_i = v_h \quad \text{iff} \quad \kappa_{h-1} < \mathbf{x}_i\boldsymbol{\beta} + \epsilon_i \leq \kappa_h$$

$$s_i = 1\left(\mathbf{z}_{si}\boldsymbol{\alpha}_s + \epsilon_{si} > 0\right)$$

where $\mathbf{x}_i$ are covariates that affect the outcome and $\mathbf{z}_{si}$ are covariates that affect selection. The outcome y_i is observed if $s_i = 1$ and is not observed if $s_i = 0$. The values $v_1, \ldots, v_H$ are real numbers such that $v_h < v_m$ for $h < m$. κ_0 is taken as $-\infty$ and κ_H is taken as $+\infty$.

The unobserved errors ϵ_i and ϵ_{si} are normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & \rho_{1s} \\ \rho_{1s} & 1 \end{bmatrix}$$

The lower and upper limits for the y_i probability, l_{1i} and u_{1i}, are as defined in (1)–(3). For the selection indicator, the lower and upper limits l_{si} and u_{si} are defined in *Probit endogenous sample selection* in [ERM] **eprobit**.

The log likelihood for the model is

$$\ln L = \sum_{i \in S} w_i \ln \Phi_2^*([\, l_{1i} \quad l_{si} \,], [\, u_{1i} \quad u_{si} \,], \boldsymbol{\Sigma}) + \sum_{i \notin S} w_i \ln \Phi_1^*(l_{si}, u_{si}, 1)$$

where S is the set of observations for which y_i is observed.

In this model, the probability of success is usually predicted conditional on the covariates $\mathbf{x}_i$ and not on the selection status s_i. The formulas for the conditional probability are thus the same as in (4).

The conditional probability of selection is

$$\Pr\left(s_i = 1 | \mathbf{z}_{si}\right) = \Phi_1^*(-\mathbf{z}_{si}\boldsymbol{\alpha}_s, \infty, 1)$$

Tobit endogenous sample selection

Instead of constraining the selection indicator to be binary, tobit endogenous sample selection uses a censored continuous sample-selection indicator. We allow the selection variable to be left- or right-censored.

An ordered probit model for outcome y_i with tobit selection on s_i has the form

$$y_i = v_h \quad \text{iff} \quad \kappa_{h-1} < \mathbf{x}_i \boldsymbol{\beta} + \epsilon_i \leq \kappa_h$$

where the values $v_1, \ldots, v_H$ are real numbers such that $v_h < v_m$ for $h < m$. κ_0 is taken as $-\infty$ and κ_H is taken as $+\infty$.

We observe the selection indicator s_i, which indicates the censoring status of the latent selection variable $s_i^\star$,

$$s_i^\star = \mathbf{z}_{si} \boldsymbol{\alpha}_s + \epsilon_{si}$$

$$s_i = \begin{cases} l_i & s_i^\star \leq l_i \\ s_i^\star & l_i < s_i^\star < u_i \\ u_i & s_i^\star \geq u_i \end{cases}$$

where $\mathbf{z}_{si}$ are covariates that affect selection and l_i and u_i are fixed lower and upper limits.

The outcome y_i is observed when $s_i^\star$ is not censored. If $l_i < s_i^\star < u_i$, then y_i is observed. y_i is not observed if $s_i^\star \leq l_i$, that is, if $s_i^\star$ is left-censored. y_i is also not observed if $s_i^\star$ is right-censored, $s_i^\star \geq u_i$. The unobserved errors ϵ_i and ϵ_{si} are normal with mean 0 and covariance

$$\begin{bmatrix} 1 & \rho_{1s}\sigma_s \\ \rho_{1s}\sigma_s & \sigma_s^2 \end{bmatrix}$$

For the selected observations, we can treat s_i as a continuous endogenous regressor, as in *Continuous endogenous covariates*. In fact, s_i may even be used as a regressor for y_i in `eoprobit` (specify `tobitselect(... main)`). On the nonselected observations, we treat s_i like the probit endogenous sample-selection indicator in *Probit endogenous sample selection*.

The conditional mean of ϵ_i is used in the lower and upper limits for the y_i probability for selected observations. Let

$$c_{i,h} = \begin{cases} -\infty & h = 0 \\ \kappa_h - \mathbf{x}_i \boldsymbol{\beta} - \rho_{1s}\sigma_s^{-1}(s_i - \mathbf{z}_{si}\alpha_c) & h = 1, \ldots, H-1 \\ \infty & h = H \end{cases}$$

The limits for the y_i probability for selected observations are

$$l_{1i} = c_{i(h-1)} \quad \text{if} \quad y_i = v_h$$

and

$$u_{1i} = c_{ih} \quad \text{if} \quad y_i = v_h$$

It follows that the log likelihood is

$$\ln L = \sum_{i \in S} w_i \left\{ \ln \Phi_1^*(l_{1i}, u_{1i}, 1 - \rho_{1s}^2) + \ln \phi(s_i - \mathbf{z}_{si}\boldsymbol{\alpha}_s, \sigma_s^2) \right\}$$
$$+ \sum_{i \in L} w_i \ln \Phi_1^*(l_{li}, u_{li}, 1)$$
$$+ \sum_{i \in U} w_i \ln \Phi_1^*(l_{ui}, u_{ui}, 1)$$

where S is the set of observations for which y_i is observed, L is the set of observations where $s_i^\star$ is left-censored, and U is the set of observations where $s_i^\star$ is right-censored. The lower and upper limits for selection—l_{li}, u_{li}, l_{ui}, and u_{ui}—are defined in *Tobit endogenous sample selection* in [ERM] **eprobit**.

The conditional probabilities on $s_i = S_i$ are

$$\Pr\left(y_i = v_h | \mathbf{x}_i\right) = \Phi_1^*(c_{i(h-1)}, c_{ih}, 1 - \rho_{1s}^2)$$

If we do not include s_i in the main outcome equation, the probability of success is calculated as (4) again.

Random effects

For an ordered probit regression with random effects, we observe panel data. For panel $i = 1, \ldots, N$ and observation $j = 1, \ldots, N_i$, an ordered probit regression of y_{ij} on covariates $\mathbf{x}_{ij}$ with random effect u_i may be written as

$$y_{ij} = v_h \quad \text{iff} \quad \kappa_{h-1} < \mathbf{x}_{ij}\boldsymbol{\beta} + \epsilon_{ij} + u_i \leq \kappa_h$$

The values $v_1, \ldots, v_H$ are real numbers such that $v_h < v_m$ for $h < m$. κ_0 is taken as $-\infty$ and κ_H is taken as $+\infty$. The random effect u_i is normal with mean 0 and variance σ_u^2. It is independent of the observation-level error ϵ_{ij}, which is standard normal.

We derive the likelihood by using the conditional density of y_{ij} on the random effect u_i and the marginal density of u_i. Multiplying them together, we have the joint density, which is integrated over u_i.

Let

$$l_{ij}(u) = \begin{bmatrix} 1(y_{ij} = v_1)\Phi\left(-\mathbf{x}_{ij}\boldsymbol{\beta} - u\right) \\ + \sum_{h=2}^{H-1} 1(y_{ij} = v_h)\left\{\Phi\left(\kappa_h - \mathbf{x}_{ij}\boldsymbol{\beta} - u\right) - \Phi\left(\kappa_{h-1} - \mathbf{x}_{ij}\boldsymbol{\beta} - u\right)\right\} \\ + 1(y_{ij} = v_H)\Phi\left(\mathbf{x}_{ij}\boldsymbol{\beta} + u\right) \end{bmatrix}$$

The likelihood for panel i is

$$L_i = \int_{-\infty}^{\infty} \phi\left(\frac{u_i}{\sigma_u}\right) \prod_{j=1}^{N_i} l_{ij}(u_i) du_i$$

We can approximate this integral using Gauss–Hermite quadrature. For q-point Gauss–Hermite quadrature, let the abscissa and weight pairs be denoted by (a_{ki}, w_{ki}), $k = 1, \ldots, q$. The Gauss–Hermite quadrature approximation is then

$$\int_{-\infty}^{\infty} f(x) \exp(-x^2) \, dx \approx \sum_{k=1}^{q} w_{ki} f(a_{ki})$$

The default approximation used by `xteoprobit` is mean–variance adaptive Gauss–Hermite quadrature. This chooses optimal abscissa and weights for each panel. See *Likelihood for multiequation models* in [ERM] **eprobit** for more information on the use of mean–variance adaptive Gauss–Hermite quadrature.

Using the quadrature approximation, the log likelihood is

$$\ln L = \sum_{i=1}^{N} \ln \left\{ \sum_{k=1}^{q} w_{ki} \prod_{j=1}^{N_i} l_{ij}(\sigma_u a_{ki}) \right\}$$

Now we will derive the conditional probabilities of success. These are similar to those given in *Introduction*, but the variance input to $\Phi_1^\star$ is the variance of the random effect plus the observation-level error.

First, let

$$\xi_{ij} = \epsilon_{ij} + u_i$$

ξ_{ij} is normal with mean 0 and variance $\sigma_\xi^2 = 1 + \sigma_u^2$.

Now, for $h = 0, \ldots, H$, define

$$c_{ijh} = \begin{cases} -\infty & h = 0 \\ (\kappa_h - \mathbf{x}_{ij}\boldsymbol{\beta}) & h = 1, \ldots, H-1 \\ \infty & h = H \end{cases}$$

For $h = 1, \ldots, H$, the conditional probabilities of success are

$$\Pr\left(y_{ij} = v_h | \mathbf{x}_{ij}\right) = \Phi_1^*(c_{ij(h-1)}, c_{ijh}, \sigma_\xi^2)$$

Combinations of features

Extended ordered probit regression models that involve multiple features can be formulated using the techniques discussed in *Likelihood for multiequation models* in [ERM] **eprobit**. Essentially, the density of the observed endogenous covariates can be written in terms of the unobserved normal errors. The observed endogenous and exogenous covariates determine the range of the errors, and the joint density can be evaluated as multivariate normal probabilities and densities.

Confidence intervals

The estimated variances will always be nonnegative, and the estimated correlations will always fall in $(-1, 1)$. We use transformations to obtain confidence intervals that accommodate these ranges.

We use the log transformation to obtain the confidence intervals for variance parameters and the atanh transformation to obtain confidence intervals for correlation parameters. For details, see *Confidence intervals* in [ERM] **eprobit**.

References

Amemiya, T. 1985. *Advanced Econometrics*. Cambridge, MA: Harvard University Press.

Bartus, T., and D. Roodman. 2014. Estimation of multiprocess survival models with cmp. *Stata Journal* 14: 756–777.

Botezat, A., and F. Pfeiffer. 2014. The impact of parents' migration on the well-being of children left behind: Initial evidence from Romania. IZA Discussion Paper No. 8225, Institute for the Study of Labor (IZA). http://ftp.iza.org/dp8225.pdf.

Chyi, H., and S. Mao. 2012. The determinants of happiness of China's elderly population. *Journal of Happiness Studies* 13: 167–185. https://doi.org/10.1007/s10902-011-9256-8.

De Luca, G., and V. Perotti. 2011. Estimation of ordered response models with sample selection. *Stata Journal* 11: 213–239.

Gregory, C. A. 2015. Estimating treatment effects for ordered outcomes using maximum simulated likelihood. *Stata Journal* 15: 756–774.

Heckman, J. J. 1976. The common structure of statistical models of truncation, sample selection and limited dependent variables and a simple estimator for such models. *Annals of Economic and Social Measurement* 5: 475–492.

———. 1979. Sample selection bias as a specification error. *Econometrica* 47: 153–161. https://doi.org/10.2307/1912352.

Maddala, G. S. 1983. *Limited-Dependent and Qualitative Variables in Econometrics*. Cambridge: Cambridge University Press.

Maddala, G. S., and L.-F. Lee. 1976. Recursive models with qualitative endogenous variables. *Annals of Economic and Social Measurement* 5: 525–545.

Roodman, D. 2011. Fitting fully observed recursive mixed-process models with cmp. *Stata Journal* 11: 159–206.

White, H. L., Jr. 1996. *Estimation, Inference and Specification Analysis*. Cambridge: Cambridge University Press.

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

———. 2014. Quasi-maximum likelihood estimation and testing for nonlinear models with endogenous explanatory variables. *Journal of Econometrics* 182: 226–234. https://doi.org/10.1016/j.jeconom.2014.04.020.

Also see

Title

> **eoprobit postestimation** — Postestimation tools for eoprobit and xteoprobit

Postestimation commands

The following postestimation command is of special interest after `eoprobit` and `xteoprobit`:

Command	Description
estat teffects	treatment effects and potential-outcome means

The following standard postestimation commands are also available after `eoprobit` and `xteo-probit`:

Command	Description
contrast	contrasts and ANOVA-style joint tests of estimates
estat ic	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC)
estat summarize	summary statistics for the estimation sample
estat vce	variance–covariance matrix of the estimators (VCE)
† estat (svy)	postestimation statistics for survey data
estimates	cataloging estimation results
etable	table of estimation results
* forecast	dynamic forecasts and simulations
* hausman	Hausman's specification test
lincom	point estimates, standard errors, testing, and inference for linear combinations of coefficients
* lrtest	likelihood-ratio test
margins	marginal means, predictive margins, marginal effects, and average marginal effects
marginsplot	graph the results from margins (profile plots, interaction plots, etc.)
nlcom	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
predict	means, probabilities, treatment effects, etc.
predictnl	point estimates, standard errors, testing, and inference for generalized predictions
pwcompare	pairwise comparisons of estimates
† suest	seemingly unrelated estimation
test	Wald tests of simple and composite linear hypotheses
testnl	Wald tests of nonlinear hypotheses

*`forecast`, `hausman`, and `lrtest` are not appropriate with `svy` estimation results.

†`suest` and the survey data `estat` commands are not available after `xteoprobit`.

predict

Predictions after `eoprobit` and `xteoprobit` are described in

[ERM] **eoprobit predict** describes the most commonly used predictions. If you fit a model with treatment effects, predictions specifically related to these models are detailed in [ERM] **predict treatment**. [ERM] **predict advanced** describes less commonly used predictions, such as predictions of outcomes in auxiliary equations.

margins

Description for margins

`margins` estimates statistics based on fitted models. These statistics include marginal means, marginal probabilities, potential-outcome means, average and conditional derivatives, average and conditional effects, and treatment effects.

Menu for margins

Statistics > Postestimation

Syntax for margins

> margins [*marginlist*] [, *options*]
>
> margins [*marginlist*] , <u>predict</u>(*statistic* ...) [<u>predict</u>(*statistic* ...) ...] [*options*]

statistic	Description
Main	
pr	probability for binary or ordinal y_j; the default
<u>m</u>ean	mean
<u>pom</u>ean	potential-outcome mean
te	treatment effect
tet	treatment effect on the treated
xb	linear prediction excluding all complications
pr(*a*,*b*)	$\Pr(a < y_j < b)$ for continuous y_j
e(*a*,*b*)	$E(y_j \mid a < y_j < b)$ for continuous y_j
<u>y</u>star(*a*,*b*)	$E(y_j^*)$, $y_j^* = \max\{a, \min(y_j, b)\}$ for continuous y_j

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

For the full syntax, see [R] **margins**.

Remarks and examples

See [ERM] **Intro 7** for an overview of using `margins` and `predict` after `eoprobit` and `xteoprobit`. For examples using `margins`, `predict`, and `estat teffects`, see *Interpreting effects* in [ERM] **Intro 9** and see [ERM] **Example 1a**.

Methods and formulas

This section contains methods and formulas for the default `asf` prediction. Methods and formulas for other predictions are given in *Methods and formulas* of [ERM] **eoprobit**. We begin with the cross-sectional model, and then we extend our discussion to the random-effect models that we use for panel data.

In the ordered probit model for exogenous covariates $\mathbf{x}_i$ and endogenous regressors $\mathbf{w}_i$, we have

$$y_i = v_h \quad \text{iff} \quad \kappa_{h-1} < \mathbf{x}_i \boldsymbol{\beta} + \mathbf{w}_i \boldsymbol{\beta}_2 + \epsilon_i \leq \kappa_h$$

The values $v_1, \ldots, v_H$ are real numbers such that $v_h < v_m$ for $h < m$. κ_0 is taken as $-\infty$ and κ_H is taken as $+\infty$. The error ϵ_i is standard normal and correlated with $\mathbf{w}_i$.

Because ϵ_i is a normally distributed, mean 0, random variable, we can split it into two mean 0, normally distributed, independent parts,

$$\epsilon_i = u_i + \psi_i$$

where $u_i = \boldsymbol{\gamma} \epsilon_{2i}$ is the unobserved heterogeneity that gives rise to the endogeneity and ψ_i is an idiosyncratic error term with variance σ_ψ^2.

For $h = 0, \ldots, H$, define

$$c_{ih} = \begin{cases} -\infty & h = 0 \\ \kappa_h - \mathbf{x}_i \boldsymbol{\beta} - \mathbf{w}_i \boldsymbol{\beta}_2 - u_i & h = 1, \ldots, H-1 \\ \infty & h = H \end{cases}$$

Conditional on the covariates and the unobserved heterogeneity, we have

$$E\{\mathbf{1}(y_i = v_h) | \mathbf{x}_i, \mathbf{w}_i, u_i\} = \Pr(y_i = v_h | \mathbf{x}_i, \mathbf{w}_i, u_i)$$
$$= \Phi_1^*(c_{i(h-1)}, c_{ih}, \sigma_\psi^2)$$

Predictions and effects are computed based on the expression above. Including u_i controls for endogeneity. Thus, all effects computed using the expression above have a structural interpretation. See Imbens and Newey (2009) and Wooldridge (2010) for a detailed description of structural functions for models with endogeneity.

Our discussion easily extends to models for panel data with random effects. In this case, we have N panels. Panel $i = 1, \ldots, N$ has observations $t = 1, \ldots, N_i$, so we observe y_{it} with random effect α_i and observation-level error ϵ_{it}. These errors are independent of each other. So the combined error $\xi_{it} = \alpha_i + \epsilon_{it}$ is normal with mean 0 and variance $1 + \sigma_\alpha^2$, where σ_α^2 is the variance of α_i. The results discussed earlier can then be applied using the combined error ξ_{it} rather than the cross-sectional error.

All predictions after `xteoprobit` assume the panel-level random effects (α_i) are zero. Put another way, predictions condition on the random effects being set to their means.

References

Imbens, G. W., and W. K. Newey. 2009. Identification and estimation of triangular simultaneous equations models without additivity. *Econometrica* 77: 1481–1512. https://doi.org/10.3982/ECTA7108.

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

Also see

[ERM] **eoprobit** — Extended ordered probit regression

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

[ERM] **predict treatment** — predict for treatment statistics

[ERM] **predict advanced** — predict's advanced features

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

[U] **20 Estimation and postestimation commands**

Title

eoprobit predict — predict after eoprobit and xteoprobit

Description

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

Syntax

You previously fit the model

    eoprobit y x1 ..., ...

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

$$\Pr(y_i = m) = \Pr(c_{m-1} \leq \mathbf{x}_i\beta + e_i.\mathbf{y} \leq c_m)$$

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

    xteoprobit y x1 ..., ...

Then the main equation would be

$$\Pr(y_{ij} = m) = \Pr(c_{m-1} \leq \mathbf{x}_{ij}\beta + u_i.\mathbf{y} + v_{ij}.\mathbf{y} \leq c_m)$$

In either case, note that the equation produces a probability for each ordered category recorded in y. We will call the ordered categories outcomes. If there are M outcomes, a probability is produced for each of m, $m = 1$ to M outcomes. `predict` calculates predictions for the probabilities 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, $e_{ij}.\mathbf{y} = u_i.\mathbf{y} + v_{ij}.\mathbf{y}$.

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

The syntax of `predict` is

> `predict` [*type*] { *newvar* | *stub** | *newvarlist* } [*if*] [*in*]
>> [, *statistic asfmethod counterfactual*]

statistic	Description
Main	
pr	probability of each outcome; the default
xb	linear prediction excluding all complications
outlevel(*#*)	calculate probability for $m = \#$ only

You specify one or M new variables with `pr`, where M is the number of outcomes. If you specify one new variable and you do not specify `outlevel()`, then `outlevel(#1)` is assumed.

You specify one new variable with `xb`.

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

`pr`, the default, calculates the predicted probability for all outcomes or for a specific outcome. To compute probabilities for all outcomes, you specify M new variables, where M is the number of categories of the dependent variable. Alternatively, you can specify *stub**, in which case `pr` will store predicted probabilities in variables *stub*1, *stub*2, ..., *stub*M. To compute the probability for a specific outcome, you specify one new variable and, optionally, the outcome value in option `outlevel()`; if you omit `outlevel()`, the first outcome value, `outlevel(#1)`, is assumed.

  In each observation, the predictions are the probabilities 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.

`outlevel(#)` specifies to calculate only the probability for outcome $m = $ # rather than calculating M probabilities, one for each ordered category (outcome) recorded in the dependent variable. If you do not specify this option and y records three outcomes, you type

  `. predict p1 p2 p3`

to obtain the probabilities for each outcome. If you want only the probability of the third outcome, you can type

  `. predict p3, outlevel(#3)`

If the third outcome corresponded to `y==3`, you could instead type

  `. predict p3, outlevel(3)`

If the third outcome corresponded to `y==57`, you could instead type

  `. predict p3, outlevel(57)`

Most users number the outcomes 1, 2, and 3. Some users number them 0, 1, and 2. You could even number them 3, 5, and 57. Stata does not care how they are numbered.

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 probability of one or more ordered categories (outcomes) recorded in the dependent variable.

   `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 $\beta_0 + \beta_1 \mathtt{x1}_i + \cdots$. Use of `target()` is discussed in *Remarks and examples* of [ERM] **eregress predict**.

Remarks and examples

Remarks are presented under the following headings:

> *Using predict after eoprobit and xteoprobit*
> *How to think about nonlinear models*

Using predict after eoprobit and xteoprobit

`eoprobit` and `xteoprobit` fit ordinal probit models. The outcome variable y takes on various values such as 1, 2, 3, and 4, and each represents an ordered category, such as cannot walk, walks with difficulty, walks with few problems, and walks well. When you use `predict` after `eoprobit` or `xteoprobit`, remember to specify variables corresponding to each category.

```
. predict p1 p2 p3 p4
```

Alternatively, specify the `outlevel(#)` option.

With this exception, predictions after fitting models with `eoprobit` and `xteoprobit` are handled the same as they are after fitting models with `eregress` and `xteregress`. The issues are the same. See [ERM] **eregress predict**.

How to think about nonlinear models

What we wrote in [ERM] **eprobit predict** applies equally to the use of `predict` after `eoprobit` and `xteoprobit`. We wrote

Probit is a nonlinear model, and yet we just said that predictions after fitting models with `eprobit` and `xteprobit` are handled the same as they are after fitting models with `eregress` and `xteregress`. That statement is partly true, not misleading, but false in its details.

The regression-based discussion that we routed you to is framed in terms of expected values. In the nonlinear models, it needs to be framed in terms of distributional assumptions about the errors. For instance, `predict` after `eprobit` does not predict the expected value (mean) of $e_i.\mathtt{y}$. It calculates the probability that $e_i.\mathtt{y}$ exceeds $-\mathbf{x}_i\boldsymbol{\beta}$. These details matter hugely in implementation but can be glossed over for understanding the issues. For a full treatment of the issues, see *Methods and formulas* of [ERM] **eprobit**.

Methods and formulas

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

Also see

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

[ERM] **eoprobit** — Extended ordered probit regression

Title

eprobit — Extended probit regression

Description

eprobit fits a probit regression model that accommodates any combination of endogenous co-variates, nonrandom treatment assignment, and endogenous sample selection. Continuous, binary, and ordinal endogenous covariates are allowed. Treatment assignment may be endogenous or exogenous. A probit or tobit model may be used to account for endogenous sample selection.

xteprobit fits a random-effects probit regression model that accommodates endogenous covariates, treatment, and sample selection in the same way as eprobit and also accounts for correlation of observations within panels or within groups.

Quick start

Probit regression of y on x with continuous endogenous covariate y2 modeled by x and z

    eprobit y x, endogenous(y2 = x z)

Same as above, but adding continuous endogenous covariate y3 modeled by x and z2

    eprobit y x, endogenous(y2 = x z) endogenous(y3 = x z2)

Probit regression of y on x with binary endogenous covariate d modeled by x and z

    eprobit y x, endogenous(d = x z, probit)

Probit regression of y on x with endogenous treatment recorded in trtvar and modeled by x and z

    eprobit y x, entreat(trtvar = x z)

Probit regression of y on x with exogenous treatment recorded in trtvar

    eprobit y x, extreat(trtvar)

Random-effects probit regression of y on x using [xtset](#) data

    xteprobit y x

Probit regression of y on x with endogenous sample-selection indicator selvar modeled by x and z

    eprobit y x, select(selvar = x z)

Same as above, but adding endogenous covariate y2 modeled by x and z2

    eprobit y x, select(selvar = x z) endogenous(y2 = x z2)

Same as above, but adding endogenous treatment recorded in trtvar and modeled by x and z3

    eprobit y x, select(selvar = x z) endogenous(y2 = x z2) ///
        entreat(trtvar = x z3)

Same as above, but with random effects and without endogenous treatment

    xteprobit y x, select(selvar = x z) endogenous(y2 = x z2)

131

Menu

eprobit

Statistics > Endogenous covariates > Models adding selection and treatment > Probit regression

xteprobit

Statistics > Longitudinal/panel data > Endogenous covariates > Models adding selection and treatment > Probit regression (RE)

Syntax

Basic probit regression with endogenous covariates

> eprobit *depvar* [*indepvars*] , <u>endog</u>enous(*depvars*$_{en}$ = *varlist*$_{en}$) [*options*]

Basic probit regression with endogenous treatment assignment

> eprobit *depvar* [*indepvars*] , <u>entr</u>eat(*depvar*$_{tr}$ [= *varlist*$_{tr}$]) [*options*]

Basic probit regression with exogenous treatment assignment

> eprobit *depvar* [*indepvars*] , <u>extr</u>eat(*tvar*) [*options*]

Basic probit regression with sample selection

> eprobit *depvar* [*indepvars*] , <u>sel</u>ect(*depvar*$_s$ = *varlist*$_s$) [*options*]

Basic probit regression with tobit sample selection

> eprobit *depvar* [*indepvars*] , <u>tobitsel</u>ect(*depvar*$_s$ = *varlist*$_s$) [*options*]

Basic probit regression with random effects

> xteprobit *depvar* [*indepvars*] [, *options*]

Probit regression combining endogenous covariates, treatment, and selection

> eprobit *depvar* [*indepvars*] [*if*] [*in*] [*weight*] [, *extensions options*]

Probit regression combining random effects, endogenous covariates, treatment, and selection

> xteprobit *depvar* [*indepvars*] [*if*] [*in*] [, *extensions options*]

extensions	Description
Model	
endogenous(*enspec*)	model for endogenous covariates; may be repeated
entreat(*entrspec*)	model for endogenous treatment assignment
extreat(*extrspec*)	exogenous treatment
select(*selspec*)	probit model for selection
tobitselect(*tselspec*)	tobit model for selection

options	Description
Model	
noconstant	suppress constant term
offset(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1
constraints(*numlist*)	apply specified linear constraints
SE/Robust	
vce(*vcetype*)	*vcetype* may be oim, robust, cluster *clustvar*, opg, bootstrap, or jackknife
Reporting	
level(*#*)	set confidence level; default is level(95)
nocnsreport	do not display constraints
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Integration	
intpoints(*#*)	set the number of integration (quadrature) points for integration over four or more dimensions; default is intpoints(128)
triintpoints(*#*)	set the number of integration (quadrature) points for integration over three dimensions; default is triintpoints(10)
reintpoints(*#*)	set the number of integration (quadrature) points for random-effects integration; default is reintpoints(7)
reintmethod(*intmethod*)	integration method for random effects; *intmethod* may be mvaghermite (the default) or ghermite
Maximization	
maximize_options	control the maximization process; seldom used
collinear	keep collinear variables
coeflegend	display legend instead of statistics

enspec is *depvars*$_{en}$ = *varlist*$_{en}$ [, *enopts*]

   where *depvars*$_{en}$ is a list of endogenous covariates. Each variable in *depvars*$_{en}$ specifies an endogenous covariate model using the common *varlist*$_{en}$ and options.

entrspec is *depvar*$_{tr}$ [= *varlist*$_{tr}$] [, *entropts*]

   where *depvar*$_{tr}$ is a variable indicating treatment assignment. *varlist*$_{tr}$ is a list of covariates predicting treatment assignment.

extrspec is *tvar* $\begin{bmatrix} , & extropts \end{bmatrix}$

where *tvar* is a variable indicating treatment assignment.

selspec is *depvar$_s$* = *varlist$_s$* $\begin{bmatrix} , & selopts \end{bmatrix}$

where *depvar$_s$* is a variable indicating selection status. *depvar$_s$* must be coded as 0, indicating that the observation was not selected, or 1, indicating that the observation was selected. *varlist$_s$* is a list of covariates predicting selection.

tselspec is *depvar$_s$* = *varlist$_s$* $\begin{bmatrix} , & tselopts \end{bmatrix}$

where *depvar$_s$* is a continuous variable. *varlist$_s$* is a list of covariates predicting *depvar$_s$*. The censoring status of *depvar$_s$* indicates selection, where a censored *depvar$_s$* indicates that the observation was not selected and a noncensored *depvar$_s$* indicates that the observation was selected.

enopts	Description
Model	
probit	treat endogenous covariate as binary
oprobit	treat endogenous covariate as ordinal
pocorrelation	estimate different correlations for each level of a binary or an ordinal endogenous covariate
nomain	do not add endogenous covariate to main equation
nore	do not include random effects in model for endogenous covariate
noconstant	suppress constant term

nore is available only with xteprobit.

entropts	Description
Model	
pocorrelation	estimate different correlations for each potential outcome
nomain	do not add treatment indicator to main equation
nointeract	do not interact treatment with covariates in main equation
nore	do not include random effects in model for endogenous treatment
noconstant	suppress constant term
offset(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1

nore is available only with xteprobit.

extropts	Description
Model	
pocorrelation	estimate different correlations for each potential outcome
nomain	do not add treatment indicator to main equation
nointeract	do not interact treatment with covariates in main equation

selopts	Description
Model	
<u>nore</u>	do not include random effects in selection model
<u>nocons</u>tant	suppress constant term
<u>off</u>set(*varname_o*)	include *varname_o* in model with coefficient constrained to 1

nore is available only with xteprobit.

tselopts	Description
Model	
*ll(*varname* | #)	left-censoring variable or limit
*ul(*varname* | #)	right-censoring variable or limit
main	add censored selection variable to main equation
<u>nore</u>	do not include random effects in tobit selection model
<u>nocons</u>tant	suppress constant term
<u>off</u>set(*varname_o*)	include *varname_o* in model with coefficient constrained to 1

* You must specify either ll() or ul().

nore is available only with xteprobit.

indepvars, *varlist*_{en}, *varlist*_{tr}, and *varlist*_s may contain factor variables; see [U] **11.4.3 Factor variables**.

depvar, *indepvars*, *depvars*_{en}, *varlist*_{en}, *depvar*_{tr}, *varlist*_{tr}, *tvar*, *depvar*_s, and *varlist*_s may contain time-series operators; see [U] **11.4.4 Time-series varlists**.

bootstrap, by, collect, jackknife, and statsby are allowed with eprobit and xteprobit. rolling and svy are allowed with eprobit. See [U] **11.1.10 Prefix commands**.

Weights are not allowed with the bootstrap prefix; see [R] **bootstrap**.

vce() and weights are not allowed with the svy prefix; see [SVY] **svy**.

fweights, iweights, and pweights are allowed with eprobit; see [U] **11.1.6 weight**.

reintpoints() and reintmethod() are available only with xteprobit.

collinear and coeflegend do not appear in the dialog box.

See [U] **20 Estimation and postestimation commands** for more capabilities of estimation commands.

Options

> __ Model __

endogenous(*enspec*), entreat(*entrspec*), extreat(*extrspec*), select(*selspec*),
   tobitselect(*tselspec*); see [ERM] **ERM options**.

noconstant, offset(*varname_o*), constraints(*numlist*); see [R] **Estimation options**.

> __ SE/Robust __

vce(*vcetype*); see [ERM] **ERM options**.

> __ Reporting __

level(#), nocnsreport; see [R] **Estimation options**.

display_options: noci, <u>nopv</u>alues, <u>noomit</u>ted, vsquish, noemptycells, <u>base</u>levels,
   <u>allbase</u>levels, <u>nofvlabel</u>, fvwrap(#), fvwrapon(*style*), cformat(%*fmt*), pformat(%*fmt*),
   sformat(%*fmt*), and nolstretch; see [R] **Estimation options**.

> Integration

intpoints(*#*), triintpoints(*#*), reintpoints(*#*), reintmethod(*intmethod*); see [ERM] **ERM options**.

> Maximization

maximize_options: <u>di</u>fficult, <u>tech</u>nique(*algorithm_spec*), <u>iter</u>ate(*#*), [no]log, <u>trace</u>, gradient, showstep, <u>hess</u>ian, <u>showtol</u>erance, <u>tol</u>erance(*#*), <u>ltol</u>erance(*#*), <u>nrtol</u>erance(*#*), <u>nonrtol</u>erance, and from(*init_specs*); see [R] **Maximize**.

> The default technique for eprobit is technique(nr). The default technique for xteprobit is technique(bhhh 10 nr 2).

> Setting the optimization type to technique(bhhh) resets the default *vcetype* to vce(opg).

The following options are available with eprobit and xteprobit but are not shown in the dialog box:

collinear, coeflegend; see [R] **Estimation options**.

Remarks and examples

eprobit and xteprobit fit models that we refer to as "extended probit regression models", meaning that they accommodate endogenous covariates, nonrandom treatment assignment, endogenous sample selection, and panel data or other grouped data.

eprobit fits models for cross-sectional data (one-level models). eprobit can account for endogenous covariates, treatment, and sample selection, whether these complications arise individually or in combination.

xteprobit fits random-effects models (two-level models) for panel data or grouped data. xteprobit accounts for endogenous covariates, treatment, and sample selection in the same way as eprobit and also accounts for within-panel or within-group correlation among observations.

In this entry, you will find information on the syntax for the eprobit and xteprobit commands. You can see *Methods and formulas* for a full description of the models that can be fit with eprobit and xteprobit and details about how those models are fit.

More information on extended probit regression models is found in the separate introductions and example entries. We recommend reading those entries to learn how to use eprobit and xteprobit. Below, we provide a guide to help you locate the ones that will be helpful to you.

For an introduction to eprobit and xteprobit and the other extended regression commands for continuous, interval, and ordinal outcomes, see [ERM] **Intro 1**–[ERM] **Intro 9**.

[ERM] **Intro 1** introduces the ERM commands, the problems they address, and their syntax.

[ERM] **Intro 2** provides background on the four types of models—linear regression, interval regression, probit regression, and ordered probit regression—that can be fit using ERM commands.

[ERM] **Intro 3** considers the problem of endogenous covariates and how to solve it using ERM commands.

[ERM] **Intro 4** gives an overview of endogenous sample selection and using ERM commands to account for it.

[ERM] **Intro 5** covers nonrandom treatment assignment and how to account for it using eprobit or any of the other ERM commands.

[ERM] **Intro 6** covers random-effects models for panel data and other grouped data. It discusses xteprobit and the other ERM commands for panel data.

[ERM] **Intro 7** discusses interpretation of results. You can interpret coefficients from eprobit and xteprobit in the usual way, but this introduction goes beyond the interpretation of coefficients. We demonstrate how to find answers to interesting questions by using margins. If your model includes an endogenous covariate or an endogenous treatment, the use of margins differs from its use after other estimation commands, so we strongly recommend reading this intro if you are fitting these types of models.

[ERM] **Intro 8** will be particularly helpful if you are familiar with ivprobit, heckprobit, xtprobit, and other commands that address endogenous covariates, sample selection, nonrandom treatment assignment, or random effects. This introduction is a Rosetta stone that maps the syntax of those commands to the syntax of eprobit and xteprobit.

[ERM] **Intro 9** walks you through an example that gives insight into the concepts of endogenous covariates, treatment assignment, and sample selection while fitting models with eregress that address these complications. Although the example uses eregress, the discussion applies equally to eprobit. This intro also demonstrates how to interpret results by using margins and estat teffects.

Additional examples are presented in [ERM] **Example 1a**–[ERM] **Example 9**. For examples using eprobit, see

[ERM] **Example 3a**	Probit regression with continuous endogenous covariate
[ERM] **Example 3b**	Probit regression with endogenous covariate and treatment
[ERM] **Example 4a**	Probit regression with endogenous sample selection
[ERM] **Example 4b**	Probit regression with endogenous treatment and sample selection
[ERM] **Example 5**	Probit regression with endogenous ordinal treatment
[ERM] **Example 9**	Probit regression with endogenous treatment and random effects

See *Examples* in [ERM] **Intro** for an overview of all the examples. All examples may be interesting because they handle complications in the same way.

eprobit and xteprobit fit many models discussed in the literature. This includes the probit model with continuous endogenous covariates (Newey 1987), the probit model with multiple endogenous binary covariates (Arendt and Holm 2006), the probit model with an endogenous treatment (Angrist 2001 and Pindyck and Rubinfeld 1998), and the random-effects probit model (Conway 1990). eprobit can also be used for probit models with selection, such as that discussed by Van de Ven and Van Pragg (1981), and for the model with a tobit selection equation, discussed in Wooldridge (2010, sec. 19.7).

xteprobit can be used for the random-effects probit model with selection discussed in Semykina and Wooldridge (2018). The bivariate probit model with random effects discussed in Mulkay (2015) may also be fit using xteprobit. Roodman (2011) investigated probit models with endogenous covariates and endogenous sample selection and demonstrated how multiple observational data complications could be addressed with a triangular model structure. He and Tamás Bartus showed how random effects could be used in the triangular model structure in Bartus and Roodman (2014). Roodman's work has been used to model processes like the impact of finance on the probability of being an entrepreneur (Karymshakov, Sultakeev, and Sulaimanova 2015) and the impact of foreign direct investment on the probability of creating an innovative product (Vahter 2011).

Stored results

eprobit stores the following in e():

Scalars
    e(N)   number of observations
    e(N_selected)   number of selected observations
    e(N_nonselected)   number of nonselected observations
    e(k)   number of parameters
    e(k_cat#)   number of categories for the #th *depvar*, ordinal
    e(k_eq)   number of equations in e(b)
    e(k_eq_model)   number of equations in overall model test
    e(k_dv)   number of dependent variables
    e(k_aux)   number of auxiliary parameters
    e(df_m)   model degrees of freedom
    e(ll)   log likelihood
    e(N_clust)   number of clusters
    e(chi2) χ^2
    e(p)   *p*-value for model test
    e(n_quad)   number of integration points for multivariate normal
    e(n_quad3)   number of integration points for trivariate normal
    e(rank)   rank of e(V)
    e(ic)   number of iterations
    e(rc)   return code
    e(converged)   1 if converged, 0 otherwise

Macros
    e(cmd)   eprobit
    e(cmdline)   command as typed
    e(depvar)   names of dependent variables
    e(tsel_ll)   left-censoring limit for tobit selection
    e(tsel_ul)   right-censoring limit for tobit selection
    e(wtype)   weight type
    e(wexp)   weight expression
    e(title)   title in estimation output
    e(clustvar)   name of cluster variable
    e(offset#)   offset for the #th *depvar*, where # is determined by equation order in output
    e(chi2type)   Wald; type of model χ^2 test
    e(vce)   *vcetype* specified in vce()
    e(vcetype)   title used to label Std. err.
    e(opt)   type of optimization
    e(which)   max or min; whether optimizer is to perform maximization or minimization
    e(ml_method)   type of ml method
    e(user)   name of likelihood-evaluator program
    e(technique)   maximization technique
    e(properties)   b V
    e(estat_cmd)   program used to implement estat
    e(predict)   program used to implement predict
    e(marginsok)   predictions allowed by margins
    e(marginsnotok)   predictions disallowed by margins
    e(asbalanced)   factor variables fvset as asbalanced
    e(asobserved)   factor variables fvset as asobserved

Matrices
    e(b)   coefficient vector
    e(cat#)   categories for the #th *depvar*, ordinal
    e(Cns)   constraints matrix
    e(ilog)   iteration log (up to 20 iterations)
    e(gradient)   gradient vector
    e(V)   variance–covariance matrix of the estimators
    e(V_modelbased)   model-based variance

Functions
    e(sample)   marks estimation sample

In addition to the above, the following is stored in `r()`:

Matrices
    `r(table)`   matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in `r()` are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

`xteprobit` stores the following in `e()`:

Scalars
`e(N)`	number of observations
`e(N_g)`	number of groups
`e(N_selected)`	number of selected observations
`e(N_nonselected)`	number of nonselected observations
`e(k)`	number of parameters
`e(k_cat#)`	number of categories for the #th *depvar*, ordinal
`e(k_eq)`	number of equations in `e(b)`
`e(k_eq_model)`	number of equations in overall model test
`e(k_dv)`	number of dependent variables
`e(k_aux)`	number of auxiliary parameters
`e(df_m)`	model degrees of freedom
`e(ll)`	log likelihood
`e(N_clust)`	number of clusters
`e(chi2)`	χ^2
`e(p)`	p-value for model test
`e(n_quad)`	number of integration points for multivariate normal
`e(n_quad3)`	number of integration points for trivariate normal
`e(n_requad)`	number of integration points for random effects
`e(g_min)`	smallest group size
`e(g_avg)`	average group size
`e(g_max)`	largest group size
`e(rank)`	rank of `e(V)`
`e(ic)`	number of iterations
`e(rc)`	return code
`e(converged)`	1 if converged, 0 otherwise

Macros
`e(cmd)`	`xteprobit`
`e(cmdline)`	command as typed
`e(depvar)`	names of dependent variables
`e(tsel_ll)`	left-censoring limit for tobit selection
`e(tsel_ul)`	right-censoring limit for tobit selection
`e(ivar)`	variable denoting groups
`e(title)`	title in estimation output
`e(clustvar)`	name of cluster variable
`e(offset#)`	offset for the #th *depvar*, where # is determined by equation order in output
`e(chi2type)`	Wald; type of model χ^2 test
`e(vce)`	*vcetype* specified in `vce()`
`e(vcetype)`	title used to label Std. err.
`e(reintmethod)`	integration method for random effects
`e(opt)`	type of optimization
`e(which)`	`max` or `min`; whether optimizer is to perform maximization or minimization
`e(ml_method)`	type of ml method
`e(user)`	name of likelihood-evaluator program
`e(technique)`	maximization technique
`e(properties)`	b V
`e(estat_cmd)`	program used to implement `estat`
`e(predict)`	program used to implement `predict`
`e(marginsok)`	predictions allowed by `margins`
`e(marginsnotok)`	predictions disallowed by `margins`
`e(asbalanced)`	factor variables fvset as asbalanced
`e(asobserved)`	factor variables fvset as asobserved

Matrices
    e(b)   coefficient vector
    e(cat#)   categories for the #th *depvar*, ordinal
    e(Cns)   constraints matrix
    e(ilog)   iteration log (up to 20 iterations)
    e(gradient)   gradient vector
    e(V)   variance–covariance matrix of the estimators
    e(V_modelbased)   model-based variance
Functions
    e(sample)   marks estimation sample

In addition to the above, the following is stored in r():

Matrices
    r(table)   matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Methods and formulas

The methods and formulas presented here are for the probit model. The estimators implemented in eprobit and xteprobit are maximum likelihood estimators covered by the results in chapter 13 of Wooldridge (2010) and White (1996).

The log-likelihood functions maximized by eprobit and xteprobit are implied by the triangular structure of the model. Specifically, the joint distribution of the endogenous variables is a product of conditional and marginal distributions because the model is triangular. For a few of the many relevant applications of this result in literature, see chapter 10 of Amemiya (1985); Heckman (1976, 1979); chapter 5 of Maddala (1983); Maddala and Lee (1976); sections 15.7.2, 15.7.3, 16.3.3, 17.5.2, and 19.7.1 in Wooldridge (2010); and Wooldridge (2014). Roodman (2011) and Bartus and Roodman (2014) used this result to derive the formulas discussed below.

Methods and formulas are presented under the following headings:

> *Introduction*
> *Endogenous covariates*
>   *Continuous endogenous covariates*
>   *Binary and ordinal endogenous covariates*
> *Treatment*
> *Endogenous sample selection*
>   *Probit endogenous sample selection*
>   *Tobit endogenous sample selection*
> *Random effects*
> *Combined model*
> *Confidence intervals*
> *Likelihood for multiequation models*

Introduction

A probit regression of outcome y_i on covariates $\mathbf{x}_i$ may be written as

$$y_i = 1\left(\mathbf{x}_i\boldsymbol{\beta} + \epsilon_i > 0\right)$$

where the errors ϵ_i are distributed as standard normal. The log likelihood is

$$\ln L = \sum_{i=1}^{N} w_i \left\{ y_i \ln\Phi\left(\mathbf{x}_i\boldsymbol{\beta}\right) + (1 - y_i)\ln\Phi\left(-\mathbf{x}_i\boldsymbol{\beta}\right) \right\}$$

where w_i are the weights. The conditional probability of success is

$$E\left(y_i|\mathbf{x}_i\right) = \Pr\left(y_i = 1|\mathbf{x}_i\right) = \Phi\left(\mathbf{x}_i\boldsymbol{\beta}\right)$$

The standard normal cumulative distribution function $\Phi(\cdot)$ used in these expressions is a one-sided probability that the random variable is below a certain point. In the models we describe later, it will be useful to use two-sided probabilities. For two-sided probabilities, we define Φ_d^* with three inputs. The first two inputs are d-dimensional row vectors $\mathbf{l}$ and $\mathbf{u}$ that have values in $IR \cup \{-\infty, \infty\}$, the extended real line. The final input is a $d \times d$ real-valued and positive-definite matrix $\boldsymbol{\Sigma}$.

$$\Phi_d^*(\mathbf{l}, \mathbf{u}, \boldsymbol{\Sigma}) = \int_{l_1}^{u_1} \ldots \int_{l_d}^{u_d} \phi_d(\boldsymbol{\epsilon}, \boldsymbol{\Sigma}) \, d\epsilon_1 \ldots d\epsilon_d$$

where ϕ_d is the density of a mean 0, multivariate normal random variable. For details on the calculation of Φ_d^*, see [M-5] **mvnormal( )**. The probabilities are approximated using numeric integration. The number of integration or quadrature points can be varied to attain better approximations. For trivariate errors, we use the method of Drezner (1994). For four or more errors, we use the method of Miwa, Hayter, and Kuriki (2003).

The lower and upper limits l_{1i} and u_{1i} on the unobserved ϵ_i are based on the observed values of y_i and $\mathbf{x}_i$ and are defined as

$$l_{1i} = \begin{cases} -\infty & y_i = 0 \\ -\mathbf{x}_i\boldsymbol{\beta} & y_i = 1 \end{cases} \qquad u_{1i} = \begin{cases} -\mathbf{x}_i\boldsymbol{\beta} & y_i = 0 \\ \infty & y_i = 1 \end{cases} \tag{1}$$

They let us rewrite the log likelihood concisely as

$$\ln L = \sum_{i=1}^{N} w_i \ln\Phi_1^*(l_{1i}, u_{1i}, 1)$$

The conditional probability of success can be written using similar notation:

$$\Pr\left(y_i = 1|\mathbf{x}_i\right) = \Phi_1^*(-\mathbf{x}_i\boldsymbol{\beta}, \infty, 1) \tag{2}$$

Endogenous covariates

Continuous endogenous covariates

A probit regression of y_i on exogenous covariates $\mathbf{x}_i$ and C continuous endogenous covariates $\mathbf{w}_{ci}$ has the form

$$y_i = 1 \left(\mathbf{x}_i \boldsymbol{\beta} + \mathbf{w}_{ci} \boldsymbol{\beta}_c + \epsilon_i > 0 \right)$$

$$\mathbf{w}_{ci} = \mathbf{z}_{ci} \mathbf{A}_c + \boldsymbol{\epsilon}_{ci}$$

The vector $\mathbf{z}_{ci}$ contains variables from $\mathbf{x}_i$ and other covariates that affect $\mathbf{w}_{ci}$. The unobserved errors ϵ_i and $\boldsymbol{\epsilon}_{ci}$ are multivariate normal with mean 0 and covariance

$$\begin{bmatrix} 1 & \boldsymbol{\sigma}'_{1c} \\ \boldsymbol{\sigma}_{1c} & \boldsymbol{\Sigma}_c \end{bmatrix}$$

We can write the joint density of the dependent variables as a product:

$$f(y_i, \mathbf{w}_{ci} | \mathbf{x}_i, \mathbf{z}_{ci}) = f(y_i | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{ci}) f(\mathbf{w}_{ci} | \mathbf{x}_i, \mathbf{z}_{ci})$$

The conditional density of $\mathbf{w}_{ci}$ is

$$f(\mathbf{w}_{ci} | \mathbf{x}_i, \mathbf{z}_{ci}) = \phi_C(\mathbf{w}_{ci} - \mathbf{z}_{ci} \mathbf{A}_c, \boldsymbol{\Sigma}_c)$$

Note that

$$\Pr(y_i = 1 | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{ci}) = \Pr(\mathbf{x}_i \boldsymbol{\beta} + \mathbf{w}_{ci} \boldsymbol{\beta}_c + \epsilon_i > 0 | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{ci})$$

So the conditional density of y_i can be written as a probability for ϵ_i. Thus, the conditional distribution of ϵ_i can be used to find the conditional density of y_i. Conditional on the endogenous and exogenous covariates, ϵ_i has mean and variance

$$E\left(\epsilon_i | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{ci}\right) = \boldsymbol{\sigma}'_{1c} \boldsymbol{\Sigma}_c^{-1} \left(\mathbf{w}_{ci} - \mathbf{z}_{ci} \mathbf{A}_c\right)'$$

$$\mathrm{Var}\left(\epsilon_i | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{ci}\right) = 1 - \boldsymbol{\sigma}'_{1c} \boldsymbol{\Sigma}_c^{-1} \boldsymbol{\sigma}_{1c}$$

The conditional mean is used in the lower and upper limits for the y_i probability, which are

$$l_{1i} = \begin{cases} -\infty & y_i = 0 \\ -\mathbf{x}_i \boldsymbol{\beta} - \boldsymbol{\sigma}'_{1c} \boldsymbol{\Sigma}_c^{-1} \left(\mathbf{w}_{ci} - \mathbf{z}_{ci} \mathbf{A}_c\right)' & y_i = 1 \end{cases}$$

$$u_{1i} = \begin{cases} -\mathbf{x}_i \boldsymbol{\beta} - \boldsymbol{\sigma}'_{1c} \boldsymbol{\Sigma}_c^{-1} \left(\mathbf{w}_{ci} - \mathbf{z}_{ci} \mathbf{A}_c\right)' & y_i = 0 \\ \infty & y_i = 1 \end{cases}$$

Using these limits, the conditional variance, and the conditional density of $\mathbf{w}_{ci}$, we obtain the log likelihood

$$\ln L = \sum_{i=1}^N w_i \left\{ \ln \Phi_1^* \left(l_{1i}, u_{1i}, 1 - \boldsymbol{\sigma}'_{1c} \boldsymbol{\Sigma}_c^{-1} \boldsymbol{\sigma}_{1c} \right) + \ln \phi_C(\mathbf{w}_{ci} - \mathbf{z}_{ci} \mathbf{A}_c, \boldsymbol{\Sigma}_c) \right\}$$

Letting

$$l_{1i1} = -\mathbf{x}_i \boldsymbol{\beta} - \boldsymbol{\sigma}'_{1c} \boldsymbol{\Sigma}_c^{-1} \left(\mathbf{w}_{ci} - \mathbf{z}_{ci} \mathbf{A}_c\right)'$$

$$u_{1i1} = \infty$$

the conditional probability of success is

$$\Pr\left(y_i = 1 | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{ci}\right) = \Phi_1^*(l_{1i1}, u_{1i1}, 1 - \boldsymbol{\sigma}'_{1c} \boldsymbol{\Sigma}_c^{-1} \boldsymbol{\sigma}_{1c})$$

Binary and ordinal endogenous covariates

Here we begin by formulating the probit regression of y_i on exogenous covariates $\mathbf{x}_i$ and B binary and ordinal endogenous covariates $\mathbf{w}_{bi} = [w_{b1i}, \ldots, w_{bBi}]$. Indicator (dummy) variables for the levels of each binary and ordinal covariate are used in the model. You can also interact other covariates with the binary and ordinal endogenous covariates, as in treatment-effect models.

Let $j = 1, \ldots, B$. We use a probit model for binary endogenous covariates

$$w_{bji} = 1\left(\mathbf{z}_{bji}\boldsymbol{\alpha}_{bj} + \epsilon_{bji} > 0\right)$$

For ordinal endogenous covariate w_{bji} that takes values $v_{bj1}, \ldots, v_{bjB_j}$ with covariates $\mathbf{z}_{bji}$, we have the ordered probit model

$$w_{bji} = v_{bjh} \quad \text{iff} \quad \kappa_{bj(h-1)} < \mathbf{z}_{bji}\boldsymbol{\alpha}_{bj} + \epsilon_{bji} \leq \kappa_{bjh} \tag{3}$$

The values $v_{bj1}, \ldots, v_{bjB_j}$ are real numbers such that $v_{bjh} < v_{bjm}$ for $h < m$. κ_{bj0} is taken as $-\infty$ and κ_{bjB_j} is taken as $+\infty$. The errors $\epsilon_{b1i}, \ldots, \epsilon_{bBi}$ are multivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma}_b = \begin{bmatrix} 1 & \rho_{b12} & \cdots & \rho_{b1B} \\ \rho_{b12} & 1 & \cdots & \rho_{b2B} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{b1B} & \rho_{b2B} & \cdots & 1 \end{bmatrix}$$

Because the covariate w_{bji} is binary or ordinal, the effect of each category in the outcome equation is made with an indicator variable.

$$\mathbf{wind}_{bji} = \begin{bmatrix} 1(w_{bji} = v_{bj1}) \\ \vdots \\ 1(w_{bji} = v_{bjB_j}) \end{bmatrix}' \tag{4}$$

The model for the outcome can be formulated with or without different correlation parameters for each level of $\mathbf{w}_{bi}$. Level-specific parameters are obtained by specifying `pocorrelation` in the `endogenous()` option.

If the correlation parameters are not level specific, we have

$$y_i = 1(\mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB} + \epsilon_i > 0)$$

where the outcome error ϵ_i and binary and ordinal endogenous errors $\epsilon_{b1i}, \ldots, \epsilon_{bBi}$ are multivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & \boldsymbol{\rho}'_{1b} \\ \boldsymbol{\rho}_{1b} & \boldsymbol{\Sigma}_b \end{bmatrix}$$

From here, we discuss the model with ordinal endogenous covariates. The results for binary endogenous covariates are similar.

For $j = 1, \ldots, B$ and $h = 0, \ldots, B_j$, let

$$c_{bjih} = \begin{cases} -\infty & h = 0 \\ \kappa_{bjh} - \mathbf{z}_{bji}\boldsymbol{\alpha}_{bj} & h = 1, \ldots, B_j - 1 \\ \infty & h = B_j \end{cases}$$

The probability for w_{bji} has lower limit

$$l_{bji} = c_{bji(h-1)} \quad \text{if} \quad w_{bji} = v_{bjh} \tag{5}$$

and upper limit

$$u_{bji} = c_{bjih} \quad \text{if} \quad w_{bji} = v_{bjh} \tag{6}$$

Letting

$$c_{bi} = -\mathbf{x}_i\boldsymbol{\beta} - \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} - \cdots - \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB}$$

the lower and upper limits for the y_i probability are

$$l_{1i} = \begin{cases} -\infty & y_i = 0 \\ c_{bi} & y_i = 1 \end{cases} \qquad u_{1i} = \begin{cases} c_{bi} & y_i = 0 \\ \infty & y_i = 1 \end{cases}$$

and

$$\mathbf{l}_i = \begin{bmatrix} l_{1i} & l_{b1i} & \ldots & l_{bBi} \end{bmatrix}$$
$$\mathbf{u}_i = \begin{bmatrix} u_{1i} & u_{b1i} & \ldots & u_{bBi} \end{bmatrix}$$

The log likelihood for this model is

$$\ln L = \sum_{i=1}^{N} w_i \ln\Phi_{B+1}^*(\mathbf{l}_i, \mathbf{u}_i, \boldsymbol{\Sigma})$$

Now let

$$\mathbf{l}_{bi} = \begin{bmatrix} l_{b1i} & \ldots & l_{bBi} \end{bmatrix}$$
$$\mathbf{u}_{bi} = \begin{bmatrix} u_{b1i} & \ldots & u_{bBi} \end{bmatrix}$$
$$\mathbf{l}_{i1} = \begin{bmatrix} -\infty & \mathbf{l}_{bi} \end{bmatrix}$$
$$\mathbf{u}_{i1} = \begin{bmatrix} c_{bi} & \mathbf{u}_{bi} \end{bmatrix}$$

The conditional probability of success is

$$\Pr(y_i = 1 | \mathbf{x}_i, \mathbf{z}_{b1i}, \ldots, \mathbf{z}_{bBi}, \mathbf{w}_{bi}) = \frac{\Phi_{B+1}^*(\mathbf{l}_{i1}, \mathbf{u}_{i1}, \boldsymbol{\Sigma})}{\Phi_B^*(\mathbf{l}_{bi}, \mathbf{u}_{bi}, \boldsymbol{\Sigma}_b)}$$

When the endogenous ordinal variables are different treatments, holding the correlation parameters constant over the treatment levels is a constrained form of the potential-outcome model. In an unconstrained potential-outcome model, the correlations between the outcome and the treatments—the endogenous ordinal regressors $\mathbf{w}_{bi}$—vary over the levels of each treatment.

In this unconstrained model, there is a different potential-outcome error for each level of each treatment. For example, when the endogenous treatment variable w_1 has three levels (0, 1, and 2) and the endogenous treatment variable w_2 has four levels (0, 1, 2, and 3), the unconstrained model has $12 = 3 \times 4$ outcome errors. Because there is a different correlation between each potential outcome and each endogenous treatment, there are 2×12 correlation parameters between the potential outcomes and the treatments in this example model.

We denote the number of different combinations of values for the endogenous treatments $\mathbf{w}_{bi}$ by M, and we denote the vector of values in each combination by $\mathbf{v}_j$ ($j \in \{1, 2, \ldots, M\}$). Letting k_{wp} be the number of levels of endogenous ordinal treatment variable $p \in \{1, 2, \ldots, B\}$ implies that $M = k_{w1} \times k_{w2} \times \cdots \times k_{wB}$.

Denoting the outcome errors $\epsilon_{1i}, \ldots, \epsilon_{Mi}$, we have

$$y_{1i} = 1(\mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB} + \epsilon_{1i} > 0)$$

$$\vdots$$

$$y_{Mi} = 1(\mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB} + \epsilon_{Mi} > 0)$$

$$y_i = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j)y_{ji}$$

For $j = 1, \ldots, M$, the outcome error ϵ_{ji} and the endogenous errors $\epsilon_{b1i}, \ldots, \epsilon_{bBi}$ are multivariate normal with 0 mean and covariance

$$\boldsymbol{\Sigma}_j = \begin{bmatrix} 1 & \boldsymbol{\rho}'_{j1b} \\ \boldsymbol{\rho}_{j1b} & \boldsymbol{\Sigma}_b \end{bmatrix}$$

Now let

$$\boldsymbol{\Sigma}_{i,b} = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j)\boldsymbol{\Sigma}_j$$

Now the log likelihood for this model is

$$\ln L = \sum_{i=1}^{N} w_i \ln \Phi^*_{B+1}(\mathbf{l}_i, \mathbf{u}_i, \boldsymbol{\Sigma}_{i,b})$$

The conditional probability of success is

$$\Pr(y_i = 1|\mathbf{x}_i, \mathbf{z}_{b1i}, \ldots, \mathbf{z}_{bBi}, \mathbf{w}_{bi}) = \frac{\Phi^*_{B+1}(\mathbf{l}_{i1}, \mathbf{u}_{i1}, \boldsymbol{\Sigma}_{i,b})}{\Phi^*_B(\mathbf{l}_{bi}, \mathbf{u}_{bi}, \boldsymbol{\Sigma}_b)}$$

Treatment

In the potential-outcomes framework, the treatment t_i is a discrete variable taking T values, indexing the T potential outcomes of the outcome y_i: $y_{1i}, \ldots, y_{Ti}$.

When we observe treatment t_i with levels $v_1, \ldots, v_T$, we have

$$y_i = \sum_{j=1}^{T} 1(t_i = v_j)y_{ji}$$

So for each observation, we observe only the potential outcome associated with that observation's treatment value.

For exogenous treatments, our approach is equivalent to the regression adjustment treatment-effect estimation method. See [CAUSAL] **teffects intro advanced**. We do not model the treatment assignment process. The formulas for the treatment effects and potential-outcome means (POMs) are equivalent to what we provide here for endogenous treatments. The treatment effect on the treated for $\mathbf{x}_i$ for an exogenous treatment is equivalent to what we provide here for the endogenous treatment when the correlation parameter between the outcome and treatment errors is set to 0. The average treatment effects (ATEs) and POMs for exogenous treatments are estimated as predictive margins in an analogous manner to what we describe here for endogenous treatments.

From here, we assume an endogenous treatment t_i. For ordinal treatment t_i with covariates $\mathbf{z}_{ti}$, we have the ordered probit model

$$t_i = v_h \quad \text{iff} \quad \kappa_{h-1} < \mathbf{z}_{ti}\boldsymbol{\alpha}_t + \epsilon_{ti} \leq \kappa_h \tag{7}$$

The treatment values $v_1, \ldots, v_T$ are real numbers such that $v_h < v_m$ for $h < m$. κ_0 is taken as $-\infty$ and κ_T is taken as $+\infty$. The treatment error ϵ_{ti} is standard normal.

We use a probit model for binary treatments that take values in $\{0, 1\}$,

$$t_i = 1\left(\mathbf{z}_{ti}\boldsymbol{\alpha}_t + \epsilon_{ti} > 0\right)$$

A probit regression of y_i on exogenous covariates $\mathbf{x}_i$ and endogenous treatment t_i taking values $v_1, \ldots, v_T$ has the form

$$y_{1i} = 1\left(\mathbf{x}_i\boldsymbol{\beta}_1 + \epsilon_{1i} > 0\right)$$

$$\vdots$$

$$y_{Ti} = 1\left(\mathbf{x}_i\boldsymbol{\beta}_T + \epsilon_{Ti} > 0\right)$$

$$y_i = \sum_{j=1}^{T} 1(t_i = v_j)y_{ji}$$

This model can be formulated with or without different correlation parameters for each potential outcome. Potential-outcome specific parameters are obtained by specifying `pocorrelation` in the `entreat()` option.

If the correlation parameters are not potential-outcome specific, for $j = 1, \ldots, T$, ϵ_{ji} and ϵ_{ti} are bivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & \rho_{1t} \\ \rho_{1t} & 1 \end{bmatrix}$$

The treatment is exogenous if $\rho_{1t} = 0$. Note that we did not specify the structure of the correlations between the potential-outcome errors. We do not need information about these correlations to estimate POMs and treatment effects because all covariates and the outcome are observed in observations from each group.

From here, we discuss a model with an ordinal endogenous treatment. The results for binary treatment models are similar. Because the unobserved errors are bivariate normal, we can express the log likelihood in terms of the Φ_2^* function.

For $j = 1, \ldots, T$, let

$$c_{1ij} = -\mathbf{x}_i\boldsymbol{\beta}_j$$

The lower and upper limits for the y_i probability are

$$l_{1i} = \begin{cases} -\infty & y_i = 0 \\ c_{1ij} & y_i = 1, t_i = v_j \end{cases} \qquad u_{1i} = \begin{cases} c_{1ij} & y_i = 0, t_i = v_j \\ \infty & y_i = 1 \end{cases}$$

For $j = 0, \ldots, T$, define

$$
c_{tij} = \begin{cases} -\infty & j = 0 \\ \kappa_j - \mathbf{z}_{ti}\boldsymbol{\alpha}_t & j = 1, \ldots, T-1 \\ \infty & j = T \end{cases}
$$

So for the t_i probability, we have lower limit

$$
l_{ti} = c_{ti(j-1)} \quad \text{if} \quad t_i = v_j \tag{8}
$$

and upper limit

$$
u_{ti} = c_{tij} \quad \text{if} \quad t_i = v_j \tag{9}
$$

The log likelihood for the model is

$$
\ln L = \sum_{i=1}^{N} w_i \ln \Phi_2^*([l_{1i} \quad l_{ti}], [u_{1i} \quad u_{ti}], \boldsymbol{\Sigma})
$$

The conditional probability of obtaining treatment level v_h is

$$
\Pr(t_i = v_h | \mathbf{z}_{ti}) = \Phi_1^*(c_{ti(h-1)}, c_{tih}, 1)
$$

The conditional probability of success at treatment level v_j is

$$
\Pr(y_i = 1 | \mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_j) = \frac{\Phi_2^*([c_{1ij} \quad c_{ti(j-1)}], [\infty \quad c_{tij}], \boldsymbol{\Sigma})}{\Phi_1^*(c_{ti(j-1)}, c_{tij}, 1)}
$$

The conditional POM for treatment group j is

$$
\text{POM}_j(\mathbf{x}_i) = E(y_{ji} | \mathbf{x}_i) = \Phi_1^*(c_{1ij}, \infty, 1)
$$

Conditional on the covariates $\mathbf{x}_i$ and $\mathbf{z}_{ti}$ and the treatment $t_i = v_h$, the POM for treatment group j is

$$
\begin{aligned}
\text{POM}_j(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_h) &= E(y_{ji} | \mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_h) \\
&= \frac{\Phi_2^*([c_{1ij} \quad c_{ti(h-1)}], [\infty \quad c_{tih}], \boldsymbol{\Sigma})}{\Phi_1^*(c_{ti(h-1)}, c_{tih}, 1)}
\end{aligned}
$$

The treatment effect $y_{ji} - y_{1i}$ is the difference in the outcome for individual i if the individual receives the treatment $t_i = v_j$ instead of the control $t_i = v_1$ and what the difference would have been if the individual received the control treatment instead.

For treatment group j, the treatment effect (TE) conditioned on $\mathbf{x}_i$ is

$$
\text{TE}_j(\mathbf{x}_i) = E(y_{ji} - y_{1i} | \mathbf{x}_i) = \text{POM}_j(\mathbf{x}_i) - \text{POM}_1(\mathbf{x}_i)
$$

For treatment group j, the treatment effect on the treated (TET) in treatment group h conditioned on $\mathbf{x}_i$ and $\mathbf{z}_{ti}$ is

$$\text{TET}_j(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_h) = E\left(y_{ji} - y_{1i} | \mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_h\right)$$
$$= \text{POM}_j(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_h) - \text{POM}_1(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_h)$$

We can take the expectation of these conditional predictions over the covariates to get population average parameters. The [margins](#) command is used to estimate the expectations as predictive margins once the model is fit with eprobit. The POM for treatment group j is

$$\text{POM}_j = E\left(y_{ji}\right) = E\left\{\text{POM}_j(\mathbf{x}_i)\right\}$$

The ATE for treatment group j is

$$\text{ATE}_j = E\left(y_{ji} - y_{1i}\right) = E\left\{\text{TE}_j(\mathbf{x}_i)\right\}$$

For treatment group j, the average treatment effect on the treated (ATET) in treatment group h is

$$\text{ATET}_{jh} = E\left(y_{ji} - y_{1i} | t_i = v_h\right)$$
$$= E\left\{\text{TET}_j(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_h) | t_i = v_h\right\}$$

If the correlation parameters are potential-outcome specific, for $j = 1, \ldots, T$, ϵ_{ji} and ϵ_{ti} are bivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma}_j = \begin{bmatrix} 1 & \rho_{j1t} \\ \rho_{j1t} & 1 \end{bmatrix}$$

Now define

$$\boldsymbol{\Sigma}_i = \sum_{j=1}^{T} 1(t_i = v_j)\boldsymbol{\Sigma}_j$$

The log likelihood for the potential-outcome specification correlation model is

$$\ln L = \sum_{i=1}^{N} w_i \ln \Phi_2^*([\,l_{1i} \quad l_{ti}\,], [\,u_{1i} \quad u_{ti}\,], \boldsymbol{\Sigma}_i)$$

The conditional probability of success at treatment level v_j is

$$\Pr(y_i = 1 | \mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_j) = \frac{\Phi_2^*([\,c_{1ij} \quad c_{ti(j-1)}\,], [\,\infty \quad c_{tij}\,], \boldsymbol{\Sigma}_j)}{\Phi_1^*(c_{ti(j-1)}, c_{tij}, 1)}$$

The conditional POM for exogenous covariates $\mathbf{x}_i$ and treatment group j has the same definition as in the single correlation case. However, when we also condition on the treatment level $t_i = v_h$ and $\mathbf{z}_{ti}$, the POM for treatment group j is

$$\text{POM}_j(\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_h) = E\left(y_{ji} | \mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_h\right)$$
$$= \frac{\Phi_2^*([\,c_{1ij} \quad c_{ti(h-1)}\,], [\,\infty \quad c_{tih}\,], \boldsymbol{\Sigma}_j)}{\Phi_1^*(c_{ti(h-1)}, c_{tih}, 1)}$$

Treatment effects are formulated as in the single correlation case but using these updated POM definitions. We can take the expectation of these conditional predictions over the covariates to get population-averaged parameters. The [estat teffects](#) or [margins](#) command is used to estimate the expectations as predictive margins once the model is fit with eprobit.

Endogenous sample selection

Probit endogenous sample selection

A probit model for outcome y_i with selection on s_i has the form

$$y_i = 1\,(\mathbf{x}_i\boldsymbol{\beta} + \epsilon_i > 0)$$
$$s_i = 1\,(\mathbf{z}_{si}\boldsymbol{\alpha}_s + \epsilon_{si} > 0)$$

where $\mathbf{x}_i$ are covariates that affect the outcome and $\mathbf{z}_{si}$ are covariates that affect selection. The outcome y_i is observed if $s_i = 1$ and not observed if $s_i = 0$. The unobserved errors ϵ_i and ϵ_{si} are normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & \rho_{1s} \\ \rho_{1s} & 1 \end{bmatrix}$$

The lower and upper limits for the y_i probability, l_{1i} and u_{1i}, are as defined in (1). For the selection indicator, we have lower and upper limits

$$l_{si} = \begin{cases} -\infty & s_i = 0 \\ -\mathbf{z}_{si}\boldsymbol{\alpha}_s & s_i = 1 \end{cases} \qquad u_{si} = \begin{cases} -\mathbf{z}_{si}\boldsymbol{\alpha}_s & s_i = 0 \\ \infty & s_i = 1 \end{cases} \tag{10}$$

The log likelihood for the model is

$$\ln L = \sum_{i \in S} w_i \ln\Phi_2^*([\,l_{1i} \quad l_{si}\,], [\,u_{1i} \quad u_{si}\,], \boldsymbol{\Sigma}) \quad +$$
$$\sum_{i \notin S} w_i \ln\Phi_1^*(l_{si}, u_{si}, 1)$$

where S is the set of observations for which y_i is observed.

In this model, the probability of success is usually predicted conditional on the covariates $\mathbf{x}_i$ and not on the selection status s_i. The formulas for the conditional probability are thus the same as in (2).

The conditional probability of selection is

$$\Pr\,(s_i = 1 | \mathbf{z}_{si}) = \Phi_1^*(-\mathbf{z}_{si}\boldsymbol{\alpha}_s, \infty, 1)$$

Tobit endogenous sample selection

Instead of constraining the selection indicator to be binary, tobit endogenous sample selection uses a censored continuous sample-selection indicator. We allow the selection variable to be left- or right-censored.

A probit model for outcome y_i with tobit selection on s_i has the form

$$y_i = 1\,(\mathbf{x}_i\boldsymbol{\beta} + \epsilon_i > 0)$$

We observe the selection indicator s_i, which indicates the censoring status of the latent selection variable $s_i^\star$,

$$s_i^\star = \mathbf{z}_{si}\boldsymbol{\alpha}_s + \epsilon_{si}$$

$$s_i = \begin{cases} l_i & s_i^\star \leq l_i \\ s_i^\star & l_i < s_i^\star < u_i \\ u_i & s_i^\star \geq u_i \end{cases}$$

where $\mathbf{z}_{si}$ are covariates that affect selection and l_i and u_i are fixed lower and upper limits.

The outcome y_i is observed when $s_i^\star$ is not censored ($l_i < s_i^\star < u_i$). The outcome y_i is not observed when $s_i^\star$ is left-censored ($s_i^\star \leq l_i$) or $s_i^\star$ is right-censored ($s_i^\star \geq u_i$). The unobserved errors ϵ_i and ϵ_{si} are normal with mean 0 and covariance

$$\begin{bmatrix} 1 & \rho_{1s}\sigma_s \\ \rho_{1s}\sigma_s & \sigma_s^2 \end{bmatrix}$$

For the selected observations, we can treat s_i as a continuous endogenous regressor, as in *Continuous endogenous covariates*. In fact, s_i may even be used as a regressor for y_i in eprobit (specify tobitselect(... main)). On the nonselected observations, we treat s_i like the probit endogenous sample-selection indicator in *Probit endogenous sample selection*.

For nonselected observations, we have

$$\Pr(s_i^\star \leq l_i | \mathbf{z}_{si}, \mathbf{x}_i) = \Pr(\mathbf{z}_{si}\boldsymbol{\alpha}_s + \epsilon_{si} \leq l_i)$$
$$= \Phi\left(\frac{l_i - \mathbf{z}_{si}\boldsymbol{\alpha}_s}{\sigma_s}\right)$$

and

$$\Pr(s_i^\star \geq u_i | \mathbf{z}_{si}, \mathbf{x}_i) = \Pr(\mathbf{z}_{si}\boldsymbol{\alpha}_s + \epsilon_{si} \geq u_i)$$
$$= \Phi\left(\frac{\mathbf{z}_{si}\boldsymbol{\alpha}_s - u_i}{\sigma_s}\right)$$

The lower and upper limits for the s_i probability for nonselected observations where $s_i^\star$ is left-censored are

$$l_{li} = -\infty$$
$$u_{li} = \frac{l_i - \mathbf{z}_{si}\boldsymbol{\alpha}_s}{\sigma_s}$$

The lower and upper limits for the s_i probability for nonselected observations where $s_i^\star$ is right-censored are

$$l_{ui} = \frac{u_i - \mathbf{z}_{si}\boldsymbol{\alpha}_s}{\sigma_s}$$
$$u_{ui} = \infty$$

Now we consider the selected observations. For $s_i = s_i^\star = S_i$, we can write the joint density of the dependent variables as a product,

$$f(y_i, s_i = S_i | \mathbf{x}_i, \mathbf{z}_{si}) = f(y_i | s_i = S_i, \mathbf{x}_i, \mathbf{z}_{si}) f(s_i = S_i | \mathbf{x}_i, \mathbf{z}_{si})$$

The marginal density of $s_i = S_i$ is

$$f(s_i = S_i | \mathbf{x}_i, \mathbf{z}_{s,i}) = \phi(S_i - \mathbf{z}_{si}\boldsymbol{\alpha}_s, \sigma_s^2)$$

The conditional density of y_i can be written as a probability for ϵ_i. Thus, the conditional distribution of ϵ_i can be used to find the conditional density of y_i. Conditional on $s_i = S_i$, ϵ_i has mean and variance

$$E\left(\epsilon_i | s_i = S_i, \mathbf{x}_i, \mathbf{z}_{si}\right) = \rho_{1s}\sigma_s^{-1}\left(S_i - \mathbf{z}_{si}\boldsymbol{\alpha}\right)$$
$$\text{Var}\left(\epsilon_i | s_i = S_i, \mathbf{x}_i, \mathbf{z}_{si}\right) = 1 - \rho_{1,s}^2$$

The conditional mean is used in the lower and upper limits for the y_i probability for selected observations, which are

$$l_{1i} = \begin{cases} -\infty & y_i = 0 \\ -\mathbf{x}_i\boldsymbol{\beta} - \rho_{1s}\sigma_s^{-1}\left(s_i - \mathbf{z}_{si}\boldsymbol{\alpha}\right) & y_i = 1 \end{cases}$$

$$u_{1i} = \begin{cases} -\mathbf{x}_i\boldsymbol{\beta} - \rho_{1s}\sigma_s^{-1}\left(s_i - \mathbf{z}_{si}\boldsymbol{\alpha}\right) & y_i = 0 \\ \infty & y_i = 1 \end{cases}$$

It follows that the log likelihood is

$$\ln L = \sum_{i \in S} w_i \left\{ \ln \Phi_1^*(l_{1i}, u_{1i}, 1 - \rho_{1s}^2) + \ln \phi(s_i - \mathbf{z}_{si}\boldsymbol{\alpha}_s, \sigma_s^2) \right\}$$
$$+ \sum_{i \in L} w_i \ln \Phi_1^*(l_{li}, u_{li}, 1)$$
$$+ \sum_{i \in U} w_i \ln \Phi_1^*(l_{ui}, u_{ui}, 1)$$

where S is the set of observations for which y_i is observed, L is the set of observations where $s_i^\star$ is left-censored, and U is the set of observations where $s_i^\star$ is right-censored.

The probability of success conditional on $s_i = s_i^\star = S_i$ is

$$\Pr(y_i = 1 | \mathbf{x}_i, s_i = s_i^\star = S_i) = \Phi_1^*\{-\mathbf{x}_i\boldsymbol{\beta} - \rho_{1s}\sigma_s^{-1}\left(S_i - \mathbf{z}_{si}\boldsymbol{\alpha}\right), \infty, 1 - \rho_{1s}^2\}$$

If we do not include s_i in the main outcome equation, the probability of success is calculated as (2) again.

Random effects

For a probit regression with random effects, we observe panel data. For panel $i = 1, \ldots, N$ and observation $j = 1, \ldots, N_i$, a probit regression of outcome y_{ij} on covariates $\mathbf{x}_{ij}$ may be written as

$$y_{ij} = 1\left(\mathbf{x}_{ij}\boldsymbol{\beta} + \epsilon_{ij} + u_i > 0\right)$$

The random effect u_i is normal with mean 0 and variance σ_u^2. It is independent of the observation-level error ϵ_{ij}, which is standard normal.

We derive the likelihood by using the conditional density of y_{ij} on the random effect u_i and the marginal density of u_i. Multiplying them together, we have the joint density, which is integrated over u_i.

Let

$$l_{ij}(u) = y_{ij}\Phi\left(\mathbf{x}_{ij}\boldsymbol{\beta} + u\right) + (1 - y_{ij})\Phi\left(-\mathbf{x}_{ij}\boldsymbol{\beta} - u\right)$$

The likelihood for panel i is

$$L_i = \int_{-\infty}^{\infty} \phi\left(\frac{u_i}{\sigma_u}\right) \prod_{j=1}^{N_i} l_{ij}(u_i)du_i$$

We can approximate this integral using Gauss–Hermite quadrature. For q-point Gauss–Hermite quadrature, let the abscissa and weight pairs be denoted by (a_{ki}, w_{ki}), $k = 1, \ldots, q$. Then, the Gauss–Hermite quadrature approximation is

$$\int_{-\infty}^{\infty} f(x)\exp(-x^2)\,dx \approx \sum_{k=1}^{q} w_{ki}f(a_{ki})$$

The default approximation used by xteprobit is mean–variance adaptive Gauss–Hermite quadrature. This chooses optimal abscissa and weights for each panel. See *Likelihood for multiequation models* in [ERM] **eprobit** for more information on the use of mean–variance adaptive Gauss–Hermite quadrature.

Using the quadrature approximation, the log likelihood is

$$\ln L = \sum_{i=1}^{N} \ln\left\{\sum_{k=1}^{q} w_{ki}\prod_{j=1}^{N_i} l_{ij}(\sigma_u a_{ki})\right\}$$

Now we will derive the conditional probability of success. This is similar to what was given in *Introduction*, but the variance input to $\Phi_1^{\star}$ is the variance of the random effect plus the observation-level error.

First, let

$$\xi_{ij} = \epsilon_{ij} + u_i$$

where ξ_{ij} is normal with mean 0 and variance $\sigma_\xi^2 = 1 + \sigma_u^2$.

Then, the conditional probability of success is

$$\Pr\left(y_{ij} = 1|\mathbf{x}_{ij}\right) = \Phi_1^*(-\mathbf{x}_{ij}\boldsymbol{\beta}, \infty, \sigma_\xi^2)$$

Combined model

Here we present the likelihood for the probit model with continuous endogenous covariates, ordinal endogenous covariates, an ordinal endogenous treatment, and endogenous sample selection. This combines all the extensions to the standard probit model that are supported by eprobit. In *Likelihood for multiequation models*, we describe the general framework for ERMs with multiple features and show how random effects may be combined with other features, how xteprobit can support the other ERM features.

Deriving the combined model with tobit rather than probit endogenous sample selection is straightforward. On selected observations, the selection indicator would be treated like a continuous endogenous covariate. On nonselected observations, the model would be identical to the combined model with probit selection. The correlations between the outcome errors and other errors are also the same between treatment groups and levels of ordinal endogenous covariates. Deriving the model with different correlations for the treatment groups and endogenous covariate groups is straightforward. Take the likelihood given here in this section, and use a different covariance matrix depending on the levels of treatment and the ordinal endogenous covariates.

In this model, the treatment t_i takes T values, indexing the potential outcomes of the main outcome y_i: $y_{1i}, \ldots, y_{Ti}$. The relationship between the ordinal treatment t_i, treatment covariates $\mathbf{z}_{t,i}$, and error ϵ_{ti} is described in (7). For $j = 1, \ldots, B$, the relationship between the ordinal endogenous covariates w_{bji}, exogenous covariates $\mathbf{z}_{bji}$, and error ϵ_{bji} is given in (3). The model also uses the $\mathbf{wind}_{bji}$ terms that are defined in (4).

The probit regression of y_i on exogenous covariates $\mathbf{x}_i$, C continuous endogenous covariates $\mathbf{w}_{ci}$, and B ordinal endogenous covariates $\mathbf{w}_{bi} = [w_{b1i}, \ldots, w_{bBi}]$ with endogenous treatment t_i and endogenous sample selection on s_i has the form

$$y_{1i} = 1\left(\mathbf{x}_i\boldsymbol{\beta}_1 + \mathbf{w}_{ci}\boldsymbol{\beta}_{c1} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b11} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB1} + \epsilon_{1i} > 0\right)$$

$$\vdots$$

$$y_{Ti} = 1\left(\mathbf{x}_i\boldsymbol{\beta}_T + \mathbf{w}_{ci}\boldsymbol{\beta}_{cT} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1T} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bBT} + \epsilon_{Ti} > 0\right)$$

$$y_i = \sum_{j=1}^{T} 1(t_i = v_j)y_{ji}$$

$$\mathbf{w}_{ci} = \mathbf{z}_{ci}\mathbf{A}_c + \boldsymbol{\epsilon}_{ci}$$

$$s_i = 1\left(\mathbf{z}_{si}\boldsymbol{\alpha}_s + \epsilon_{si} > 0\right)$$

where $\mathbf{z}_{si}$ are covariates that affect selection and $\mathbf{z}_{ci}$ are covariates that affect the continuous endogenous covariates. The outcome y_i is observed if $s_i = 1$ and is not observed if $s_i = 0$.

For $j = 1, \ldots, T$, the unobserved errors $\epsilon_{ji}, \epsilon_{si}, \epsilon_{ti}, \epsilon_{b1i}, \ldots, \epsilon_{bBi}, \boldsymbol{\epsilon}_{ci}$ are multivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & \rho_{1s} & \rho_{1t} & \boldsymbol{\rho}'_{1b} & \boldsymbol{\sigma}'_{1c} \\ \rho_{1s} & 1 & \rho_{st} & \boldsymbol{\rho}'_{sb} & \boldsymbol{\sigma}'_{sc} \\ \rho_{1t} & \rho_{st} & 1 & \boldsymbol{\rho}'_{tb} & \boldsymbol{\sigma}'_{tc} \\ \boldsymbol{\rho}_{1b} & \boldsymbol{\rho}_{sb} & \boldsymbol{\rho}_{tb} & \boldsymbol{\Sigma}_b & \boldsymbol{\Sigma}'_{bc} \\ \boldsymbol{\sigma}_{1c} & \boldsymbol{\sigma}_{sc} & \boldsymbol{\sigma}_{tc} & \boldsymbol{\Sigma}_{bc} & \boldsymbol{\Sigma}_c \end{bmatrix}$$

As in *Continuous endogenous covariates*, we can write the joint density of the dependent variables as a product. We have

$$f(y_i, s_i, t_i, \mathbf{w}_{bi}, \mathbf{w}_{ci} | \mathbf{x}_i, \mathbf{z}_{si}, \mathbf{z}_{ti}, \mathbf{z}_{b1i}, \ldots, \mathbf{z}_{bBi}, \mathbf{z}_{ci}) =$$
$$f(y_i, s_i, t_i, \mathbf{w}_{bi} | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{si}, \mathbf{z}_{ti}, \mathbf{z}_{b1i}, \ldots, \mathbf{z}_{bBi}, \mathbf{z}_{ci}) f(\mathbf{w}_{ci} | \mathbf{z}_{ci})$$

We can then use the conditional distribution of $\epsilon_{ji}, \epsilon_{si}, \epsilon_{ti}, \epsilon_{b1i}, \ldots, \epsilon_{bBi}$ to obtain the conditional density of y_i, s_i, t_i, and $\mathbf{w}_{bi}$.

For $j = 1, \ldots, T$, conditional on $\mathbf{w}_{ci}$ and the exogenous covariates, ϵ_{ji} has mean

$$
\begin{aligned}
e_{1i} &= E\left(\epsilon_{ji} | \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{si}, \mathbf{z}_{ti}, \mathbf{z}_{b1i}, \ldots, \mathbf{z}_{bBi}, \mathbf{z}_{ci}\right) \\
&= \boldsymbol{\sigma}'_{1,c} \boldsymbol{\Sigma}_c^{-1} \left(\mathbf{w}_{ci} - \mathbf{z}_{c,i}\mathbf{A}_c\right)'
\end{aligned}
$$

Now, for $j = 1, \ldots, T$, let

$$
c_{1ij} = \begin{cases}
-\mathbf{x}_i\boldsymbol{\beta}_1 - \mathbf{w}_{ci}\boldsymbol{\beta}_{c,1} - \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b11} - \cdots - \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB1} - e_{1i} & j = 1 \\
\vdots & \\
-\mathbf{x}_i\boldsymbol{\beta}_T - \mathbf{w}_{ci}\boldsymbol{\beta}_{cT} - \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1T} - \cdots - \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bBT} - e_{1i} & j = T
\end{cases}
$$

The lower and upper limits for the y_i probability are

$$
l_{1i} = \begin{cases} -\infty & y_i = 0 \\ c_{1ij} & y_i = 1, t_i = v_j \end{cases} \qquad u_{1i} = \begin{cases} c_{1ij} & y_i = 0, t_i = v_j \\ \infty & y_i = 1 \end{cases}
$$

The conditional means of the unobserved errors $\epsilon_{si}, \epsilon_{ti}, \epsilon_{b1i}, \ldots, \epsilon_{bBi}$ have similar forms to e_{1i}. Denote these means by $e_{si}, e_{ti}, e_{b1i}, \ldots, e_{bBi}$. The lower and upper probability limits for s_i, t_i, and the ordinal endogenous covariates are obtained by subtracting the means from the limits defined in (10), (8), (9), (5), and (6).

$$
\begin{aligned}
l_{si}^* &= l_{si} - e_{si} \\
u_{si}^* &= u_{si} - e_{si} \\
l_{ti}^* &= l_{ti} - e_{ti} \\
u_{ti}^* &= u_{ti} - e_{ti} \\
l_{b1i}^* &= l_{b1i} - e_{b1i} \\
u_{b1i}^* &= u_{b1i} - e_{b1i} \\
&\vdots \\
l_{bBi}^* &= l_{bBi} - e_{bBi} \\
u_{bBi}^* &= u_{bBi} - e_{bBi}
\end{aligned}
$$

We have lower and upper limits; we need a conditional covariance and the conditional density of $\mathbf{w}_{ci}$ to formulate the likelihood. For $j = 1, \ldots, T$, conditional on $\mathbf{w}_{ci}$ and the exogenous covariates, $\epsilon_{ji}, \epsilon_{si}, \epsilon_{ti}, \epsilon_{b1i}, \ldots, \epsilon_{bBi}$ have covariance

$$
\boldsymbol{\Sigma}_{o|c} = \begin{bmatrix} 1 & \rho_{1s} & \rho_{1t} & \boldsymbol{\rho}'_{1b} \\ \rho_{1s} & 1 & \rho_{st} & \boldsymbol{\rho}'_{sb} \\ \rho_{1t} & \rho_{st} & 1 & \boldsymbol{\rho}'_{tb} \\ \boldsymbol{\rho}_{1b} & \boldsymbol{\rho}_{sb} & \boldsymbol{\rho}_{tb} & \boldsymbol{\Sigma}_b \end{bmatrix} - \begin{bmatrix} \boldsymbol{\sigma}'_{1c} \\ \boldsymbol{\sigma}'_{sc} \\ \boldsymbol{\sigma}'_{tc} \\ \boldsymbol{\Sigma}'_{bc} \end{bmatrix} \boldsymbol{\Sigma}_c^{-1} \begin{bmatrix} \boldsymbol{\sigma}'_{1c} \\ \boldsymbol{\sigma}'_{sc} \\ \boldsymbol{\sigma}'_{tc} \\ \boldsymbol{\Sigma}'_{bc} \end{bmatrix}'
$$

The conditional density of $\mathbf{w}_{ci}$ is

$$
f(\mathbf{w}_{ci} | \mathbf{z}_{ci}) = \phi_C(\mathbf{w}_{ci} - \mathbf{z}_{ci}\mathbf{A}_c, \boldsymbol{\Sigma}_c)
$$

Let

$$
\mathbf{l}_{1i} = \begin{bmatrix} l_{1i} & l_{si}^* & l_{ti}^* & l_{b1i}^* & \ldots & l_{bBi}^* \end{bmatrix}
$$
$$
\mathbf{u}_{1i} = \begin{bmatrix} u_{1i} & u_{si}^* & u_{ti}^* & u_{b1i}^* & \ldots & u_{bBi}^* \end{bmatrix}
$$
$$
\mathbf{l}_{i} = \begin{bmatrix} l_{si}^* & l_{ti}^* & l_{b1i}^* & \ldots & l_{bBi}^* \end{bmatrix}
$$
$$
\mathbf{u}_{i} = \begin{bmatrix} u_{si}^* & u_{ti}^* & u_{b1i}^* & \ldots & u_{bBi}^* \end{bmatrix}
$$

The log likelihood of the model is

$$
\begin{aligned}
\ln L = \sum_{i \in S} w_i \ln \Phi_{3+B}^* \left(\mathbf{l}_{1i}, \mathbf{u}_{1i}, \boldsymbol{\Sigma}_{o|c} \right) + \\
\sum_{i \notin S} w_i \ln \Phi_{2+B}^* \left(\mathbf{l}_{i}, \mathbf{u}_{i}, \boldsymbol{\Sigma}_{o|c,-1} \right) + \\
\sum_{i=1}^{N} w_i \ln \phi_C (\mathbf{w}_{ci} - \mathbf{z}_{ci} \mathbf{A}_c, \boldsymbol{\Sigma}_c)
\end{aligned}
$$

where S is the set of observations where y_i is observed and $\boldsymbol{\Sigma}_{o|c,-1}$ is $\boldsymbol{\Sigma}_{o|c}$ with the first row and column removed.

As in previous sections, we use the joint and marginal probabilities to determine conditional probabilities.

For $j = 1, \ldots, T$ and i such that $t_i = v_j$, let

$$
\mathbf{l}_{i11} = \begin{bmatrix} c_{1ij} & l_{ti}^* & l_{b1i}^* & \ldots & l_{bBi}^* \end{bmatrix}
$$
$$
\mathbf{u}_{i11} = \begin{bmatrix} \infty & u_{ti}^* & u_{b1i}^* & \ldots & u_{bBi}^* \end{bmatrix}
$$
$$
\mathbf{l}_{i12} = \begin{bmatrix} l_{ti}^* & l_{b1i}^* & \ldots & l_{bBi}^* \end{bmatrix}
$$
$$
\mathbf{u}_{i12} = \begin{bmatrix} u_{ti}^* & u_{b1i}^* & \ldots & u_{bBi}^* \end{bmatrix}
$$

Let $\boldsymbol{\Sigma}_{o|c,-s}$ be $\boldsymbol{\Sigma}_{o|c}$ with the second row and column removed. This is the conditional covariance matrix without the endogenous sample-selection equation components. Let $\boldsymbol{\Sigma}_{o|c,-s-1}$ be $\boldsymbol{\Sigma}_{o|c,-s}$ with the first row and column removed.

The conditional probability of success at treatment level $t_i = v_j$ is

$$
\Pr(y_i = 1 | \mathbf{t}_i = v_j, \mathbf{w}_{bi}, \mathbf{w}_{ci}, \mathbf{x}_i, \mathbf{z}_{si}, \mathbf{z}_{ti}, \mathbf{z}_{b1i}, \ldots, \mathbf{z}_{bBi}, \mathbf{z}_{ci}) = \frac{\Phi_{2+B}^* \left(\mathbf{l}_{i11}, \mathbf{u}_{i11}, \boldsymbol{\Sigma}_{o|c,-s} \right)}{\Phi_{1+B}^* \left(\mathbf{l}_{i12}, \mathbf{u}_{i12}, \boldsymbol{\Sigma}_{o|c,-s-1} \right)}
$$

The conditional probabilities of treatment, selection, and the ordinal endogenous covariates are derived in similar ways. We condition on the treatment and the other endogenous covariates together with the exogenous covariates that affect the outcome. POMs and treatment effects are conditioned on the endogenous and exogenous covariates. See *Predictions using the full model* in [ERM] **eprobit postestimation** for more details.

Confidence intervals

The estimated variances will always be nonnegative, and the estimated correlations will always fall in $(-1, 1)$. We use transformations to obtain confidence intervals that accommodate these ranges.

We use the log transformation to obtain the confidence intervals for variance parameters. Let $\widehat{\sigma}^2$ be a point estimate for the variance parameter σ^2, and let $\widehat{\mathrm{SE}}(\widehat{\sigma}^2)$ be its standard error. The $(1 - \alpha) \times 100\%$ confidence interval for $\ln(\sigma^2)$ is

$$\ln(\widehat{\sigma}^2) \pm z_{\alpha/2} \frac{\widehat{\mathrm{SE}}(\widehat{\sigma}^2)}{\widehat{\sigma}^2}$$

where $z_{\alpha/2}$ is the $1 - \alpha/2$ quantile of the standard normal distribution. Let k_u be the upper endpoint of this interval, and let k_l be the lower. The $(1 - \alpha) \times 100\%$ confidence interval for σ^2 is then given by

$$\left(e^{k_l}, e^{k_u} \right)$$

We use the inverse hyperbolic tangent transformation to obtain confidence intervals for correlation parameters; for details on the hyperbolic functions, see [FN] **Trigonometric functions**. Let $\widehat{\rho}$ be a point estimate for the correlation parameter ρ, and let $\widehat{\mathrm{SE}}(\widehat{\rho})$ be its standard error. The $(1 - \alpha) \times 100\%$ confidence interval for $\mathtt{atanh}(\rho)$ is

$$\mathtt{atanh}(\widehat{\rho}) \pm z_{\alpha/2} \widehat{\mathrm{SE}}(\widehat{\rho}) \frac{1}{1 - \widehat{\rho}^2}$$

where $z_{\alpha/2}$ is the $1 - \alpha/2$ quantile of the standard normal distribution. Let k_u be the upper endpoint of this interval, and let k_l be the lower. The $(1 - \alpha) \times 100\%$ confidence interval for ρ is then given by

$$\{\mathtt{tanh}(k_l), \mathtt{tanh}(k_u)\}$$

Likelihood for multiequation models

The general framework for ERMs is formulated such that it accommodates multiple features. Binary and ordinal endogenous covariates may occur together with continuous endogenous covariates in ERMs. Endogenous covariates may also occur together with endogenous sample selection or treatments in ERMs. Random effects may occur in any combination with the other features as well.

Here we show how the log likelihood is formulated when we have multiple auxiliary equations. We begin with the cross-sectional case, where there are no random effects.

Suppose that we have H auxiliary equations with endogenous outcomes $y_{1i}, \ldots, y_{Hi}$. We will treat the main outcome y_i as stage $J = H + 1$, so $y_{Ji} = y_i$. The ERMs that we fit with `eintreg`, `eoprobit`, `eprobit`, and `eregress` are triangular, so we can order the equations such that the first depends only on exogenous covariates—say, $\mathbf{w}_{1i} = \mathbf{z}_i$—and for $j = 2, \ldots, J$, equation j depends only on the exogenous covariates $\mathbf{z}_i$ and the endogenous covariates from equation $h = j - 1$ and $y_{1i}, \ldots, y_{hi}$ below. These are stored together in $\mathbf{w}_{ji}$.

So we have

$$y_{1i} = g_{1i}(\mathbf{w}_{1i}\boldsymbol{\beta}_1 + v_{1i})$$

$$\vdots$$

$$y_{Hi} = g_{Hi}(\mathbf{w}_{Hi}\boldsymbol{\beta}_H + v_{Hi})$$

$$y_i = y_{Ji} = g_{Ji}(\mathbf{w}_{Ji}\boldsymbol{\beta}_J + v_{Ji})$$

where the form of the functions $g_{ji}(\cdot)$ is determined by whether the outcome y_{ji} has a linear, probit, or interval model. The errors $v_{1i}, \ldots, v_{Ji}$ are multivariate normal with mean 0 and covariance $\boldsymbol{\Sigma}$.

The covariates $\mathbf{w}_{ji}$ and the outcome y_{ji} determine a range for the error v_{ji}. For example, if y_{ji} has a linear model, then $v_{ji} = y_{ji} - \mathbf{w}_{ji}\boldsymbol{\beta}_j$, the residual. If $y_{ji} = 1$ and y_{ji} has a probit model, then v_{ji} is in the range $(-\mathbf{w}_{ji}\boldsymbol{\beta}_j, \infty)$. If y_{ij} is left-censored at l_i, then v_{ji} is in the range $(-\infty, l_i - \mathbf{w}_{ji}\boldsymbol{\beta}_j)$.

The density of the endogenous variables can be represented using a multivariate normal density function that is evaluated at the residuals for the continuous outcomes and integrated over the error ranges of the noncontinuous outcomes.

The conditional density of the error v_{ji} on $\mathbf{w}_{ji}$ has the form

$$f(v_{ji}|\mathbf{w}_{ji}) = \frac{\int_{\mathbf{V}_{hi}^\star} \phi_j(v_{1i}, \ldots, v_{ji}, \boldsymbol{\Sigma}_j) d\mathbf{v}_{hi}^\star}{\int_{\mathbf{V}_{hi}^\star} \phi_h(v_{1i}, \ldots, v_{hi}, \boldsymbol{\Sigma}_h) d\mathbf{v}_{hi}^\star}$$

where $\boldsymbol{\Sigma}_j$ is the covariance of $v_{1i}, \ldots, v_{ji}$ and $\boldsymbol{\Sigma}_h$ is the covariance of $v_{1i}, \ldots, v_{hi}$, where $h = j - 1$. The vector $\mathbf{v}_{hi}^\star$ contains the errors that correspond to binary, ordinal, or censored outcomes in y_{1i}, $\ldots, y_{hi}$. These outcomes induce the error ranges $\mathbf{V}_{hi}^\star$, which we integrate over. The other errors are determined by the outcomes and covariates as residuals.

If y_{ji} is continuous, then

$$f(y_{ji}|\mathbf{w}_{ji}) = f(v_{ji}|\mathbf{w}_{ji}) \tag{11}$$

When y_{ji} is a binary, ordinal, or censored outcome, we have

$$f(y_{ji}|\mathbf{w}_{ji}) = \frac{\int_{\mathbf{V}_{ji}^\star} \phi_j(v_{1i}, \ldots, v_{ji}, \boldsymbol{\Sigma}_j) d\mathbf{v}_{ji}^\star}{\int_{\mathbf{V}_{hi}^\star} \phi_h(v_{1i}, \ldots, v_{hi}, \boldsymbol{\Sigma}_h) d\mathbf{v}_{hi}^\star} \tag{12}$$

So we also integrate over the range of the error v_{ji} when y_{ji} is not continuous.

We can express the joint density of the main outcome and the endogenous covariates in terms of the marginal and conditional densities. The denominator in (11) or (12) in the higher stage will cancel out the numerator of (11) or (12) in the lower stage, so we have

$$f(y_{1i}, \ldots, y_{ji}|\mathbf{z}_i) = \int_{\mathbf{V}_{ji}^\star} \phi_j(v_{1i}, \ldots, v_{ji}, \boldsymbol{\Sigma}_j) d\mathbf{v}_{ji}^\star \tag{13}$$

If we have only continuous endogenous variables, we have

$$f(y_{1i}, \ldots, y_{ji}|\mathbf{z}_i) = \phi_j(v_{1i}, \ldots, v_{ji}, \boldsymbol{\Sigma}_j)$$

If $\mathbf{V}_{ji}^\star$ has dimension j, we can calculate the integral given in (13) by using the Φ_j^*. Let $\mathbf{l}_i$ contain the lower endpoints and $\mathbf{u}_i$ contain the upper endpoints for $\mathbf{V}_{ji}^\star$. When we do not have continuous endogenous covariates, we have

$$f(y_{1i}, \ldots, y_{ji}|\mathbf{z}_i) = \Phi_j^*(\mathbf{l}_i, \mathbf{u}_i, \boldsymbol{\Sigma}_j)$$

Now suppose that we have $C < j$ continuous outcomes in $y_{1i}, \ldots, y_{ji}$, so the dimension of $\mathbf{V}_{ji}^{\star}$ is $j - C$. Without loss of generality, these C correspond to the last C endogenous covariates $y_{(j-C+1)i}, \ldots, y_{ji}$. The covariates can be reordered as needed.

We partition the covariance

$$\mathbf{\Sigma}_j = \begin{bmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12}' \\ \mathbf{\Sigma}_{12} & \mathbf{\Sigma}_{22} \end{bmatrix}$$

where $\mathbf{\Sigma}_{22}$ is the covariance of the last C errors.

Conditional on $v_{(j-C+1)i}, \ldots, v_{ji}$, the errors $v_{1i}, \ldots, v_{(j-C)i}$ have mean and variance

$$\boldsymbol{\mu}_{1|2,i} = \mathbf{\Sigma}_{12}\mathbf{\Sigma}_{22}^{-1} \begin{bmatrix} v_{(j-C+1)i} \\ \vdots \\ v_{ji} \end{bmatrix}$$

$$\mathbf{\Sigma}_{1|2} = \mathbf{\Sigma}_{11} - \mathbf{\Sigma}_{12}\mathbf{\Sigma}_{22}^{-1}\mathbf{\Sigma}_{12}'$$

By conditioning on $v_{(j-C+1)i}, \ldots, v_{ji}$, we can express the density in terms of ϕ_C and Φ_{j-C}^*. We can write the joint density in terms of the marginal and conditional densities to obtain

$$f(y_{1i}, \ldots, y_{ji}|\mathbf{z}_i) = \phi_C(v_{(j-C+1)i}, \ldots, v_{ji}, \mathbf{\Sigma}_{22})\Phi_{j-C}^*(\mathbf{l}_i - \boldsymbol{\mu}_{1|2,i}, \mathbf{u}_i - \boldsymbol{\mu}_{1|2,i}, \mathbf{\Sigma}_{1|2})$$

The natural logarithm of the density $f(y_{1i}, \ldots, y_{Ji}|\mathbf{z}_i)$ is the log likelihood of the model. We maximize the log likelihood to estimate the model parameters.

We can relax the assumption that the errors $v_{1i}, \ldots, v_{Ji}$ are multivariate normal with mean 0 and covariance $\mathbf{\Sigma}$. We will allow the covariance matrix to vary based on the M different levels of the binary or ordinal endogenous covariates $\mathbf{w}_{poi}$: $\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_M$. These are the different combinations of values for the covariates $\mathbf{w}_{poi}$.

We use a potential-outcome framework for the outcome errors v_{Ji}. For the potential-outcome errors $v_{1Ji}, \ldots, v_{MJi}$, we have

$$v_{Ji} = \sum_{m=1}^{M} 1(\mathbf{w}_{poi} = \boldsymbol{\omega}_m)v_{mJi}$$

For $m = 1, \ldots, M$, v_{mJi} and $v_{1i}, \ldots, v_{Hi}$ are multivariate normal mean 0 and covariance

$$\mathbf{\Sigma}_m = \begin{bmatrix} \sigma_m^2 & \boldsymbol{\sigma}_{mo}' \\ \boldsymbol{\sigma}_{mo} & \mathbf{\Sigma}_o \end{bmatrix}$$

For observations where $\mathbf{w}_{poi} = \boldsymbol{\omega}_m$, the log likelihood can be derived with $\mathbf{\Sigma}_m$ in place of $\mathbf{\Sigma}$. The log likelihoods from the different potential-outcome group observations can then be summed together to get the log likelihood of the model.

Now we assume that we have random effects in each equation and a panel-data structure. This discussion applies to the models fit by xteintreg, xteoprobit, xteprobit, and xteregress. For simplicity, we assume that the errors do not follow a potential-outcome framework. We have N panels. For panel $i = 1, \ldots, N$, there are N_i observations, and for $t = 1, \ldots, N_i$, we have

$$y_{1it} = g_{1it}(\mathbf{w}_{1it}\boldsymbol{\beta}_1 + v_{1it} + u_{1i})$$

$$\vdots$$

$$y_{Hit} = g_{Hit}(\mathbf{w}_{Hit}\boldsymbol{\beta}_H + v_{Hit} + u_{Hi})$$

$$y_{it} = y_{Jit} = g_{Jit}(\mathbf{w}_{Jit}\boldsymbol{\beta}_J + v_{Jit} + u_{Ji})$$

The observation-level errors $v_{1it}, \ldots, v_{Jit}$ are multivariate normal with mean 0 and covariance $\boldsymbol{\Sigma}$. They are independent of the panel-level errors, or random effects $u_{1i}, \ldots, u_{Ji}$, which are multivariate normal with mean 0 and covariance $\boldsymbol{\Sigma}_u$. We further assume that the observation-level errors are independent within panels.

Now the covariates $\mathbf{w}_{jit}$, random effect u_{ji}, and the outcome y_{jit} determine a range for the error v_{jit}. For example, if y_{jit} has a linear model, then $v_{jit} = y_{jit} - \mathbf{w}_{jit}\boldsymbol{\beta}_j - u_{ji}$, the residual. If $y_{jit} = 1$ and y_{jit} has a probit model, then v_{jit} is in the range $(-\mathbf{w}_{jit}\boldsymbol{\beta}_j - u_{ji}, \infty)$. If y_{ijt} is left-censored at l_{it}, then v_{jit} is in the range $(-\infty, l_i - \mathbf{w}_{jit}\boldsymbol{\beta}_j - u_{ji})$.

Conditional on the random effects $u_{1i}, \ldots, u_{Ji}$, the density of the endogenous variables can be represented using a multivariate normal density function that is evaluated at the residuals for the continuous outcomes and integrated over the error ranges of the noncontinuous outcomes. So the conditional density is formulated as in the cross-sectional case. The random effects are essentially added to the covariates $\mathbf{w}_{1it}, \ldots, \mathbf{w}_{Jit}$.

Note that each panel has the same random effects for every observation. So if panel i has random effects $\mathbf{u}_i = (u_{1i}, \ldots, u_{Ji})$, its likelihood is

$$L_i = \int_{\Re^J} \left\{ \prod_{t=1}^{N_i} f(y_{1it}, \ldots, y_{Jit}|\mathbf{z}_{it}, \mathbf{u_i}) \right\} \phi_J(\mathbf{u}_i, \boldsymbol{\Sigma}_u)d\mathbf{u_i} \tag{14}$$

This multivariate integral is generally not tractable. We can use a change-of-variables technique to transform the multivariate integral in (14) into a set of nested univariate integrals. Let $\mathbf{L}$ be the Cholesky decomposition of $\boldsymbol{\Sigma}_u$; that is, $\boldsymbol{\Sigma}_u = \mathbf{L}\mathbf{L}'$. It follows that $\mathbf{u}_i = \mathbf{L}\boldsymbol{\psi}_i$, where $\boldsymbol{\psi}_i$ is a vector of independent standard normal random variables.

So we can rewrite (14) as

$$L_i = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \left\{ \prod_{t=1}^{N_i} f(y_{1it}, \ldots, y_{Jit}|\mathbf{z}_{it}, \mathbf{u_i} = \mathbf{L}\boldsymbol{\psi}_i) \right\} \phi(\psi_{1i}) \ldots \phi(\psi_{Ji})d\psi_{1i} \ldots d\psi_{Ji} \tag{15}$$

Now the univariate integral can be approximated using Gauss–Hermite quadrature (GHQ). For q-point GHQ, let the abscissa and weight pairs be denoted by (a_k^*, w_k^*), $k = 1, \ldots, q$. Then, the GHQ approximation is

$$\int_{-\infty}^{\infty} f(x) \exp(-x^2) \, dx \approx \sum_{k=1}^{q} w_k^* f(a_k^*)$$

Consider a J-dimensional quadrature grid containing q quadrature points in each dimension. Let the vector of abscissas $\mathbf{a}_k = (a_{k_1}, \ldots, a_{k_J})'$ be a point in this grid, and let $\mathbf{w}_k = (w_{k_1}, \ldots, w_{k_J})'$ be the vector of corresponding weights. The GHQ approximation to the likelihood for a given panel is

$$
L_i = \sum_{k_1=1}^{q} \cdots \sum_{k_J=1}^{q} \left[\left\{ \prod_{t=1}^{N_i} f(y_{1it}, \ldots, y_{Jit} | \mathbf{z}_{it}, \mathbf{u_i} = \mathbf{La}_k) \right\} \left\{ \prod_{s=1}^{J} w_{k_s} \right\} \right] \tag{16}
$$

Rather than using regular GHQ, we can use mean–variance adaptive Gauss–Hermite quadrature. Fixing the observed variables and model parameters in the integrand of (14), we see the posterior density for ψ_i is proportional to

$$
\left\{ \prod_{t=1}^{N_i} f(y_{1it}, \ldots, y_{Jit} | \mathbf{z}_{it}, \mathbf{u_i} = \mathbf{L}\psi_i) \right\} \phi(\psi_i)
$$

It is reasonable to assume that this posterior density can be approximated by a multivariate normal density with mean vector $\boldsymbol{\mu}_{vi}$ and variance matrix $\boldsymbol{\tau}_{vi}$. Instead of using the prior density of ψ_i as the weighting distribution in the integral, we can use our approximation for the posterior density,

$$
L_i = \int_{\Re^J} \frac{\left\{ \prod_{t=1}^{N_i} f(y_{1it}, \ldots, y_{Jit} | \mathbf{z}_{it}, \mathbf{u_i} = \mathbf{L}\psi_i) \right\} \phi(\psi_i)}{\phi(\psi_i, \boldsymbol{\mu}_{vi}, \boldsymbol{\tau}_{vi})} \phi(\psi_i, \boldsymbol{\mu}_{vi}, \boldsymbol{\tau}_{vi}) \, d\psi_i
$$

The likelihood is then approximated by

$$
L_i = \sum_{k_1=1}^{q} \cdots \sum_{k_J=1}^{q} \left[\left\{ \prod_{t=1}^{N_i} f(y_{1it}, \ldots, y_{Jit} | \mathbf{z}_{it}, \mathbf{u_i} = \mathbf{L}\boldsymbol{\alpha}_k) \right\} \left\{ \prod_{s=1}^{J} \omega_{k_s} \right\} \right] \tag{17}
$$

where $\boldsymbol{\alpha}_k$ and ω_{k_s} are the adaptive versions of the abscissas and weights after an orthogonalizing transformation, which eliminates posterior covariances between elements of ψ_i. The posterior means $\boldsymbol{\mu}_{vi}$ and posterior variances $\boldsymbol{\tau}_{vi}$ are computed iteratively by updating the posterior moments by using the mean–variance adaptive Gauss–Hermite approximation, starting with a 0 mean vector and identity variance matrix.

Then, the log likelihood for all panels is

$$
\ln L = \sum_{i=1}^{N} \left(\ln \sum_{k_1=1}^{q} \cdots \sum_{k_J=1}^{q} \left[\left\{ \prod_{t=1}^{N_i} f(y_{1it}, \ldots, y_{Jit} | \mathbf{z}_{it}, \mathbf{u_i} = \mathbf{L}\boldsymbol{\alpha}_k) \right\} \left\{ \prod_{s=1}^{J} \omega_{k_s} \right\} \right] \right) \tag{18}
$$

As in the cross-sectional case, we can relax the assumption that the errors $v_{1it}, \ldots, v_{Jit}$ are multivariate normal with mean 0 and covariance $\boldsymbol{\Sigma}$. We will allow the covariance matrix to vary based on the M different levels of the binary or ordinal endogenous covariates $\mathbf{w}_{poit}$: $\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_M$. These are the different combinations of values for the covariates $\mathbf{w}_{poit}$.

We use a potential-outcome framework for the outcome errors v_{Jit}. For the potential-outcome errors $v_{1Jit}, \ldots, v_{MJit}$, we have

$$
v_{Jit} = \sum_{m=1}^{M} 1(\mathbf{w}_{poit} = \boldsymbol{\omega}_m) v_{mJit}
$$

For $m = 1, \ldots, M$, v_{mJit} and v_{1it}, ..., v_{Hit} are multivariate normal mean 0 and covariance

$$\mathbf{\Sigma}_m = \begin{bmatrix} \sigma_m^2 & \boldsymbol{\sigma}_{mo}' \\ \boldsymbol{\sigma}_{mo} & \mathbf{\Sigma}_o \end{bmatrix}$$

For observations where $\mathbf{w}_{poit} = \boldsymbol{\omega}_m$, the likelihood can be derived with $\mathbf{\Sigma}_m$ in place of Σ.

References

Amemiya, T. 1985. *Advanced Econometrics*. Cambridge, MA: Harvard University Press.

Angrist, J. D. 2001. Estimation of limited dependent variable models with dummy endogenous regressors: Simple strategies for empirical practice. *Journal of Business and Economic Statistics* 19: 2–28. https://doi.org/10.1198/07350010152472571.

Arendt, J. N., and A. Holm. 2006. Probit models with binary endogenous regressors. Discussion Papers on Business and Economics No. 4/2006, Department of Business and Economics, University of Southern Denmark, Odense, Demark. https://www.sdu.dk/da/om_sdu/institutter_centre/ivoe_virksomhedsledelse_og_oekonomi/forskning/forskningspublikationer/~/media/Files/Om_SDU/Institutter/Ivoe/Disc_papers/Disc_2006/dpbe4%202006%20pdf.ashx.

Bartus, T., and D. Roodman. 2014. Estimation of multiprocess survival models with cmp. *Stata Journal* 14: 756–777.

Conway, M. R. 1990. A random effects model for binary data. *Biometrics* 46: 317–328. https://doi.org/10.2307/2531437.

Drezner, Z. 1994. Computation of the trivariate normal integral. *Mathematics of Computation* 62: 289–294. http://doi.org/10.2307/2153409.

Hasebe, T. 2022. Endogenous models of binary choice outcomes: Copula-based maximum-likelihood estimation and treatment effects. *Stata Journal* 22: 734–771.

Heckman, J. J. 1976. The common structure of statistical models of truncation, sample selection and limited dependent variables and a simple estimator for such models. *Annals of Economic and Social Measurement* 5: 475–492.

———. 1979. Sample selection bias as a specification error. *Econometrica* 47: 153–161. https://doi.org/10.2307/1912352.

Karymshakov, K., K. Sultakeev, and B. Sulaimanova. 2015. The impact of microfinance on entrepreneurship in Kyrgyzstan. *International Conference on Eurasian Economies*, paper ID 1412. Eurasian Economists Association: Kazan, Russia. http://www.avekon.org/papers/1412.pdf.

Maddala, G. S. 1983. *Limited-Dependent and Qualitative Variables in Econometrics*. Cambridge: Cambridge University Press.

Maddala, G. S., and L.-F. Lee. 1976. Recursive models with qualitative endogenous variables. *Annals of Economic and Social Measurement* 5: 525–545.

Miwa, T., A. J. Hayter, and S. Kuriki. 2003. The evaluation of general non-centred orthant probabilities. *Journal of the Royal Statistical Society, Series B* 65: 223–234. https://doi.org/10.1111/1467-9868.00382.

Miwa, T., A. J. Hayter, and W. Liu. 2000. Calculations of level probabilities for normal random variables with unequal variances with applications to Bartholomew's test in unbalanced one-way models. *Computational Statistics and Data Analysis* 34: 17–32. https://doi.org/10.1016/S0167-9473(99)00075-4.

Mulkay, B. 2015. Bivariate probit estimation for panel data: A two-step Gauss–Hermite quadrature approach with an application to product and process innovations for France. Working paper, Université de Montpellier, Faculté d'Economie, Montepellier, France. https://afse2015.sciencesconf.org/61055/document.

Naylor, J. C., and A. F. M. Smith. 1982. Applications of a method for the efficient computation of posterior distributions. *Journal of the Royal Statistical Society, Series C* 31: 214–225. https://doi.org/10.2307/2347995.

Newey, W. K. 1987. Efficient estimation of limited dependent variable models with endogenous explanatory variables. *Journal of Econometrics* 36: 231–250. https://doi.org/10.1016/0304-4076(87)90001-7.

Pindyck, R. S., and D. L. Rubinfeld. 1998. *Econometric Models and Economic Forecasts*. 4th ed. New York: McGraw–Hill.

Roodman, D. 2011. Fitting fully observed recursive mixed-process models with cmp. *Stata Journal* 11: 159–206.

Semykina, A., and J. M. Wooldridge. 2018. Binary response panel data models with sample selection and self-selection. *Journal of Applied Econometrics* 33: 179–197. https://doi.org/10.1002/jae.2592.

Skrondal, A., and S. Rabe-Hesketh. 2004. *Generalized Latent Variable Modeling: Multilevel, Longitudinal, and Structural Equation Models*. Boca Raton, FL: Chapman and Hall/CRC.

Vahter, P. 2011. Does FDI spur productivity, knowledge sourcing and innovation by incumbent firms? Evidence from manufacturing industry in Estonia. *World Economy* 34: 1308–1326. http://doi.org/10.1111/j.1467-9701.2011.01379.x.

Van de Ven, W. P. M. M., and B. M. S. Van Pragg. 1981. The demand for deductibles in private health insurance: A probit model with sample selection. *Journal of Econometrics* 17: 229–252. https://doi.org/10.1016/0304-4076(81)90028-2.

White, H. L., Jr. 1996. *Estimation, Inference and Specification Analysis*. Cambridge: Cambridge University Press.

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

———. 2014. Quasi-maximum likelihood estimation and testing for nonlinear models with endogenous explanatory variables. *Journal of Econometrics* 182: 226–234. https://doi.org/10.1016/j.jeconom.2014.04.020.

Also see

Title

> **eprobit postestimation** — Postestimation tools for eprobit and xteprobit

Postestimation commands   predict   margins   Remarks and examples
Methods and formulas   References   Also see

Postestimation commands

The following postestimation command is of special interest after `eprobit` and `xteprobit`:

Command	Description
estat teffects	treatment effects and potential-outcome means

The following standard postestimation commands are also available after `eprobit` and `xteprobit`:

Command	Description
contrast	contrasts and ANOVA-style joint tests of estimates
estat ic	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC)
estat summarize	summary statistics for the estimation sample
estat vce	variance–covariance matrix of the estimators (VCE)
† estat (svy)	postestimation statistics for survey data
estimates	cataloging estimation results
etable	table of estimation results
* forecast	dynamic forecasts and simulations
* hausman	Hausman's specification test
lincom	point estimates, standard errors, testing, and inference for linear combinations of coefficients
* lrtest	likelihood-ratio test
margins	marginal means, predictive margins, marginal effects, and average marginal effects
marginsplot	graph the results from margins (profile plots, interaction plots, etc.)
nlcom	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
predict	means, probabilities, treatment effects, etc.
predictnl	point estimates, standard errors, testing, and inference for generalized predictions
pwcompare	pairwise comparisons of estimates
† suest	seemingly unrelated estimation
test	Wald tests of simple and composite linear hypotheses
testnl	Wald tests of nonlinear hypotheses

*`forecast`, `hausman`, and `lrtest` are not appropriate with `svy` estimation results.

†`suest` and the survey data `estat` commands are not available after `xteprobit`.

predict

Predictions after `eprobit` and `xteprobit` are described in

[ERM] **eprobit predict**	predict after eprobit and xteprobit
[ERM] **predict treatment**	predict for treatment statistics
[ERM] **predict advanced**	predict's advanced features

[ERM] **eprobit predict** describes the most commonly used predictions. If you fit a model with treatment effects, predictions specifically related to these models are detailed in [ERM] **predict treatment**. [ERM] **predict advanced** describes less commonly used predictions, such as predictions of outcomes in auxiliary equations.

margins

Description for margins

`margins` estimates statistics based on fitted models. These statistics include marginal means, marginal probabilities, potential-outcome means, average and conditional derivatives, average and conditional effects, and treatment effects.

Menu for margins

Statistics > Postestimation

Syntax for margins

> margins [*marginlist*] [, *options*]
>
> margins [*marginlist*] , <u>pred</u>ict(*statistic* ...) [<u>pred</u>ict(*statistic* ...) ...] [*options*]

statistic	Description
Main	
pr	probability for binary or ordinal y_j; the default
<u>me</u>an	mean
<u>pom</u>ean	potential-outcome mean
te	treatment effect
tet	treatment effect on the treated
xb	linear prediction excluding all complications
pr(*a,b*)	$\Pr(a < y_j < b)$ for continuous y_j
e(*a,b*)	$E(y_j \mid a < y_j < b)$ for continuous y_j
<u>y</u>star(*a,b*)	$E(y_j^*)$, $y_j^* = \max\{a, \min(y_j, b)\}$ for continuous y_j

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

For the full syntax, see [R] **margins**.

Remarks and examples

See [ERM] **Intro 7** for an overview of using `margins` and `predict` after `eprobit` and `xteprobit`. For examples using `margins`, `predict`, and `estat teffects`, see *Interpreting effects* in [ERM] **Intro 9** and see [ERM] **Example 1a**.

Methods and formulas

These methods build on the discussions in *Methods and formulas* of [ERM] **eprobit**.

Methods and formulas are presented under the following headings:

> *Predictions and inferences using the default asf*
> *General prediction framework*

Predictions and inferences using the default asf

In the probit model, for exogenous covariates $\mathbf{x}_i$ and endogenous covariates $\mathbf{w}_i$, we have

$$y_i = \mathbf{1}(\mathbf{x_i}\boldsymbol{\beta} + \mathbf{w_i}\boldsymbol{\beta_2} + \epsilon_\mathbf{i} > \mathbf{0})$$

where ϵ_i is a standard normal error.

Because ϵ_i is a normally distributed, mean 0, random variable, we can split it into two mean 0, normally distributed, independent parts,

$$\epsilon_i = u_i + \psi_i$$

where $u_i = \boldsymbol{\gamma}\epsilon_{2i}$ is the unobserved heterogeneity that gives rise to the endogeneity and ψ_i is an idiosyncratic error term with variance σ_ψ^2. Conditional on the covariates and the unobserved heterogeneity, for one endogenous covariate, the probability that $y_i = 1$ is

$$\Pr(y_i = 1 | \mathbf{x}_i, w_i, u_i) = \Phi\left(\frac{\mathbf{x}_i\boldsymbol{\beta} + \mathbf{w}_i\boldsymbol{\beta}_2 + u_i}{\sigma_\psi}\right)$$

Default predictions and effects are computed based on the expression above. Including u_i controls for endogeneity. Thus, all effects computed using the expression above have a structural interpretation. See Imbens and Newey (2009) and Wooldridge (2010) for a detailed description of structural functions for models with endogeneity.

Our discussion easily extends to models for panel data with random effects. In this case, we have N panels. Panel $i = 1, \dots, N$ has observations $t = 1, \dots, N_i$, so we observe y_{it} with random effect α_i and observation-level error ϵ_{it}. These errors are independent of each other. So the combined error $\xi_{it} = \alpha_i + \epsilon_{it}$ is normal with mean 0 and variance $1 + \sigma_\alpha^2$, where σ_α^2 is the variance of α_i. The results discussed earlier can then be applied using the combined error ξ_{it} rather than the cross-sectional error.

All predictions after `xteprobit` assume the panel-level random effects (α_i) are zero. Put another way, predictions condition on the random effects being set to their means.

General prediction framework

In this section, we discuss the general framework for predictions made after ERMs with multiple auxiliary equations and conditioned on both the covariates and the instruments. The predictions consider the total effect of all the covariates and instruments on the outcome.

First, assume that we have a model with random effects in each equation and a panel-data structure. We have N panels. For panel $i = 1, \ldots, N$, there are N_i observations, and for $t = 1, \ldots, N_i$, we have

$$y_{1it} = g_{1it}(\mathbf{w}_{1it}\boldsymbol{\beta}_1 + v_{1it} + u_{1i})$$

$$\vdots$$

$$y_{Hit} = g_{Hit}(\mathbf{w}_{Hit}\boldsymbol{\beta}_H + v_{Hit} + u_{Hi})$$

$$y_{it} = y_{Jit} = g_{Jit}(\mathbf{w}_{Jit}\boldsymbol{\beta}_J + v_{Jit} + u_{Ji})$$

The observation-level errors $v_{1it}, \ldots, v_{Jit}$ are multivariate normal with mean 0 and covariance $\boldsymbol{\Sigma}$. They are independent of the panel-level errors, or random effects $u_{1i}, \ldots, u_{Ji}$, which are multivariate normal with mean 0 and covariance $\boldsymbol{\Sigma}_u$. We further assume that the observation-level errors are independent within panels.

We will perform prediction conditional on the observed covariates, so we can collapse the random effects and observation-level errors. The new observation-level errors are $\xi_{jit} = v_{jit} + u_{ji}$. These errors, $\xi_{1it}, \ldots, \xi_{Jit}$, are multivariate normal with mean 0 and variance $\boldsymbol{\Sigma}_\xi = \boldsymbol{\Sigma} + \boldsymbol{\Sigma}_u$.

In the following, we will derive prediction formulas for the cross-sectional case without a panel structure, but our results will apply to the random-effects model we have just discussed, using the combined covariance $\boldsymbol{\Sigma}_\xi$ rather than the cross-sectional covariance matrix $\boldsymbol{\Sigma}$.

In the cross-sectional case, we have H auxiliary equations with endogenous outcomes $y_{1i}, \ldots, y_{Hi}$. We will treat the main outcome y_{it} as stage $J = H + 1$, so $y_{Ji} = y_{it}$. The ERMs that we fit with `eintreg`, `eoprobit`, `eprobit`, and `eregress` are triangular, so we can order the equations such that the first depends only on exogenous covariates and instruments—say, $\mathbf{w}_{1i} = \mathbf{z}_i$—and for $j = 2$, $\ldots, J$, equation j depends only on the exogenous covariates and instruments $\mathbf{z}_i$ and the endogenous covariates from equation $h = j - 1$ and $y_{1i}, \ldots, y_{hi}$ below. These are stored together in $\mathbf{w}_{ji}$.

When we predict conditional probabilities for binary and ordinal outcomes, we condition on all the endogenous and exogenous covariates and instruments that affect y_{ji}. Conditional probabilities are calculated as the ratio of the joint density over the marginal density of the conditioning covariates. For binary or ordinal outcome y_{ji}, we have

$$\Pr(y_{ji} = Y | y_{1i}, \ldots, y_{(j-1)i}, \mathbf{z}_i) = \frac{f(Y, y_{1i}, \ldots, y_{(j-1)i} | \mathbf{z}_i)}{f(y_{1i}, \ldots, y_{(j-1)i} | \mathbf{z}_i)}$$

where the densities can be computed as described in [ERM] **eprobit**.

Now, suppose instead that y_{ji} is continuous. We can predict the probability that y_{ji} lies in the range (l_{ji}, u_{ji}):

$$\Pr(l_{ji}, u_{ji}) = \Pr(l_{ji} < y_{ji} < u_{ji} | y_{1i}, \ldots, y_{(j-1)i}, \mathbf{z}_i)$$

$$= \int_{(l_{ji}, u_{ji}) \times \mathbf{V}^\star_{(j-1)i}} \phi_j(v_{1i}, \ldots, v_{ji}, \boldsymbol{\Sigma}_j) dv_{ji} d\mathbf{v}^\star_{(j-1)i}$$

This integral can be evaluated using the methods discussed in *Likelihood for multiequation models* in [ERM] **eprobit**.

The conditional mean of continuous outcome y_{ji} is

$$E(y_{ji}|\mathbf{w}_{ji}) = \mathbf{w}_{ji}\boldsymbol{\beta}_j + E(v_{ji}|\mathbf{w}_{ji})$$

where $\mathbf{w}_{ji}$ contains the endogenous covariates $y_{1i}, \ldots, y_{(j-1)i}$ and exogenous covariates $\mathbf{z}_i$ that affect y_{ji}.

By conditioning on the binary and ordinal endogenous covariates $y_{1i}, \ldots, y_{(j-1)i}$, the errors $v_{hi}, \ldots, v_{Ji}$ become truncated normal. Together with v_{ji}, they have a truncated multivariate distribution. So the mean of the continuous endogenous covariate is calculated using the moment formulas for the truncated multivariate normal. The first and second moments of the doubly truncated multivariate normal were derived in Manjunath and Wilhelm (2012). Tallis (1961) derived the first and second moments of the multivariate normal with one-sided truncation.

A key result in Manjunath and Wilhelm (2012) is that

$$\int_{l_1}^{u_1} \ldots \int_{l_d}^{u_d} \epsilon_f \phi_d(\boldsymbol{\epsilon}, \boldsymbol{\Sigma}) \quad d\epsilon_1 \ldots d\epsilon_d = \sum_{k=1}^{d} \sigma_{fk} \{F_k(l_k) - F_k(u_k)\} \tag{1}$$

where the functions $F_k(\cdot)$ are defined as

$$F_k(e) = \int_{l_1}^{u_1} \ldots \int_{l_{k-1}}^{u_{k-1}} \int_{l_{k+1}}^{u_{k+1}} \phi_d(e_1, \ldots, e_{k-1}, e, e_{k+1}, \ldots, e_k, \boldsymbol{\Sigma}) de_1 \ldots de_{k-1} de_{k+1} \ldots de_d$$

The $F_k(\cdot)$ functions can be computed like the joint density in *Likelihood for multiequation models* in [ERM] **eprobit**. So we have

$$E(v_{ji}|\mathbf{w}_{ji}) = \frac{\sum_{k=j}^{J} \sigma_{jk} \{F_k(l_{ki}) - F_k(u_{ki})\}}{\Phi_J^*(\mathbf{l}_i, \mathbf{u}_i, \boldsymbol{\Sigma}_j)}$$

where $l_{ji} = -\infty$ and $u_{ji} = \infty$.

If there are continuous endogenous regressors in $y_{1i}, \ldots, y_{ji}$, we condition on them in calculating (1). As in the calculation of the joint density in *Likelihood for multiequation models* in [ERM] **eprobit**, we multiply by the marginal density and adjust the cutpoints and variance.

The constrained mean of continuous outcome y_{ji}, the mean of y_{ji} when y_{ji} falls between l_{ji} and u_{ji}, is

$$\begin{aligned} E(l_{ji}, u_{ji}) &= E(y_{ji}|\mathbf{w}_{ji}, l_{ji} < y_{ji} < u_{ji}) \\ &= \mathbf{w}_{ji}\boldsymbol{\beta}_j + E(v_{ji}|\mathbf{w}_{ji}, l_{ji} - \mathbf{w}_{ji}\boldsymbol{\beta}_j < \epsilon_{ji} < v_{ji} - \mathbf{w}_{ji}\boldsymbol{\beta}_j) \end{aligned}$$

We use the same method as for the unconstrained mean, with cutpoints $l_{ji} - \mathbf{w}_{ji}\boldsymbol{\beta}_j$ and $u_{ji} - \mathbf{w}_{ji}\boldsymbol{\beta}_j$ instead of $-\infty$ and ∞.

The expected value of continuous y_{ji} with censoring at l_{ji} and u_{ji} is

$$\begin{aligned} E(y_{ji}^{\star}|\mathbf{w}_{ji}) &= l_{ji}\mathbf{1}(\mathbf{w}_{ji}\boldsymbol{\beta}_j + \epsilon_{ji} < l_{ji}) + u_{ji}\mathbf{1}(\mathbf{w}_{ji}\boldsymbol{\beta}_j + \epsilon_{ji} > u_{ji}) \\ &\quad + (\mathbf{w}_{ji}\boldsymbol{\beta}_j + \epsilon_{ji})\mathbf{1}(l_{ji} \leq \mathbf{w}_{ji}\boldsymbol{\beta}_j + \epsilon_{ji} \leq u_{ji}) \end{aligned}$$

where $y_{ji}^{\star} = \max\{l_{ij}, \min(y_{ij}, u_{ij})\}$. This can be calculated using predictions we have already discussed:

$$E(y_{ji}^{\star}|\mathbf{w}_{ji}) = \Pr(-\infty, l_{ji})l_{ji} + \Pr(l_{ji}, u_{ji})E(l_{ji}, u_{ji}) + \Pr(u_{ji}, \infty)u_{ji}$$

Sometimes, we model a continuous outcome y_{ji} that is the natural logarithm of another outcome y_{ji}^e. In this case, the conditional mean of y_{ji}^e is

$$
E(y_{ji}^e | \mathbf{w}_{ji}) = E\left\{\exp(y_{ji}) | \mathbf{w}_{ji}\right\} = E\left\{\exp\left(\mathbf{w}_{ji}\boldsymbol{\beta}_j + v_{ji}\right)\mathbf{w}_{ji}\right\}
$$
$$
= \exp\left(\mathbf{w}_{ji}\boldsymbol{\beta}_j\right) E\left\{\exp\left(v_{ji}\right) | \mathbf{w}_{ji}\right\}
$$

As discussed earlier, v_{ji} can be truncated normal when we condition on $\mathbf{w}_{ji}$. So the conditional expectation above is the moment-generating function of a truncated normal random variable. This function was also derived in Manjunath and Wilhelm (2012). Letting $\boldsymbol{\sigma}_j$ be the jth column of $\boldsymbol{\Sigma}_j$, we have

$$
E\left\{\exp\left(v_{ji}\right) | \mathbf{w}_{ji}\right\} = \exp\left(\frac{\sigma_j^2}{2}\right)\frac{\Phi_j^*(\mathbf{l}_i - \boldsymbol{\sigma}_j, \mathbf{u}_i - \boldsymbol{\sigma}_j, \boldsymbol{\Sigma}_j)}{\Phi_j^*(\mathbf{l}_i, \mathbf{u}_i, \boldsymbol{\Sigma}_j)}
$$

All the predictions above can be made after estimation by using `predict`. By also specifying either the `pr` or the `pr(`l_{ji}`,`u_{ji}`)` option in `predict`, we can obtain conditional probabilities for a binary or ordinal outcome or the conditional probability that a continuous outcome lies in the specified range (l_{ji}, u_{ji}).

By also specifying the `mean` option, we obtain the conditional mean of a continuous endogenous covariate. The `e(`l_{ji}`,`u_{ji}`)` option is used to obtain the constrained mean, and `ystar(`l_{ji}`,`u_{ji}`)` is used to obtain the expected value with censoring.

Prediction of treatment effects and potential-outcome means in models with endogenous covariates use the above formulas for the conditional mean and probabilities applied to the potential outcomes $y_{1i}, \ldots, y_{Ti}$ rather than the observed y_i. Methods and formulas for other predictions are given in the *Methods and formulas* sections of [ERM] **eoprobit**, [ERM] **eintreg**, and [ERM] **eregress**.

References

Imbens, G. W., and W. K. Newey. 2009. Identification and estimation of triangular simultaneous equations models without additivity. *Econometrica* 77: 1481–1512. https://doi.org/10.3982/ECTA7108.

Manjunath, B. G., and S. Wilhelm. 2012. Moments calculation for the doubly truncated multivariate normal density. Unpublished manuscript. https://arxiv.org/pdf/1206.5387.pdf.

Tallis, G. M. 1961. The moment generating function of the truncated multi-normal distribution. *Journal of the Royal Statistical Society, Series B* 23: 223–229. https://doi.org/10.1111/j.2517-6161.1961.tb00408.x.

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

Also see

[ERM] **eprobit** — Extended probit regression

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

[ERM] **predict treatment** — predict for treatment statistics

[ERM] **predict advanced** — predict's advanced features

[U] **20 Estimation and postestimation commands**

Title

eprobit predict — predict after eprobit and xteprobit

Description

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

Syntax

You previously fit the model

```
eprobit y x1 ..., ...
```

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

$$\Pr(y_i) = \Pr(\beta_0 + \beta_1 \mathtt{x1}_i + \cdots + e_i.\mathtt{y} > 0)$$

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

```
xteprobit y x1 ..., ...
```

Then the main equation would be

$$\Pr(y_{ij}) = \Pr(\beta_0 + \beta_1 \mathtt{x1}_{ij} + \cdots + u_i.\mathtt{y} + v_{ij}.\mathtt{y} > 0)$$

In either case, `predict` calculates predictions for $\Pr(\mathtt{y})$ 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, $e_{ij}.\mathtt{y} = u_i.\mathtt{y} + v_{ij}.\mathtt{y}$.

All predictions after `xteprobit` assume the panel-level random effects ($u_i.\mathtt{y}$) 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	
pr	probability of a positive outcome; the default
xb	linear prediction excluding all complications

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

pr, the default, calculates the predicted probability of a positive outcome. In each observation, the prediction is the probability 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.

Options for asfmethod

`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 probability of a positive outcome.
>
> > `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

`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 $\beta_0 + \beta_1 \mathtt{x1}_i + \cdots$. Use of `target()` is discussed in *Remarks and examples* of [ERM] **eregress predict**.

Remarks and examples

Remarks are presented under the following headings:

> *Using predict after eprobit*
> *How to think about nonlinear models*

Using predict after eprobit

Predictions after fitting models with `eprobit` or `xteprobit` are handled the same as they are after fitting models with `eregress` and `xteregress`. The issues are the same. See [ERM] **eregress predict**.

How to think about nonlinear models

Probit is a nonlinear model, and yet we just said that predictions after fitting models with `eprobit` and `xteprobit` are handled the same as they are after fitting models with `eregress`. That statement is partly true, not misleading, but false in its details.

The regression-based discussion that we routed you to is framed in terms of expected values. In the nonlinear models, it needs to be framed in terms of distributional assumptions about the errors. For instance, `predict` after `eprobit` does not predict the expected value (mean) of $e_i.y$. It calculates the probability that $e_i.y$ exceeds $-\mathbf{x}_i\boldsymbol{\beta}$. These details matter hugely in implementation but can be glossed over for understanding the issues. For a full treatment of the issues, see *Methods and formulas* in [ERM] **eprobit**.

Methods and formulas

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

Also see

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

[ERM] **eprobit** — Extended probit regression

Title

eregress — Extended linear regression

Description

eregress fits a linear regression model that accommodates any combination of endogenous co-variates, nonrandom treatment assignment, and endogenous sample selection. Continuous, binary, and ordinal endogenous covariates are allowed. Treatment assignment may be endogenous or exogenous. A probit or tobit model may be used to account for endogenous sample selection.

xteregress fits a random-effects linear regression model that accommodates endogenous covari-ates, treatment, and sample selection in the same way as eregress and also accounts for correlation of observations within panels or within groups.

Quick start

Regression of y on x with continuous endogenous covariate y2 modeled by x and z

        eregress y x, endogenous(y2 = x z)

Same as above, but adding continuous endogenous covariate y3 modeled by x and z2

        eregress y x, endogenous(y2 = x z) endogenous(y3 = x z2)

Regression of y on x with binary endogenous covariate d modeled by x and z

        eregress y x, endogenous(d = x z, probit)

Regression of y on x with endogenous treatment recorded in trtvar and modeled by x and z

        eregress y x, entreat(trtvar = x z)

Regression of y on x with exogenous treatment recorded in trtvar

        eregress y x, extreat(trtvar)

Random-effects regression of y on x using xtset data

        xteregress y x

Regression of y on x with endogenous sample-selection indicator selvar modeled by x and z

        eregress y x, select(selvar = x z)

Same as above, but adding endogenous covariate y2 modeled by x and z2

        eregress y x, select(selvar = x z) endogenous(y2 = x z2)

Same as above, but adding endogenous treatment recorded in trtvar and modeled by x and z3

        eregress y x, select(selvar = x z) endogenous(y2 = x z2) ///
            entreat(trtvar = x z3)

Same as above, but with random effects and without endogenous treatment

        xteregress y x, select(selvar = x z) endogenous(y2 = x z2)

Menu

eregress

Statistics > Endogenous covariates > Models adding selection and treatment > Linear regression

xteregress

Statistics > Longitudinal/panel data > Endogenous covariates > Models adding selection and treatment > Linear regression (RE)

Syntax

Basic linear regression with endogenous covariates

    eregress *depvar* [*indepvars*] , <u>endog</u>enous(*depvars*$_{en}$ = *varlist*$_{en}$) [*options*]

Basic linear regression with endogenous treatment assignment

    eregress *depvar* [*indepvars*] , <u>entr</u>eat(*depvar*$_{tr}$ [= *varlist*$_{tr}$]) [*options*]

Basic linear regression with exogenous treatment assignment

    eregress *depvar* [*indepvars*] , <u>extr</u>eat(*tvar*) [*options*]

Basic linear regression with sample selection

    eregress *depvar* [*indepvars*] , <u>sel</u>ect(*depvar*$_s$ = *varlist*$_s$) [*options*]

Basic linear regression with tobit sample selection

    eregress *depvar* [*indepvars*] , <u>tobitsel</u>ect(*depvar*$_s$ = *varlist*$_s$) [*options*]

Basic linear regression with random effects

    xteregress *depvar* [*indepvars*] [, *options*]

Linear regression combining endogenous covariates, treatment, and selection

    eregress *depvar* [*indepvars*] [*if*] [*in*] [*weight*] [, *extensions options*]

Linear regression combining random effects, endogenous covariates, treatment, and selection

    xteregress *depvar* [*indepvars*] [*if*] [*in*] [, *extensions options*]

extensions	Description
Model	
endogenous(*enspec*)	model for endogenous covariates; may be repeated
entreat(*entrspec*)	model for endogenous treatment assignment
extreat(*extrspec*)	exogenous treatment
select(*selspec*)	probit model for selection
tobitselect(*tselspec*)	tobit model for selection

options	Description
Model	
noconstant	suppress constant term
offset(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1
constraints(*numlist*)	apply specified linear constraints
SE/Robust	
vce(*vcetype*)	*vcetype* may be oim, robust, cluster *clustvar*, opg, bootstrap, or jackknife
Reporting	
level(*#*)	set confidence level; default is level(95)
nocnsreport	do not display constraints
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Integration	
intpoints(*#*)	set the number of integration (quadrature) points for integration over four or more dimensions; default is intpoints(128)
triintpoints(*#*)	set the number of integration (quadrature) points for integration over three dimensions; default is triintpoints(10)
reintpoints(*#*)	set the number of integration (quadrature) points for random-effects integration; default is reintpoints(7)
reintmethod(*intmethod*)	integration method for random effects; *intmethod* may be mvaghermite (the default) or ghermite
Maximization	
maximize_options	control the maximization process; seldom used
collinear	keep collinear variables
coeflegend	display legend instead of statistics

enspec is *depvars*$_{en}$ = *varlist*$_{en}$ [, *enopts*]

   where *depvars*$_{en}$ is a list of endogenous covariates. Each variable in *depvars*$_{en}$ specifies an endogenous covariate model using the common *varlist*$_{en}$ and options.

entrspec is *depvar*$_{tr}$ [= *varlist*$_{tr}$] [, *entropts*]

   where *depvar*$_{tr}$ is a variable indicating treatment assignment. *varlist*$_{tr}$ is a list of covariates predicting treatment assignment.

extrspec is *tvar* [, *extropts*]

 where *tvar* is a variable indicating treatment assignment.

selspec is *depvar$_s$* = *varlist$_s$* [, *selopts*]

 where *depvar$_s$* is a variable indicating selection status. *depvar$_s$* must be coded as 0, indicating
 that the observation was not selected, or 1, indicating that the observation was selected. *varlist$_s$*
 is a list of covariates predicting selection.

tselspec is *depvar$_s$* = *varlist$_s$* [, *tselopts*]

 where *depvar$_s$* is a continuous variable. *varlist$_s$* is a list of covariates predicting *depvar$_s$*. The
 censoring status of *depvar$_s$* indicates selection, where a censored *depvar$_s$* indicates that the
 observation was not selected and a noncensored *depvar$_s$* indicates that the observation was
 selected.

enopts	Description
Model	
probit	treat endogenous covariate as binary
oprobit	treat endogenous covariate as ordinal
povariance	estimate a different variance for each level of a binary or an ordinal endogenous covariate
pocorrelation	estimate different correlations for each level of a binary or an ordinal endogenous covariate
nomain	do not add endogenous covariate to main equation
nore	do not include random effects in model for endogenous covariate
noconstant	suppress constant term

nore is available only with xteregress.

entropts	Description
Model	
povariance	estimate a different variance for each potential outcome
pocorrelation	estimate different correlations for each potential outcome
nomain	do not add treatment indicator to main equation
nointeract	do not interact treatment with covariates in main equation
nore	do not include random effects in model for endogenous treatment
noconstant	suppress constant term
offset(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1

nore is available only with xteregress.

extropts	Description
Model	
povariance	estimate a different variance for each potential outcome
pocorrelation	estimate different correlations for each potential outcome
nomain	do not add treatment indicator to main equation
nointeract	do not interact treatment with covariates in main equation

selopts	Description
Model	
<u>nore</u>	do not include random effects in selection model
<u>noco</u>nstant	suppress constant term
<u>off</u>set(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1

nore is available only with xteregress.

tselopts	Description
Model	
*<u>ll</u>(*varname* | #)	left-censoring variable or limit
*<u>ul</u>(*varname* | #)	right-censoring variable or limit
main	add censored selection variable to main equation
<u>nore</u>	do not include random effects in tobit selection model
<u>noco</u>nstant	suppress constant term
<u>off</u>set(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1

* You must specify either ll() or ul().

nore is available only with xteregress.

indepvars, *varlist*$_{en}$, *varlist*$_{tr}$, and *varlist*$_s$ may contain factor variables; see [U] **11.4.3 Factor variables**.

depvar, *indepvars*, *depvars*$_{en}$, *varlist*$_{en}$, *depvar*$_{tr}$, *varlist*$_{tr}$, *tvar*, *depvar*$_s$, and *varlist*$_s$ may contain time-series operators; see [U] **11.4.4 Time-series varlists**.

bootstrap, by, collect, jackknife, and statsby are allowed with eregress and xteregress. rolling and svy are allowed with eregress. See [U] **11.1.10 Prefix commands**.

Weights are not allowed with the bootstrap prefix; see [R] **bootstrap**.

vce() and weights are not allowed with the svy prefix; see [SVY] **svy**.

fweights, iweights, and pweights are allowed with eregress; see [U] **11.1.6 weight**.

reintpoints() and reintmethod() are available only with xteregress.

collinear and coeflegend do not appear in the dialog box.

See [U] **20 Estimation and postestimation commands** for more capabilities of estimation commands.

Options

　　┌─ Model ──

endogenous(*enspec*), entreat(*entrspec*), extreat(*extrspec*), select(*selspec*),
　　tobitselect(*tselspec*); see [ERM] **ERM options**.

noconstant, offset(*varname$_o$*), constraints(*numlist*); see [R] **Estimation options**.

　　┌─ SE/Robust ──

vce(*vcetype*); see [ERM] **ERM options**.

　　┌─ Reporting ──

level(#), nocnsreport; see [R] **Estimation options**.

display_options: noci, <u>nopv</u>alues, <u>noomitted</u>, vsquish, noemptycells, <u>base</u>levels,
　　<u>allbase</u>levels, <u>nofvlabel</u>, fvwrap(#), fvwrapon(*style*), cformat(%*fmt*), pformat(%*fmt*),
　　sformat(%*fmt*), and nolstretch; see [R] **Estimation options**.

> ⎡ Integration ⎤

intpoints(*#*), triintpoints(*#*), reintpoints(*#*), reintmethod(*intmethod*); see [ERM] **ERM options**.

> ⎡ Maximization ⎤

maximize_options: difficult, technique(*algorithm_spec*), iterate(*#*), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(*#*), ltolerance(*#*), nrtolerance(*#*), nonrtolerance, and from(*init_specs*); see [R] **Maximize**.

The default technique for eregress is technique(nr). The default technique for xteregress is technique(bhhh 10 nr 2).

Setting the optimization type to technique(bhhh) resets the default *vcetype* to vce(opg).

The following options are available with eregress and xteregress but are not shown in the dialog box:

collinear, coeflegend; see [R] **Estimation options**.

Remarks and examples

eregress and xteregress fit models that we refer to as "extended linear regression models", meaning that they accommodate endogenous covariates, nonrandom treatment assignment, endogenous sample selection, and panel data or other grouped data.

eregress fits models for cross-sectional data (one-level models). eregress can account for endogenous covariates, treatment, and sample selection, whether these complications arise individually or in combination.

xteregress fits random-effects models (two-level models) for panel data or grouped data. xteregress accounts for endogenous covariates, treatment, and sample selection in the same way as eregress and also accounts for within-panel or within-group correlation among observations.

In this entry, you will find information on the syntax for the eregress and xteregress commands. You can see *Methods and formulas* for a full description of the models that can be fit with eregress and xteregress and details about how those models are fit.

More information on extended linear regression models is found in the separate introductions and example entries. We recommend reading those entries to learn how to use eregress and xteregress. Below, we provide a guide to help you locate the ones that will be helpful to you.

For an introduction to eregress and xteregress and the other extended regression commands for interval, binary, and ordinal outcomes, see [ERM] **Intro 1**–[ERM] **Intro 9**.

[ERM] **Intro 1** introduces the ERM commands, the problems they address, and their syntax.

[ERM] **Intro 2** provides background on the four types of models—linear regression, interval regression, probit regression, and ordered probit regression—that can be fit using ERM commands.

[ERM] **Intro 3** considers the problem of endogenous covariates and how to solve it using ERM commands.

[ERM] **Intro 4** gives an overview of endogenous sample selection and using ERM commands to account for it.

[ERM] **Intro 5** covers nonrandom treatment assignment and how to account for it using eregress or any of the other ERM commands.

[ERM] **Intro 6** covers random-effects models for panel data and other grouped data. It discusses xteregress and the other ERM commands for panel data.

[ERM] **Intro 7** discusses interpretation of results. You can interpret coefficients from eregress and xteregress in the usual way, but this introduction goes beyond the interpretation of coefficients. We demonstrate how to find answers to interesting questions by using margins. If your model includes an endogenous covariate or an endogenous treatment, the use of margins differs from its use after other estimation commands, so we strongly recommend reading this intro if you are fitting these types of models.

[ERM] **Intro 8** will be helpful if you are familiar with heckman, ivregress, etregress, xtreg, or xtivreg and other commands that address endogenous covariates, sample selection, nonrandom treatment assignment, or random effects. This introduction is a Rosetta stone that maps the syntax of those commands to the syntax of eregress and xteregress.

[ERM] **Intro 9** walks you through an example that gives insight into the concepts of endogenous covariates, treatment assignment, and sample selection while fitting models with eregress that address these complications. This intro also demonstrates how to interpret results by using margins and estat teffects.

Additional examples are presented in [ERM] **Example 1a**–[ERM] **Example 9**. For examples using eregress, see

[ERM] **Example 1a**	Linear regression with continuous endogenous covariate
[ERM] **Example 2a**	Linear regression with binary endogenous covariate
[ERM] **Example 2b**	Linear regression with exogenous treatment
[ERM] **Example 2c**	Linear regression with endogenous treatment

For examples using xteregress, see

[ERM] **Example 7**	Random-effects regression with continuous endogenous covariate
[ERM] **Example 8a**	Random-effects regression with constraint and endogenous covariate
[ERM] **Example 8b**	Random-effects, endogenous covariate, and endogenous sample selection

See *Examples* in [ERM] **Intro** for an overview of all the examples. All examples may be interesting because they handle complications in the same way.

eregress and xteregress fit many models discussed in the literature. For example, eregress can fit the linear regression model with endogenous sample selection (Heckman 1976), the linear regression model with an endogenous treatment (Heckman 1978; Maddala 1983), and the linear regression model with a tobit selection equation (Amemiya 1985; Wooldridge 2010, sec. 19.7). eregress also supports the linear regression model with endogenous regressors and endogenous sample selection discussed in Wooldridge (2010, sec 19.6) along with the tobit selection regression with endogenous regressors discussed in Wooldridge (2010, sec 19.7).

For panel data, xteregress can fit the linear regression model with random effects discussed in Baltagi (2013, chap. 2) and Wooldridge (2020, chap. 14). The xteregress command can also fit the linear regression model with an endogenous treatment and random effects discussed in Drukker (2016) and the linear regression model with random effects and endogenous covariates discussed in Baltagi (2013). Roodman (2011) investigated linear regression models with endogenous covariates and endogenous sample selection and demonstrated how multiple observational data complications could be addressed with a triangular model structure. He and Tamás Bartus showed how random effects could be used in the triangular model structure in Bartus and Roodman (2014). Roodman's work has been used to model processes like the effect of aphid infestations and virus outbreaks on crop yields (Elbakidze, Lu, and Eigenbrode 2011) and the effect of calorie intake per day on food security in poor neighborhoods (Maitra and Rao 2014).

Stored results

eregress stores the following in e():

Scalars
e(N)	number of observations
e(N_selected)	number of selected observations
e(N_nonselected)	number of nonselected observations
e(k)	number of parameters
e(k_cat#)	number of categories for the #th *depvar*, ordinal
e(k_eq)	number of equations in e(b)
e(k_eq_model)	number of equations in overall model test
e(k_dv)	number of dependent variables
e(k_aux)	number of auxiliary parameters
e(df_m)	model degrees of freedom
e(ll)	log likelihood
e(N_clust)	number of clusters
e(chi2)	χ^2
e(p)	*p*-value for model test
e(n_quad)	number of integration points for multivariate normal
e(n_quad3)	number of integration points for trivariate normal
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise

Macros
e(cmd)	eregress
e(cmdline)	command as typed
e(depvar)	names of dependent variables
e(tsel_ll)	left-censoring limit for tobit selection
e(tsel_ul)	right-censoring limit for tobit selection
e(wtype)	weight type
e(wexp)	weight expression
e(title)	title in estimation output
e(clustvar)	name of cluster variable
e(offset#)	offset for the #th *depvar*, where # is determined by equation order in output
e(chi2type)	Wald; type of model χ^2 test
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. err.
e(opt)	type of optimization
e(which)	max or min; whether optimizer is to perform maximization or minimization
e(ml_method)	type of ml method
e(user)	name of likelihood-evaluator program
e(technique)	maximization technique
e(properties)	b V
e(estat_cmd)	program used to implement estat
e(predict)	program used to implement predict
e(marginsok)	predictions allowed by margins
e(marginsnotok)	predictions disallowed by margins
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices
e(b)	coefficient vector
e(cat#)	categories for the #th *depvar*, ordinal
e(Cns)	constraints matrix
e(ilog)	iteration log (up to 20 iterations)
e(gradient)	gradient vector
e(V)	variance–covariance matrix of the estimators
e(V_modelbased)	model-based variance

Functions
e(sample)	marks estimation sample

In addition to the above, the following is stored in `r()`:

Matrices
    `r(table)`   matrix containing the coefficients with their standard errors, test statistics, *p*-values, and confidence intervals

Note that results stored in `r()` are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

`xteregress` stores the following in `e()`:

Scalars

`e(N)`	number of observations
`e(N_g)`	number of groups
`e(N_selected)`	number of selected observations
`e(N_nonselected)`	number of nonselected observations
`e(k)`	number of parameters
`e(k_cat#)`	number of categories for the #th *depvar*, ordinal
`e(k_eq)`	number of equations in `e(b)`
`e(k_eq_model)`	number of equations in overall model test
`e(k_dv)`	number of dependent variables
`e(k_aux)`	number of auxiliary parameters
`e(df_m)`	model degrees of freedom
`e(ll)`	log likelihood
`e(N_clust)`	number of clusters
`e(chi2)`	χ^2
`e(p)`	*p*-value for model test
`e(n_quad)`	number of integration points for multivariate normal
`e(n_quad3)`	number of integration points for trivariate normal
`e(n_requad)`	number of integration points for random effects
`e(g_min)`	smallest group size
`e(g_avg)`	average group size
`e(g_max)`	largest group size
`e(rank)`	rank of `e(V)`
`e(ic)`	number of iterations
`e(rc)`	return code
`e(converged)`	1 if converged, 0 otherwise

Macros

`e(cmd)`	`xteregress`
`e(cmdline)`	command as typed
`e(depvar)`	names of dependent variables
`e(tsel_ll)`	left-censoring limit for tobit selection
`e(tsel_ul)`	right-censoring limit for tobit selection
`e(ivar)`	variable denoting groups
`e(title)`	title in estimation output
`e(clustvar)`	name of cluster variable
`e(offset#)`	offset for the #th *depvar*, where # is determined by equation order in output
`e(chi2type)`	Wald; type of model χ^2 test
`e(vce)`	*vcetype* specified in `vce()`
`e(vcetype)`	title used to label Std. err.
`e(reintmethod)`	integration method for random effects
`e(opt)`	type of optimization
`e(which)`	max or min; whether optimizer is to perform maximization or minimization
`e(ml_method)`	type of ml method
`e(user)`	name of likelihood-evaluator program
`e(technique)`	maximization technique
`e(properties)`	b V
`e(estat_cmd)`	program used to implement estat
`e(predict)`	program used to implement predict
`e(marginsok)`	predictions allowed by margins
`e(marginsnotok)`	predictions disallowed by margins
`e(asbalanced)`	factor variables fvset as asbalanced
`e(asobserved)`	factor variables fvset as asobserved

Matrices
    e(b)   coefficient vector
    e(cat#)   categories for the #th *depvar*, ordinal
    e(Cns)   constraints matrix
    e(ilog)   iteration log (up to 20 iterations)
    e(gradient)   gradient vector
    e(V)   variance–covariance matrix of the estimators
    e(V_modelbased)   model-based variance
Functions
    e(sample)   marks estimation sample

In addition to the above, the following is stored in `r()`:

Matrices
    r(table)   matrix containing the coefficients with their standard errors, test statistics, p-values,
                            and confidence intervals

Note that results stored in `r()` are updated when the command is replayed and will be replaced when
any r-class command is run after the estimation command.

Methods and formulas

The methods and formulas presented here are for the linear model. The estimators implemented in
`eregress` and `xteregress` are maximum likelihood estimators covered by the results in chapter 13
of Wooldridge (2010) and White (1996).

The log-likelihood functions maximized by `eregress` and `xteregress` are implied by the
triangular structure of the model. Specifically, the joint distribution of the endogenous variables is a
product of conditional and marginal distributions because the model is triangular. For a few of the
many relevant applications of this result in literature, see chapter 10 of Amemiya (1985); Heckman
(1976, 1979); chapter 5 of Maddala (1983); Maddala and Lee (1976); sections 15.7.2, 15.7.3, 16.3.3,
17.5.2, and 19.7.1 in Wooldridge (2010); and Wooldridge (2014). Roodman (2011) and Bartus and
Roodman (2014) used this result to derive the formulas discussed below.

Methods and formulas are presented under the following headings:

> Introduction
> Endogenous covariates
> > Continuous endogenous covariates
> > Binary and ordinal endogenous covariates
> Treatment
> Endogenous sample selection
> > Probit endogenous sample selection
> > Tobit endogenous sample selection
> Random effects
> Combinations of features
> Confidence intervals

Introduction

A linear regression of outcome y_i on covariates $\mathbf{x}_i$ may be written as

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \epsilon_i$$

where the error ϵ_i is normal with mean 0 and variance σ^2. The log likelihood is

$$\ln L = \sum_{i=1}^{N} w_i \ln \phi\big(y_i - \mathbf{x}_i\boldsymbol{\beta}, \sigma^2\big)$$

The conditional mean of y_i is

$$E(y_i|\mathbf{x}_i) = \mathbf{x}_i\boldsymbol{\beta}$$

If you are willing to take our word for some derivations and notation, the following is complete. Longer explanations and derivations for some terms and functions are provided in *Methods and formulas* of [ERM] **eprobit**. For example, we need the two-sided probability function Φ_d^* that is discussed in *Introduction* in [ERM] **eprobit**.

If you are interested in all the details, we suggest you read *Methods and formulas* of [ERM] **eprobit** in its entirety before reading this section. Here we mainly show how the complications that arise in ERMs are handled in a linear regression framework.

Endogenous covariates

Continuous endogenous covariates

A linear regression of y_i on exogenous covariates $\mathbf{x}_i$ and C continuous endogenous covariates $\mathbf{w}_{ci}$ has the form

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{w}_{ci}\boldsymbol{\beta}_c + \epsilon_i$$
$$\mathbf{w}_{ci} = \mathbf{z}_{ci}\mathbf{A}_c + \boldsymbol{\epsilon}_{ci}$$

The vector $\mathbf{z}_{ci}$ contains variables from $\mathbf{x}_i$ and other covariates that affect $\mathbf{w}_{ci}$. For the model to be identified, $\mathbf{z}_{ci}$ must contain one extra exogenous covariate not in $\mathbf{x}_i$ for each of the endogenous regressors in $\mathbf{w}_{ci}$. The unobserved errors ϵ_i and $\boldsymbol{\epsilon}_{ci}$ are multivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma^2 & \boldsymbol{\sigma}'_{1c} \\ \boldsymbol{\sigma}_{1c} & \boldsymbol{\Sigma}_c \end{bmatrix}$$

The log likelihood is

$$\ln L = \sum_{i=1}^{N} w_i \ln\phi_{C+1}(\mathbf{r}_i, \boldsymbol{\Sigma})$$

where

$$\mathbf{r}_i = \begin{bmatrix} y_i - \mathbf{x}_i & \mathbf{w}_{ci} - \mathbf{z}_{ci}\mathbf{A}_c \end{bmatrix}$$

The conditional mean of y_i is

$$E(y_i|\mathbf{x}_i, \mathbf{w}_{ci}, \mathbf{z}_{ci}) = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{w}_{ci}\boldsymbol{\beta}_c + \boldsymbol{\sigma}'_{1c}\boldsymbol{\Sigma}_c^{-1}(\mathbf{w}_{ci} - \mathbf{z}_{ci}\mathbf{A}_c)'$$

Binary and ordinal endogenous covariates

Here we begin by formulating the linear regression of y_i on exogenous covariates $\mathbf{x}_i$ and B binary and ordinal endogenous covariates $\mathbf{w}_{bi} = [w_{b1i}, \ldots, w_{bBi}]$. Indicator (dummy) variables for the levels of each binary and ordinal covariate are used in the model. You can also interact other covariates with the binary and ordinal endogenous covariates, as in treatment-effect models.

The binary and ordinal endogenous covariates $\mathbf{w}_{bi}$ are formulated as in *Binary and ordinal endogenous covariates* in [ERM] **eprobit**.

The model for the outcome can be formulated with or without different variance and correlation parameters for each level of $\mathbf{w}_{bi}$. Level-specific parameters are obtained by specifying `povariance` or `pocorrelation` in the `endogenous()` option.

If the variance and correlation parameters are not level specific, we have

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB} + \epsilon_i$$

The $\mathbf{wind}_{bji}$ vectors are defined in *Binary and ordinal endogenous covariates* in [ERM] **eprobit**. The binary and ordinal endogenous errors $\epsilon_{b1i}, \ldots, \epsilon_{bBi}$ and outcome error ϵ_i are multivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_b & \boldsymbol{\sigma}_{1b} \\ \boldsymbol{\sigma}'_{1b} & \sigma^2 \end{bmatrix}$$

From here, we discuss the model with ordinal endogenous covariates. The results for binary endogenous covariates are similar.

Using results from *Likelihood for multiequation models* in [ERM] **eprobit**, we can write the joint density of y_i and $\mathbf{w}_{bi}$ using the conditional density of $\epsilon_{b1i}, \ldots, \epsilon_{bBi}$ on ϵ_i.

Define

$$r_i = y_i - (\mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB})$$

Let

$$\boldsymbol{\mu}_{b|1,i} = \frac{\boldsymbol{\sigma}'_{1b}}{\sigma^2} r_i = \begin{bmatrix} e_{b1i} \ldots e_{bBi} \end{bmatrix}$$

$$\boldsymbol{\Sigma}_{b|1} = \boldsymbol{\Sigma}_b - \frac{\boldsymbol{\sigma}_{1b}\boldsymbol{\sigma}'_{1b}}{\sigma^2}$$

For $j = 1, \ldots, B$ and $h = 0, \ldots, B_j$, let

$$c_{bjih} = \begin{cases} -\infty & h = 0 \\ \kappa_{bjh} - \mathbf{z}_{bji}\boldsymbol{\alpha}_{bj} - e_{bji} & h = 1, \ldots, B_j - 1 \\ \infty & h = B_j \end{cases}$$

So, for $j = 1, \ldots, B$, the probability for w_{bji} has lower limit

$$l_{bji} = c_{bji(h-1)} \quad \text{if} \quad w_{bji} = v_{bjh}$$

and upper limit

$$u_{bji} = c_{bjih} \quad \text{if} \quad w_{bji} = v_{bjh}$$

Let

$$\mathbf{l}_i = \begin{bmatrix} l_{b1i} & \ldots & l_{bBi} \end{bmatrix}$$

$$\mathbf{u}_i = \begin{bmatrix} u_{b1i} & \ldots & u_{bBi} \end{bmatrix}$$

So, the log likelihood for this model is

$$\ln L = \sum_{i=1}^{N} w_i \ln \left\{ \Phi^*_B(\mathbf{l}_i, \mathbf{u}_i, \boldsymbol{\Sigma}_{b|1})\phi(r_i, \sigma^2) \right\}$$

The expected value of y_i conditional on $\mathbf{w}_{bi}$ can be calculated using the techniques discussed in *Predictions using the full model* in [ERM] **eprobit postestimation**.

When the endogenous ordinal variables are different treatments, holding the variance and correlation parameters constant over the treatment levels is a constrained form of the potential-outcome model. In an unconstrained potential-outcome model, the variance of the outcome and the correlations between the outcome and the treatments—the endogenous ordinal regressors $\mathbf{w}_{bi}$—vary over the levels of each treatment.

In this unconstrained model, there is a different potential-outcome error for each level of each treatment. For example, when the endogenous treatment variable w_1 has three levels (0, 1, and 2) and the endogenous treatment variable w_2 has four levels (0, 1, 2, and 3), the unconstrained model has $12 = 3 \times 4$ outcome errors. So there are 12 outcome error variance parameters. Because there is a different correlation between each potential outcome and each endogenous treatment, there are 2×12 correlation parameters between the potential outcomes and the treatments in this example model.

We denote the number of different combinations of values for the endogenous treatments $\mathbf{w}_{bi}$ by M, and we denote the vector of values in each combination by $\mathbf{v}_j$ ($j \in \{1, 2, \ldots, M\}$). Letting k_{wp} be the number of levels of endogenous ordinal treatment variable $p \in \{1, 2, \ldots, B\}$ implies that $M = k_{w1} \times k_{w2} \times \cdots \times k_{wB}$.

Denoting the outcome errors ϵ_{1i}, ..., ϵ_{Mi}, we have

$$y_{1i} = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB} + \epsilon_{1i}$$

$$\vdots$$

$$y_{Mi} = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{wind}_{b1i}\boldsymbol{\beta}_{b1} + \cdots + \mathbf{wind}_{bBi}\boldsymbol{\beta}_{bB} + \epsilon_{Mi}$$

$$y_i = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j)y_{ji}$$

For $j = 1$, ..., M, the endogenous errors $\epsilon_{b1i}, \ldots, \epsilon_{bBi}$ and outcome error ϵ_{ji} are multivariate normal with 0 mean and covariance

$$\boldsymbol{\Sigma}_j = \begin{bmatrix} \boldsymbol{\Sigma}_b & \boldsymbol{\sigma}_{j1b} \\ \boldsymbol{\sigma}'_{j1b} & \sigma_j^2 \end{bmatrix}$$

Now let

$$\sigma_{i,b} = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j)\sigma_j$$

$$\boldsymbol{\Sigma}_{i,b|1} = \sum_{j=1}^{M} 1(\mathbf{w}_{bi} = \mathbf{v}_j)\left(\boldsymbol{\Sigma}_b - \frac{\boldsymbol{\sigma}_{j1b}\boldsymbol{\sigma}'_{j1b}}{\sigma_j^2}\right)$$

Now the log likelihood for this model is

$$\ln L = \sum_{i=1}^{N} w_i \ln\left\{\Phi_B^*(\mathbf{l}_i, \mathbf{u}_i, \boldsymbol{\Sigma}_{i,b|1})\phi(r_i, \sigma_{i,b}^2)\right\}$$

As in the other case, the expected value of y_i conditional on $\mathbf{w}_{bi}$ can be calculated using the techniques discussed in *Predictions using the full model* in [ERM] **eprobit postestimation**.

Treatment

In the potential-outcomes framework, the treatment t_i is a discrete variable taking T values, indexing the T potential outcomes of the outcome y_i: $y_{1i}, \ldots, y_{Ti}$.

When we observe treatment t_i with levels $v_1, \ldots, v_T$, we have

$$y_i = \sum_{j=1}^{T} 1(t_i = v_j)y_{ji}$$

So for each observation, we observe only the potential outcome associated with that observation's treatment value.

For exogenous treatments, our approach is equivalent to the regression adjustment treatment-effect estimation method. See [CAUSAL] **teffects intro advanced**. We do not model the treatment assignment process. The formulas for the treatment effects and potential-outcome means (POMs) are equivalent to what we provide here for endogenous treatments. The treatment effect on the treated for $\mathbf{x}_i$ for an exogenous treatment is equivalent to what we provide here for the endogenous treatment when the correlation parameter between the outcome and treatment errors is set to 0. The average treatment effects (ATEs) and POMs for exogenous treatments are estimated as predictive margins in an analogous manner to what we describe here for endogenous treatments. We can also obtain different variance parameters for the different exogenous treatment groups by specifying povariance in extreat().

From here, we assume an endogenous treatment t_i. As in *Treatment* in [ERM] **eprobit**, we model the treatment assignment process with a probit or ordered probit model, and we call the treatment assignment error ϵ_{ti}. A linear regression of y_i on exogenous covariates $\mathbf{x}_i$ and endogenous treatment t_i taking values $v_1, \ldots, v_T$ has the form

$$y_{1i} = \mathbf{x}_i\boldsymbol{\beta}_1 + \epsilon_{1i}$$

$$\vdots$$

$$y_{Ti} = \mathbf{x}_i\boldsymbol{\beta}_T + \epsilon_{Ti}$$

$$y_i = \sum_{j=1}^{T} 1(t_i = v_j)y_{ji}$$

This model can be formulated with or without different variance and correlation parameters for each potential outcome. Potential-outcome specific parameters are obtained by specifying povariance or pocorrelation in the entreat() option.

If the variance and correlation parameters are not potential-outcome specific, for $j = 1, \ldots, T$, ϵ_{ji} and ϵ_{ti} are bivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma^2 & \sigma\rho_{1t} \\ \sigma\rho_{1t} & 1 \end{bmatrix}$$

The treatment is exogenous if $\rho_{1t} = 0$. Note that we did not specify the structure of the correlations between the potential-outcome errors. We do not need information about these correlations to estimate POMs and treatment effects because all covariates and the outcome are observed in observations from each group.

From here, we discuss a model with an ordinal endogenous treatment. The results for binary treatment models are similar.

As in *Binary and ordinal endogenous covariates*, using the results from *Likelihood for multiequation models* in [ERM] **eprobit**, we can write the joint density of y_i and t_i using the conditional density of the treatment error ϵ_{ti} on the outcome errors $\epsilon_{i1}, \ldots, \epsilon_{Ti}$.

Define

$$r_i = y_i - \mathbf{x}_i \boldsymbol{\beta}_j \quad \text{if} \quad t_i = v_j$$

The log likelihood for the model is

$$\ln L = \sum_{i=1}^{N} w_i \ln \left\{ \Phi_1^* \left(l_{ti} - \frac{\rho_{1t}}{\sigma} r_i, u_{ti} - \frac{\rho_{1t}}{\sigma} r_i, 1 - \rho_{1t}^2 \right) \phi \left(r_i, \sigma^2 \right) \right\}$$

where l_{ti} and u_{ti} are the limits for the treatment probability given in *Treatment* in [ERM] **eprobit**.

The treatment effect $y_{ji} - y_{1i}$ is the difference in the outcome for individual i if the individual receives the treatment $t_i = v_j$ and what the difference would have been if the individual received the control treatment $t_i = v_1$ instead.

The conditional POM for treatment group j is

$$\text{POM}_j(\mathbf{x}_i) = E(y_{ji}|\mathbf{x}_i) = \mathbf{x}_i \boldsymbol{\beta}_j$$

For treatment group j, the treatment effect (TE) conditioned on $\mathbf{x}_i$ is

$$\text{TE}_j(\mathbf{x}_i) = E(y_{ji} - y_{1i}|\mathbf{x}_i) = \text{POM}_j(\mathbf{x}_i) - \text{POM}_1(\mathbf{x}_i)$$

For treatment group j, the treatment effect on the treated (TET) in group h for covariates $\mathbf{x}_i$ is

$$\begin{aligned}
\text{TET}_j(\mathbf{x}_i, t_i = v_h) &= E(y_{ji} - y_{1i}|\mathbf{x}_i, t_i = v_h) \\
&= \mathbf{x}_i \boldsymbol{\beta}_j - \mathbf{x}_i \boldsymbol{\beta}_1 + E(\epsilon_{ji}|\mathbf{x}_i, t_i = v_h) - E(\epsilon_{1i}|\mathbf{x}_i, t_i = v_h)
\end{aligned}$$

Remembering that the outcome errors and the treatment error ϵ_{ti} are multivariate normal, for $j = 1, \ldots, T$, we can decompose ϵ_{ji} such that

$$\epsilon_{ji} = \sigma \rho_{1t} \epsilon_{ti} + \psi_{ji}$$

where ψ_{ji} has mean 0.

It follows that

$$\text{TET}_j(\mathbf{x}_i, t_i = v_h) = \mathbf{x}_i \boldsymbol{\beta}_j - \mathbf{x}_i \boldsymbol{\beta}_1$$

We can take the expectation of these conditional predictions over the covariates to get population average parameters. The `estat teffects` or `margins` command is used to estimate the expectations as predictive margins once the model is estimated with `eregress`. The POM for treatment group j is

$$\text{POM}_j = E(y_{ji}) = E\{\text{POM}_j(\mathbf{x}_i)\}$$

The ATE for treatment group j is

$$\text{ATE}_j = E(y_{ji} - y_{1i}) = E\{\text{TE}_j(\mathbf{x}_i)\}$$

For treatment group j, the average treatment effect on the treated (ATET) in treatment group h is

$$\text{ATET}_{jh} = E\left(y_{ji} - y_{1i}|t_i = v_h\right) = E\left\{\text{TET}_j(\mathbf{x}_i, t_i = v_h)|t_i = v_h\right\}$$

The conditional mean of y_i at treatment level v_j is

$$E(y_i|\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_j) = \mathbf{x}_i\boldsymbol{\beta}_j + E(\epsilon_i|\mathbf{x}_i, \mathbf{z}_{ti}, t_i = v_j)$$

In *Predictions using the full model* in [ERM] **eprobit postestimation**, we discuss how the conditional mean of ϵ_i is calculated.

If the variance and correlation parameters are potential-outcome specific, for $j = 1, \ldots, T$, ϵ_{ji} and ϵ_{ti} are bivariate normal with mean 0 and covariance

$$\boldsymbol{\Sigma}_j = \begin{bmatrix} \sigma_j^2 & \sigma_j\rho_{jt} \\ \sigma_j\rho_{jt} & 1 \end{bmatrix}$$

Now define

$$\rho_i = \sum_{j=1}^{T} 1(t_i = v_j)\rho_{jt}$$

$$\sigma_i = \sum_{j=1}^{T} 1(t_i = v_j)\sigma_j$$

The log likelihood for the model is

$$\ln L = \sum_{i=1}^{N} w_i \ln\left\{\Phi_1^*\left(l_{ti} - \frac{\rho_i}{\sigma_i}r_i, u_{ti} - \frac{\rho_i}{\sigma_i}r_i, 1 - \rho_i^2\right)\phi\left(r_i, \sigma_i^2\right)\right\}$$

The definitions for the potential-outcome means and treatment effects are the same as in the case where the variance and correlation parameters did not vary by potential outcome. For the treatment effect on the treated (TET) of group j in group h, we have

$$\text{TET}_j(\mathbf{x}_i, t_i = v_h) = E\left(y_{ji} - y_{1i}|\mathbf{x}_i, t_i = v_h\right)$$
$$= \mathbf{x}_i\boldsymbol{\beta}_j - \mathbf{x}_i\boldsymbol{\beta}_1 + E\left(\epsilon_{ji}|\mathbf{x}_i, t_i = v_h\right) - E\left(\epsilon_{1i}|\mathbf{x}_i, t_i = v_h\right)$$

The outcome errors and the treatment error ϵ_{ti} are multivariate normal, so for $j = 1, \ldots, T$, we can decompose ϵ_{ji} such that

$$\epsilon_{ji} = \sigma_j\rho_j\epsilon_{ti} + \psi_{ji}$$

where ψ_{ji} has mean 0 and is independent of t_i.

It follows that

$$\text{TET}_j(\mathbf{x}_i, t_i = v_h) = E\left(y_{ji} - y_{1i}|\mathbf{x}_i, t_i = v_h\right)$$
$$= \mathbf{x}_i\boldsymbol{\beta}_j - \mathbf{x}_i\boldsymbol{\beta}_1 + (\sigma_j\rho_j - \sigma_1\rho_1)E\left(\epsilon_{ti}|\mathbf{x}_i, t_i = v_h\right)$$

The mean of ϵ_{ti} conditioned on t_i and the exogenous covariates $\mathbf{x}_i$ can be determined using the formulas discussed in *Predictions using the full model* in [ERM] **eprobit postestimation**. It is nonzero. So the treatment effect on the treated will be equal only to the treatment effect under an exogenous treatment or when the correlation and variance parameters are identical between the potential outcomes.

As in the other case, we can take the expectation of these conditional predictions over the covariates to get population-averaged parameters. The `estat teffects` or `margins` command is used to estimate the expectations as predictive margins once the model is fit with `eregress`.

Endogenous sample selection

Probit endogenous sample selection

A linear regression for outcome y_i with selection on s_i has the form

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \epsilon_i > 0$$
$$s_i = 1\left(\mathbf{z}_{si}\boldsymbol{\alpha}_s + \epsilon_{si} > 0\right)$$

where $\mathbf{x}_i$ are covariates that affect the outcome and $\mathbf{z}_{si}$ are covariates that affect selection. The outcome y_i is observed if $s_i = 1$ and is not observed if $s_i = 0$. The unobserved errors ϵ_i and ϵ_{si} are normal with mean 0 and covariance

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma^2 & \sigma\rho_{1s} \\ \sigma\rho_{1s} & 1 \end{bmatrix}$$

As in the previous section, using the results from *Likelihood for multiequation models* in [ERM] **eprobit**, we can write the joint density of y_i and s_i using the conditional density of the selection error ϵ_{si} on the outcome error ϵ_i.

For the selection indicator s_i, we have lower and upper limits

$$l_{si} = \begin{cases} -\infty & s_i = 0 \\ -\mathbf{z}_{si}\boldsymbol{\alpha}_s - \frac{\rho_{1s}}{\sigma}\left(y_i - \mathbf{x}_i\boldsymbol{\beta}\right) & s_i = 1 \end{cases} \qquad u_{si} = \begin{cases} -\mathbf{z}_{si}\boldsymbol{\alpha}_s & s_i = 0 \\ \infty & s_i = 1 \end{cases}$$

The log likelihood for the model is

$$\ln L = \sum_{i=1}^{N} w_i \ln \Phi_1^*\left(l_{si}, u_{si}, 1 - s_i\rho_{1s}^2\right) + \sum_{i \in S} w_i \ln \phi\left(y_i - \mathbf{x}_i\boldsymbol{\beta}, \sigma^2\right)$$

where S is the set of observations for which y_i is observed.

The conditional mean of y_i is
$$E(y_i|\mathbf{x}_i) = \mathbf{x}_i\boldsymbol{\beta}$$

Tobit endogenous sample selection

Instead of constraining the selection indicator to be binary, tobit endogenous sample selection uses a censored continuous sample-selection indicator. We allow the selection variable to be left-censored or right-censored.

A linear regression model for outcome y_i with tobit selection on s_i has the form

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \epsilon_i > 0$$

We observe the selection indicator s_i, which indicates the censoring status of the latent selection variable $s_i^\star$,

$$s_i^\star = \mathbf{z}_{si}\boldsymbol{\alpha}_s + \epsilon_{si}$$

$$s_i = \begin{cases} l_i & s_i^\star \leq l_i \\ s_i^\star & l_i < s_i^\star < u_i \\ u_i & s_i^\star \geq u_i \end{cases}$$

where $\mathbf{z}_{si}$ are covariates that affect selection and l_i and u_i are fixed lower and upper limits.

The outcome y_i is observed when $s_i^\star$ is not censored ($l_i < s_i^\star < u_i$). The outcome y_i is not observed when $s_i^\star$ is left-censored ($s_i^\star \leq l_i$) or $s_i^\star$ is right-censored ($s_i^\star \geq u_i$). The unobserved errors ϵ_i and ϵ_{si} are normal with mean 0 and covariance

$$\begin{bmatrix} \sigma^2 & \sigma_{1s} \\ \sigma_{1s} & \sigma_s^2 \end{bmatrix}$$

For the selected observations, we can treat s_i as a continuous endogenous regressor, as in *Continuous endogenous covariates*. In fact, s_i may even be used as a regressor for y_i in eregress (specify tobitselect(... main)). On the nonselected observations, we treat s_i like the probit sample-selection indicator in *Probit endogenous sample selection*.

The log likelihood is

$$\begin{aligned} \ln L = & \sum_{i \in S} w_i \ln\phi_2 \left(y_i - \mathbf{x}_i\boldsymbol{\beta}, s_i - \mathbf{z}_{si}\boldsymbol{\alpha}_s, \boldsymbol{\Sigma} \right) \\ & + \sum_{i \in L} w_i \ln\Phi_1^*(l_{li}, u_{li}, 1) \\ & + \sum_{i \in U} w_i \ln\Phi_1^*(l_{ui}, u_{ui}, 1) \end{aligned}$$

where S is the set of observations for which y_i is observed, L is the set of observations where $s_i^\star$ is left-censored, and U is the set of observations where $s_i^\star$ is right-censored. The lower and upper limits for selection—l_{li}, u_{li}, l_{ui}, and u_{ui}—are defined in *Tobit endogenous sample selection* in [ERM] **eprobit**.

When s_i is not a covariate in $\mathbf{x}_i$, we use the standard conditional mean formula,

$$E(y_i|\mathbf{x}_i) = \mathbf{x}_i\boldsymbol{\beta}$$

Otherwise, we use

$$E(y_i|\mathbf{x}_i, s_i, z_{si}) = \mathbf{x}_i\boldsymbol{\beta} + \frac{\sigma_{1s}}{\sigma_s^2}(s_i - z_{si}\boldsymbol{\alpha}_s)$$

Random effects

For a linear regression with random effects, we observe panel data. For panel $i = 1, \ldots, N$ and observation $j = 1, \ldots, N_i$, a linear regression of outcome y_{ij} on covariates $\mathbf{x}_{ij}$ may be written as

$$y_{ij} = \mathbf{x}_{ij}\boldsymbol{\beta} + \epsilon_{ij} + u_i$$

The random effect u_i is normal with mean 0 and variance σ_u^2. It is independent of the observation-level error ϵ_{ij}, which is normal with mean 0 and variance σ^2.

We derive the likelihood by using the conditional density of y_{ij} on the random effect u_i and the marginal density of u_i. Multiplying them together, we have the joint density, which is integrated over u_i.

Let

$$l_{ij}(u) = \phi\big(y_{ij} - \mathbf{x}_{ij}\boldsymbol{\beta} - u, \sigma^2\big)$$

The likelihood for panel i is

$$L_i = \int_{-\infty}^{\infty} \phi\left(\frac{u_i}{\sigma_u}\right) \prod_{j=1}^{N_i} l_{ij}(u_i) du_i$$

We can approximate this integral using Gauss–Hermite quadrature. For q-point Gauss–Hermite quadrature, let the abscissa and weight pairs be denoted by (a_{ki}, w_{ki}), $k = 1, \ldots, q$. The Gauss–Hermite quadrature approximation is then

$$\int_{-\infty}^{\infty} f(x) \exp(-x^2)\, dx \approx \sum_{k=1}^{q} w_{ki} f(a_{ki})$$

The default approximation used by xteregress is mean–variance adaptive Gauss–Hermite quadrature. This chooses optimal abscissa and weights for each panel. See *Likelihood for multiequation models* in [ERM] **eprobit** for more information on the use of mean–variance adaptive Gauss–Hermite quadrature.

Using the quadrature approximation, the log likelihood is

$$\ln L = \sum_{i=1}^{N} \ln \left\{ \sum_{k=1}^{q} w_{ki} \prod_{j=1}^{N_i} l_{ij}(\sigma_u a_{ki}) \right\}$$

The conditional mean of y_{ij} is

$$E(y_{ij}|\mathbf{x}_{ij}) = \mathbf{x}_{ij}\boldsymbol{\beta}$$

Combinations of features

Extended linear regression models that involve multiple features can be formulated using the techniques discussed in *Likelihood for multiequation models* in [ERM] **eprobit**. Essentially, the density of the observed endogenous covariates can be written in terms of the unobserved normal errors. The observed endogenous and exogenous covariates determine the range of the errors, and the joint density can be evaluated as multivariate normal probabilities and densities.

Confidence intervals

The estimated variances will always be nonnegative, and the estimated correlations will always fall in $(-1, 1)$. To obtain confidence intervals that accommodate these ranges, we must use transformations.

We use the log transformation to obtain the confidence intervals for variance parameters and the atanh transformation to obtain confidence intervals for correlation parameters. For details, see *Confidence intervals* in [ERM] **eprobit**.

References

Amemiya, T. 1985. *Advanced Econometrics*. Cambridge, MA: Harvard University Press.

Baltagi, B. H. 2013. *Econometric Analysis of Panel Data*. 5th ed. Chichester, UK: Wiley.

Bartus, T., and D. Roodman. 2014. Estimation of multiprocess survival models with cmp. *Stata Journal* 14: 756–777.

Drukker, D. M. 2016. A generalized regression-adjustment estimator for average treatment effects from panel data. *Stata Journal* 16: 826–836.

Elbakidze, L., L. Lu, and S. Eigenbrode. 2011. Evaluating vector-virus-yield interactions for peas and lentils under climatic variability: A limited dependent variable analysis. *Journal of Agricultural and Resource Economics* 36: 504–520. https://doi.org/10.22004/ag.econ.119177.

Heckman, J. J. 1976. The common structure of statistical models of truncation, sample selection and limited dependent variables and a simple estimator for such models. *Annals of Economic and Social Measurement* 5: 475–492.

———. 1978. Dummy endogenous variables in a simultaneous equation system. *Econometrica* 46: 931–959. https://doi.org/10.2307/1909757.

———. 1979. Sample selection bias as a specification error. *Econometrica* 47: 153–161. https://doi.org/10.2307/1912352.

Keshk, O. M. G. 2003. Simultaneous equations models: What are they and how are they estimated. Program in Statistics and Methodology, Department of Political Science, Ohio State University. https://polisci.osu.edu/sites/polisci.osu.edu/files/Simultaneous Equations.pdf.

Maddala, G. S. 1983. *Limited-Dependent and Qualitative Variables in Econometrics*. Cambridge: Cambridge University Press.

Maddala, G. S., and L.-F. Lee. 1976. Recursive models with qualitative endogenous variables. *Annals of Economic and Social Measurement* 5: 525–545.

Maitra, C., and P. Rao. 2014. An empirical investigation into measurement and determinants of food security in slums of Kolkata. School of Economics Discussion Paper No. 531, School of Economics, University of Queensland. espace.library.uq.edu.au/view/UQ:352184.

Roodman, D. 2011. Fitting fully observed recursive mixed-process models with cmp. *Stata Journal* 11: 159–206.

White, H. L., Jr. 1996. *Estimation, Inference and Specification Analysis*. Cambridge: Cambridge University Press.

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

———. 2014. Quasi-maximum likelihood estimation and testing for nonlinear models with endogenous explanatory variables. *Journal of Econometrics* 182: 226–234. https://doi.org/10.1016/j.jeconom.2014.04.020.

———. 2020. *Introductory Econometrics: A Modern Approach*. 7th ed. Boston: Cengage.

Also see

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

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

[ERM] **predict advanced** — predict's advanced features

[ERM] **predict treatment** — predict for treatment statistics

[ERM] **estat teffects** — Average treatment effects for extended regression models

[ERM] **Intro 9** — Conceptual introduction via worked example

[CAUSAL] **etregress** — Linear regression with endogenous treatment effects

[R] **heckman** — Heckman selection model

[R] **ivregress** — Single-equation instrumental-variables regression

[R] **regress** — Linear regression

[SVY] **svy estimation** — Estimation commands for survey data

[XT] **xtheckman** — Random-effects regression with sample selection

[XT] **xtreg** — Fixed-, between-, and random-effects and population-averaged linear models[+]

[XT] **xtivreg** — Instrumental variables and two-stage least squares for panel-data models

[U] **20 Estimation and postestimation commands**

Title

> **eregress postestimation** — Postestimation tools for eregress and xteregress

Postestimation commands

The following postestimation command is of special interest after `eregress` and `xteregress`:

Command	Description
estat teffects	treatment effects and potential-outcome means

The following standard postestimation commands are also available after `eregress` and `xteregress`:

Command	Description
contrast	contrasts and ANOVA-style joint tests of estimates
estat ic	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC)
estat summarize	summary statistics for the estimation sample
estat vce	variance–covariance matrix of the estimators (VCE)
† estat (svy)	postestimation statistics for survey data
estimates	cataloging estimation results
etable	table of estimation results
* forecast	dynamic forecasts and simulations
* hausman	Hausman's specification test
lincom	point estimates, standard errors, testing, and inference for linear combinations of coefficients
* lrtest	likelihood-ratio test
margins	marginal means, predictive margins, marginal effects, and average marginal effects
marginsplot	graph the results from margins (profile plots, interaction plots, etc.)
nlcom	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
predict	means, probabilities, treatment effects, etc.
predictnl	point estimates, standard errors, testing, and inference for generalized predictions
pwcompare	pairwise comparisons of estimates
† suest	seemingly unrelated estimation
test	Wald tests of simple and composite linear hypotheses
testnl	Wald tests of nonlinear hypotheses

*`forecast`, `hausman`, and `lrtest` are not appropriate with `svy` estimation results.

†`suest` and the survey data `estat` commands are not available after `xteregress`.

predict

Predictions after `eregress` and `xteregress` are described in

[ERM] **eregress predict**	predict after eregress
[ERM] **predict treatment**	predict for treatment statistics
[ERM] **predict advanced**	predict's advanced features

[ERM] **eregress predict** describes the most commonly used predictions. If you fit a model with treatment effects, predictions specifically related to these models are detailed in [ERM] **predict treatment**. [ERM] **predict advanced** describes less commonly used predictions, such as predictions of outcomes in auxiliary equations.

margins

Description for margins

`margins` estimates statistics based on fitted models. These statistics include marginal means, marginal probabilities, potential-outcome means, average and conditional derivatives, average and conditional effects, and treatment effects.

Menu for margins

Statistics > Postestimation

Syntax for margins

margins [*marginlist*] [, *options*]

margins [*marginlist*] , <u>predict</u>(*statistic* ...) [<u>predict</u>(*statistic* ...) ...] [*options*]

statistic	Description
Main	
<u>mean</u>	mean; the default
pr	probability for binary or ordinal y_j
<u>pomean</u>	potential-outcome mean
te	treatment effect
tet	treatment effect on the treated
xb	linear prediction excluding all complications
pr(a,b)	$\Pr(a < y_j < b)$ for continuous y_j
e(a,b)	$E(y_j \mid a < y_j < b)$ for continuous y_j
<u>ystar</u>(a,b)	$E(y_j^*)$, $y_j^* = \max\{a, \min(y_j, b)\}$ for continuous y_j
<u>expmean</u>	calculate $E\{\exp(y_i)\}$

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

For the full syntax, see [R] **margins**.

Remarks and examples

See [ERM] **Intro 7** for an overview of using margins and predict after eregress. For examples using margins, predict, and estat teffects, see *Interpreting effects* in [ERM] **Intro 9** and see [ERM] **Example 1a**.

Methods and formulas

This section contains methods and formulas for predictions and inference for the default average structural function. Methods and formulas for all other predictions are given in *Methods and formulas* of [ERM] **eregress**. We begin with the cross-sectional model and then extend our discussion to the random effect models that we use for panel data.

In the linear regression model, for exogenous covariates $\mathbf{x}_i$ and C endogenous regressors $\mathbf{w}_i$, we have

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{w}_i\boldsymbol{\beta}_2 + \epsilon_i$$

where the error ϵ_i is normal and correlated with $\mathbf{w}_i$.

Because ϵ_i is a normally distributed, mean 0, random variable, we can split it into two mean 0, normally distributed, independent parts,

$$\epsilon_i = \boldsymbol{\gamma}\boldsymbol{\epsilon}_{2i} + \psi_i$$

where $\boldsymbol{\epsilon}_{2i}$ is the unobserved heterogeneity that gives rise to the endogeneity and ψ_i is an idiosyncratic error term with variance σ_ψ^2.

Conditional on the covariates and the unobserved heterogeneity, the conditional mean of y_i is

$$E(y_i|\mathbf{x}_i, \mathbf{w}_i, \epsilon_i) = \mathbf{x}_i\boldsymbol{\beta} + \mathbf{w}_i\boldsymbol{\beta}_2 + \boldsymbol{\gamma}\boldsymbol{\epsilon}_{2i}$$

Predictions and effects are computed based on the expression above. Including $\boldsymbol{\epsilon}_{2i}$ controls for endogeneity. Thus, all effects computed using the expression above have a structural interpretation. See Imbens and Newey (2009) and Wooldridge (2010) for a detailed description of structural functions for models with endogeneity.

Our discussion easily extends to models for panel data with random effects. In this case, we have N panels. Panel $i = 1, \ldots, N$ has observations $t = 1, \ldots, N_i$, so we observe y_{it} with random effect α_i and observation-level error ϵ_{it}. These errors are independent of each other. So the combined error $\xi_{it} = \alpha_i + \epsilon_{it}$ is normal with mean 0 and variance $\sigma^2 + \sigma_\alpha^2$, where σ_α^2 is the variance of α_i. The results discussed earlier can then be applied using the combined error ξ_{it} rather than the cross-sectional error.

All predictions after xteregress assume the panel-level random effects (α_i) are zero. Put another way, predictions condition on the random effects being set to their means.

References

Imbens, G. W., and W. K. Newey. 2009. Identification and estimation of triangular simultaneous equations models without additivity. *Econometrica* 77: 1481–1512. https://doi.org/10.3982/ECTA7108.

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

Also see

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

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

[ERM] **predict treatment** — predict for treatment statistics

[ERM] **predict advanced** — predict's advanced features

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

[U] **20 Estimation and postestimation commands**

Title

---

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

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

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

        xteregress y x1 ..., ...

Then the main equation would be

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

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, $e_{ij}.\mathbf{y} = u_i.\mathbf{y} + v_{ij}.\mathbf{y}$.

All predictions after `xteregress` assume the panel-level random effects ($u_i.\mathbf{y}$) 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	
<u>mean</u>	linear prediction; the default
xb	linear prediction excluding all complications
<u>exp</u>mean	expected value of exponential of the mean; $E\{\exp(\text{mean})\}$

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

counterfactual	Description
Main	
<u>tar</u>get(*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 $\beta_0 + \beta_1 \mathtt{x1}_i + \cdots$. Use of target() is discussed in *Remarks and examples* below.

Remarks and examples

Remarks are presented under the following headings:

How to think about the model you fit
The default asf mean calculation for predictions
The fixedasf calculation for predictions

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 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 y_i given `x1` and `x2`.

2. The expected values of 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 $e_i.\texttt{x1}$ by 1 too.

What does it mean to increase $e_i.\texttt{x1}$? 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 $e.\texttt{x1}$ would have passed along its effect to the $e.\texttt{y}$ because of the correlation. The new value of `y` would be the direct effect of `x1` in the `y` equation plus the change in $e.\texttt{y}$.

`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 $e.\texttt{x1}$ 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 $e_i.\texttt{x1}$ 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 $e_i.\texttt{x1}$ and $e.\texttt{y}$ 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, $e.\texttt{x1}$ is correlated with $e.\texttt{y}$. 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 $e.\texttt{x1}$ and $e.\texttt{y}$ 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 $e.\texttt{x1}$ and $e.\texttt{y}$ 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

Title

ERM options — Extended regression model options

Description

This entry describes the options that are common to the extended regression commands; see [ERM] **eregress**, [ERM] **eprobit**, [ERM] **eoprobit**, and [ERM] **eintreg**.

Syntax

erm_cmd ... [, *extensions* *options*]

erm_cmd is one of eregress, eprobit, eoprobit, eintreg, xteregress, xteprobit, xteoprobit, or xteintreg

extensions	Description
Model	
endogenous(*enspec*)	model for endogenous covariates; may be repeated
entreat(*entrspec*)	model for endogenous treatment assignment
extreat(*extrspec*)	exogenous treatment
select(*selspec*)	probit model for selection
tobitselect(*tselspec*)	tobit model for selection

options	Description
Model	
<u>nocons</u>tant	suppress constant term
<u>off</u>set(*varname$_o$*)	include *varname$_o$* in model with coefficient constrained to 1
<u>constr</u>aints(*numlist*)	apply specified linear constraints
SE/Robust	
vce(*vcetype*)	*vcetype* may be oim, <u>r</u>obust, <u>cl</u>uster *clustvar*, opg, <u>boot</u>strap, or <u>jack</u>knife
Reporting	
<u>l</u>evel(*#*)	set confidence level; default is level(95)
<u>nocns</u>report	do not display constraints
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Integration	
<u>intp</u>oints(*#*)	set the number of integration (quadrature) points for integration over four or more dimensions; default is intpoints(128)
<u>triintp</u>oints(*#*)	set the number of integration (quadrature) points for integration over three dimensions; default is triintpoints(10)
<u>reintp</u>oints(*#*)	set the number of integration (quadrature) points for random-effects integration; default is reintpoints(7)
<u>reintm</u>ethod(*intmethod*)	integration method for random effects; *intmethod* may be <u>mvagh</u>ermite (the default) or <u>gh</u>ermite
Maximization	
maximize_options	control the maximization process; seldom used
<u>collin</u>ear	keep collinear variables
coeflegend	display legend instead of statistics

reintpoints() and reintmethod() are available only with xteregress, xteintreg, xteprobit, and xteoprobit.
collinear and coeflegend do not appear in the dialog box.

enspec is *depvars*$_{en}$ = *varlist*$_{en}$ [, *enopts*]

    where *depvars*$_{en}$ is a list of endogenous covariates. Each variable in *depvars*$_{en}$ specifies an endogenous covariate model using the common *varlist*$_{en}$ and options.

entrspec is *depvar*$_{tr}$ [= *varlist*$_{tr}$] [, *entropts*]

    where *depvar*$_{tr}$ is a variable indicating treatment assignment. *varlist*$_{tr}$ is a list of covariates predicting treatment assignment.

extrspec is *tvar* [, *extropts*]

    where *tvar* is a variable indicating treatment assignment.

selspec is *depvar*$_s$ = *varlist*$_s$ [, *selopts*]

    where *depvar*$_s$ is a variable indicating selection status. *depvar*$_s$ must be coded as 0, indicating that the observation was not selected, or 1, indicating that the observation was selected. *varlist*$_s$ is a list of covariates predicting selection.

tselspec is *depvar*$_s$ = *varlist*$_s$ $\left[\,,\ \textit{tselopts}\,\right]$

where *depvar*$_s$ is a continuous variable. *varlist*$_s$ is a list of covariates predicting *depvar*$_s$. The censoring status of *depvar*$_s$ indicates selection, where a censored *depvar*$_s$ indicates that the observation was not selected and a noncensored *depvar*$_s$ indicates that the observation was selected.

enopts	Description
Model	
probit	treat endogenous covariate as binary
oprobit	treat endogenous covariate as ordinal
povariance	estimate a different variance for each level of a binary or an ordinal endogenous covariate
pocorrelation	estimate different correlations for each level of a binary or an ordinal endogenous covariate
nomain	do not add endogenous covariate to main equation
nore	do not include random effects in model for endogenous covariate
noconstant	suppress constant term

povariance is available only with eregress, eintreg, xteregress, and xteintreg.

nore is available only with xteregress, xteintreg, xteprobit, and xteoprobit.

entropts	Description
Model	
povariance	estimate a different variance for each potential outcome
pocorrelation	estimate different correlations for each potential outcome
nomain	do not add treatment indicator to main equation
nocutsinteract	do not interact treatment with cutpoints
nointeract	do not interact treatment with covariates in main equation
nore	do not include random effects in model for endogenous treatment
noconstant	suppress constant term
offset(*varname*$_o$)	include *varname*$_o$ in model with coefficient constrained to 1

povariance is available only with eregress, eintreg, xteregress, and xteintreg.

nocutsinteract is available only with eoprobit.

nore is available only with xteregress, xteintreg, xteprobit, and xteoprobit.

extropts	Description
Model	
povariance	estimate a different variance for each potential outcome
pocorrelation	estimate different correlations for each potential outcome
nomain	do not add treatment indicator to main equation
nocutsinteract	do not interact treatment with cutpoints
nointeract	do not interact treatment with covariates in main equation

povariance is available only with eregress, eintreg, xteregress, and xteintreg.

nocutsinteract is available only with eoprobit.

selopts	Description
Model	
nore	do not include random effects in selection model
noconstant	suppress constant term
<u>off</u>set(*varname_o*)	include *varname_o* in model with coefficient constrained to 1

nore is available only with xteregress, xteintreg, xteprobit, and xteoprobit.

tselopts	Description
Model	
*ll(*varname* | #)	left-censoring variable or limit
*ul(*varname* | #)	right-censoring variable or limit
main	add censored selection variable to main equation
nore	do not include random effects in tobit selection model
noconstant	suppress constant term
<u>off</u>set(*varname_o*)	include *varname_o* in model with coefficient constrained to 1

* You must specify either ll() or ul().

nore is available only with xteregress, xteintreg, xteprobit, and xteoprobit.

Options

> **Model**

endogenous(*depvars*_en = *varlist*_en [, *enopts*]) specifies the model for endogenous covariates. *depvars*_en is a list of one or more endogenous covariates modeled with *varlist*_en. This option may be repeated to allow a different model specification for each endogenous covariate. By default, the endogenous covariates are assumed to be continuous, and a linear Gaussian model is used. Unless the nomain suboption is specified, the variables specified in *depvars*_en are automatically included in the main equation. The following *enopts* are available:

  probit specifies to use a probit model for the endogenous covariates. probit may not be specified with oprobit; however, you may specify endogenous(..., probit) and endogenous(..., oprobit).

  oprobit specifies to use an ordered probit model for the endogenous covariates. oprobit may not be specified with probit; however, you may specify endogenous(..., probit) and endogenous(..., oprobit).

  povariance specifies that different variance parameters be estimated for each level of the endogenous covariates. In a treatment-effects framework, we refer to levels of endogenous covariates as potential outcomes, and povariance specifies that the variance be estimated separately for each potential outcome. povariance may be specified only with eregress and eintreg and with a binary or an ordinal endogenous covariate.

  pocorrelation specifies that different correlation parameters be estimated for each level of the endogenous covariates. In a treatment-effects framework, we refer to levels of endogenous covariates as potential outcomes, and pocorrelation specifies that correlations be estimated separately for each potential outcome. pocorrelation may be specified only with a binary or an ordinal endogenous covariate.

nomain specifies that the endogenous covariate of covariates be excluded from the main model, thus removing the effect. This option is for those who intend to manually construct the effect by adding it to the main model in their own way.

nore specifies that random effects not be included in the equations for the endogenous covariates.

noconstant suppresses the constant term (intercept) in the model for the endogenous covariates.

entreat() and extreat() specify a model for treatment assignment. You may specify only one.

entreat($depvar_{tr}$ $\left[$ = $varlist_{tr}\right]$ $\left[$, $trtopts$ $modopts$ $\right]$) specifies a model for endogenous treatment assignment with $depvar_{tr} = 1$ indicating treatment and $depvar_{tr} = 0$ indicating no treatment. $varlist_{tr}$ are the covariates for the treatment model; they are optional.

extreat($depvar_{tr}$ $\left[$, $trtopts$ $\right]$) specifies a variable that signals exogenous treatment. $depvar_{tr} = 1$ indicates treatment and $depvar_{tr} = 0$ indicates no treatment.

trtopts are

povariance specifies that different variance parameters be estimated for each potential outcome (for each treatment level). povariance may be specified only with eregress and eintreg.

pocorrelation specifies that different correlation parameters be estimated for each potential outcome (for each treatment level).

nomain, nocutsinteract, and nointeract affect the way the treatment enters the main equation.

nomain specifies that the main effect of treatment be excluded from the main equation. Thus, a separate intercept is not estimated for each treatment level. In the case of eoprobit, this means separate cutpoints are not added.

nocutsinteract specifies that instead of the default of having separate cutpoints for each treatment level, you get one set of cutpoints that are shifted by a constant value for each treatment level. This is implemented by placing a separate constant in the main equation for each treatment level. nocutsinteract is available only with eoprobit.

nointeract specifies that the treatment variable not be interacted with the other covariates in the main equation.

These options allow you to customize how the treatment enters the main equation. When nomain and nointeract are specified together, they remove the effect entirely, and you will need to explicitly reintroduce the treatment effect.

modopts are

nore specifies that a random effect not be included in the treatment equation.

noconstant suppresses the constant term (intercept) in the treatment model.

offset($varname_o$) specifies that $varname_o$ be included in the treatment model with the coefficient constrained to 1.

select() and tobitselect() specify a model for endogenous sample selection. You may specify only one.

select($depvar_s$ = $varlist_s$ $\left[$, $modopts$ $\right]$) specifies a probit model for sample selection with $varlist_s$ as the covariates for the selection model. When $depvar_s = 1$, the model's dependent variable is treated as observed (selected); when $depvar_s = 0$, it is treated as unobserved (not selected).

tobitselect(*depvar$_s$* = *varlist$_s$* [, ll(*varname* | #) ul(*varname* | #) main *modopts*]) specifies
 a tobit model for sample selection with *depvar$_s$* as a censored selection variable and *varlist$_s$*
 as the covariates for the selection model.

>   ll(*arg*) specifies that when *depvar$_s$* $\leq$ *arg*, the selection variable is treated as censored and
>   the model's dependent variable is unobserved (not selected).

>   ul(*arg*) specifies that when *depvar$_s$* $\geq$ *arg*, the selection variable is treated as censored and
>   the model's dependent variable is unobserved (not selected).

>   main specifies that the censored selection variable be included as a covariate in the main
>   equation. By default, it is excluded from the main equation.

>   Only the uncensored values of the selection variable contribute to the likelihood through the
>   main equation. Thus, the selection variable participates as though it were uncensored.

>   *modopts* are

>>   nore specifies that a random effect not be included in the selection equation.

>>   noconstant suppresses the constant term (intercept) in the selection model.

>>   offset(*varname$_o$*) specifies that *varname$_o$* be included in the selection model with the
>>   coefficient constrained to 1.

noconstant, offset(*varname$_o$*), constraints(*numlist*); see [R] **Estimation options**.

SE/Robust
---

vce(*vcetype*) specifies the type of standard error reported, which includes types that are derived from
 asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that
 allow for intragroup correlation (cluster *clustvar*), and that use bootstrap or jackknife methods
 (bootstrap, jackknife); see [R] *vce_option*.

Reporting
---

level(#), nocnsreport; see [R] **Estimation options**.

display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels,
 allbaselevels, nofvlabel, fvwrap(#), fvwrapon(*style*), cformat(%*fmt*), pformat(%*fmt*),
 sformat(%*fmt*), and nolstretch; see [R] **Estimation options**.

Integration
---

intpoints(#) and triintpoints(#) control the number of integration (quadrature) points used
 to approximate multivariate normal probabilities in the likelihood and scores.

>   intpoints() sets the number of integration (quadrature) points for integration over four or more
>   dimensions. The number of integration points must be between 3 and 5,000. The default is
>   intpoints(128).

>   triintpoints() sets the number of integration (quadrature) points for integration over three
>   dimensions. The number of integration points must be between 3 and 5,000. The default is
>   triintpoints(10).

>   When four dimensions of integration are used in the likelihood, three will be used in the scores. The
>   algorithm for integration over four or more dimensions differs from the algorithm for integration
>   over three dimensions.

reintpoints(*#*) and reintmethod(*intmethod*) control how the integration of random effects is numerically calculated.

    reintpoints() sets the number of integration (quadrature) points used for integration of the random effects. The default is intpoints(7). Increasing the number increases accuracy but also increases computational time. Computational time is roughly proportional to the number specified. See *Likelihood for multiequation models* in [ERM] **eprobit** for more details.

    reintmethod() specifies the integration method. The default method is mean–variance adaptive Gauss–Hermite quadrature, reintmethod(mvaghermite). We recommend this method. reintmethod(ghermite) specifies that nonadaptive Gauss–Hermite quadrature be used. This method is less computationally intensive and less accurate. It is sometimes useful to try reintmethod(ghermite) to get the model to converge and then perhaps use the results as initial values specified in option from when fitting the model using the more accurate int-method(mvaghermite). See *Likelihood for multiequation models* in [ERM] **eprobit** for more details.

> ⌐ Maximization ̄

maximize_options: <u>difficult</u>, <u>tech</u>nique(*algorithm_spec*), <u>iter</u>ate(*#*), [no]log, <u>trace</u>, gradient, showstep, <u>hess</u>ian, <u>showtol</u>erance, <u>tol</u>erance(*#*), <u>ltol</u>erance(*#*), <u>nrtol</u>erance(*#*), nonrtolerance, and from(*init_specs*); see [R] **Maximize**.

The default technique for eintreg, eoprobit, eprobit, and eregress is technique(nr). The default technique for xteintreg, xteoprobit, xteprobit, and xteregress is technique(bhhh 10 nr 2).

Setting the optimization type to technique(bhhh) resets the default *vcetype* to vce(opg).

The following option is available with *erm_cmd* but is not shown in the dialog box:

collinear, coeflegend; see [R] **Estimation options**.

Also see

[ERM] **eintreg** — Extended interval regression

[ERM] **eoprobit** — Extended ordered probit regression

[ERM] **eprobit** — Extended probit regression

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

Title

> **estat teffects** — Average treatment effects for extended regression models

Description

estat teffects estimates the average treatment effect, average treatment effect on the treated, and potential-outcome mean for ERMs.

Menu

Statistics > Postestimation

Syntax

estat teffects [, *options*]

options	Description
ate	estimate average treatment effect; the default
atet	estimate average treatment effect on the treated
pomean	estimate potential-outcome mean
tlevel(*numlist*)	calculate treatment effects or potential-outcome means for specified treatment levels
outlevel(*numlist*)	calculate treatment effects or potential-outcome means for specified levels of ordinal dependent variable
subpop(*subspec*)	estimate for subpopulation
level(#)	set confidence level; default is level(95)
display_options	control columns and column formats, row spacing, line width and factor-variable labeling

collect is allowed; see [U] **11.1.10 Prefix commands**.

Options

ate estimates the average treatment effect (ATE). This is the default.

atet estimates the average treatment effect on the treated (ATET). For binary treatments, the ATET is reported for the treated group subpopulation. For ordinal treatments, by default, the ATET is reported for the first noncontrol treatment group subpopulation. You can use the subpop() option to calculate the ATET for a different treatment group.

pomean estimates the potential-outcome mean (POM).

tlevel(*numlist*) specifies the treatment levels for which treatment effects or POMs are calculated. By default, the treatment effects are computed for all noncontrol treatment levels, and the POMs are computed for all treatment levels.

outlevel(*numlist*) specifies the levels of the ordinal dependent variable for which treatment effects or POMs are to be calculated. By default, treatment effects or POMs are computed for all levels of the ordinal dependent variable. This option is only available after eoprobit and xteoprobit.

subpop([*varname*] [*if*]) specifies the subpopulation for which the ATE, ATET, and POM are calculated. The subpopulation is identified by the indicator variable, by the if expression, or by both. A 0 indicates that the observation be excluded, a nonzero indicates that it be included, and a missing value indicates that it be treated as outside of the population (and thus ignored). For instance, for an ordinal treatment trtvar with levels 1, 2, and 3, you can specify subpop(if trtvar==3) to obtain the ATETs for trtvar = 3.

level(*#*) specifies the confidence level, as a percentage, for confidence intervals. The default is level(95) or as set by set level; see [U] **20.8 Specifying the width of confidence intervals**.

display_options: noci, nopvalues, vsquish, nofvlabel, fvwrap(*#*), fvwrapon(*style*), cformat(%*fmt*), pformat(%*fmt*), sformat(%*fmt*), and nolstretch.

noci suppresses confidence intervals from being reported in the coefficient table.

nopvalues suppresses p-values and their test statistics from being reported in the coefficient table.

vsquish specifies that the blank space separating factor-variable terms or time-series–operated variables from other variables in the model be suppressed.

nofvlabel displays factor-variable level values rather than attached value labels. This option overrides the fvlabel setting; see [R] **set showbaselevels**.

fvwrap(*#*) allows long value labels to wrap the first *#* lines in the coefficient table. This option overrides the fvwrap setting; see [R] **set showbaselevels**.

fvwrapon(*style*) specifies whether value labels that wrap will break at word boundaries or break based on available space.

fvwrapon(word), the default, specifies that value labels break at word boundaries.

fvwrapon(width) specifies that value labels break based on available space.

This option overrides the fvwrapon setting; see [R] **set showbaselevels**.

cformat(%*fmt*) specifies how to format estimates, standard errors, and confidence limits in the estimates table. The maximum format width is 9.

pformat(%*fmt*) specifies how to format p-values in the estimates table. The maximum format width is 5.

sformat(%*fmt*) specifies how to format test statistics in the estimates table. The maximum format width is 8.

nolstretch specifies that the width of the estimates table not be automatically widened to accommodate longer variable names. The default, lstretch, is to automatically widen the estimates table up to the width of the Results window. Specifying lstretch or nolstretch overrides the setting given by set lstretch. If set lstretch has not been set, the default is lstretch. nolstretch is not shown in the dialog box.

Remarks and examples

`estat teffects` estimates ATEs, ATETs, and POMs after extended regression commands. These are calculated as means of predictions by using margins on the predictions from `predict` after the extended regression commands. If the ERM command reported robust standard errors, `estat teffects` reports unconditional standard errors so that inference is for the population effect instead of the sample effect. See *Unconditional standard errors* in [R] **margins** for more information.

See [ERM] **Intro 9** for an example using `estat teffects`. Methods and formulas for treatment-effect estimation are given in *Methods and formulas* of [ERM] **eprobit**, [ERM] **eoprobit**, [ERM] **eregress**, and [ERM] **eintreg**.

Stored results

`estat teffects` stores the following in `r()`:

Macros
`r(vce)`	*vcetype* specified in `vce()`
`r(vcetype)`	title used to label Std. err.
`r(clustvar)`	name of cluster variable

Matrices
`r(b)`	estimates
`r(V)`	variance–covariance matrix of the estimates
`r(table)`	matrix containing the estimates with their standard errors, test statistics, p-values, and confidence intervals

Also see

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

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

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

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

Title

Example 1a — Linear regression with continuous endogenous covariate

---

Description

In this example, we show how to estimate and interpret the results of an extended regression model with a continuous outcome and continuous endogenous covariate.

Remarks and examples

The fictional State University is studying the relationship between the high school grade point average (GPA) of the students it admits and their final college GPA. They suspect that unobserved ability affects both high school GPA and college GPA. Thus, high school GPA is an endogenous covariate.

Using data on the 2,500 students in the cohort expected to graduate in 2010, the researchers at State U model college GPA (gpa) as a function of high school GPA (hsgpa). In both cases, GPA is measured in 0.01 increments, and we ignore complications due to the boundary points. We also ignore that, unfortunately, State U has a high dropout rate and college GPA is missing for these students, leaving the researchers with a sample of about 1,500 students.

The State U researchers expect that the effect of high school competitiveness on college GPA is negligible once high school GPA is controlled for. So they include a ranking of the high school (hscomp) as an instrumental covariate for high school GPA. They include parental income measured in $10,000s, which they believe may also influence student performance, in the main model and in the model for high school GPA.

```
. use https://www.stata-press.com/data/r18/class10
(Class of 2010 profile)
. eregress gpa income, endogenous(hsgpa = income i.hscomp)
Iteration 0:  Log likelihood = -638.58598
Iteration 1:  Log likelihood = -638.58194
Iteration 2:  Log likelihood = -638.58194
```

Extended linear regression

Log likelihood = -638.58194

Number of obs = 1,528
Wald chi2(2) = 1167.79
Prob > chi2 = 0.0000

	Coefficient	Std. err.	z	P>|z|	[95% conf. interval]	
gpa						
income	.0575145	.0055174	10.42	0.000	.0467007	.0683284
hsgpa	1.235868	.133686	9.24	0.000	.9738484	1.497888
_cons	-1.217141	.3828614	-3.18	0.001	-1.967535	-.4667464
hsgpa						
income	.0356403	.0019553	18.23	0.000	.0318079	.0394726
hscomp						
Moderate	-.1310549	.0136503	-9.60	0.000	-.1578091	-.1043008
High	-.2331173	.0232712	-10.02	0.000	-.278728	-.1875067
_cons	2.951233	.0164548	179.35	0.000	2.918982	2.983483
var(e.gpa)	.1436991	.0083339			.1282592	.1609977
var(e.hsgpa)	.0591597	.0021403			.05511	.063507
corr(e.hsgpa, e.gpa)	.2642138	.0832669	3.17	0.002	.0948986	.4186724

The estimate of the correlation between the errors from the main and auxiliary equations is 0.26. The z statistic may be used for a Wald test of the null hypothesis that there is no endogeneity. The researchers reject this hypothesis. Because the estimate is positive, they conclude that unobservable factors that increase high school GPA tend to also increase college GPA.

Having satisfied themselves that it is appropriate to account for endogeneity of high school GPA, they examine the coefficient estimates. The estimates for the main equation are interpreted just like those from regress; see [R] **regress**. For example, the researchers expect the difference in college GPA is about 1.24 points for students with a difference of 1 point in high school GPA.

As we discussed in [ERM] **Intro 9**, the coefficients on hsgpa and income in this regression pretty much say everything there is to say about how college GPA changes when either high school GPA or parents' income changes. This is true because our model is linear and we have no interactions. We could make this the end of our story. But it is not the end if we want to ask questions about expected levels of college GPA.

Let's look at a single observation; we will pretend it is for Billy.

```
. generate str name = "Billy" in 537
(2,499 missing values generated)

. list gpa hsgpa income hscomp if name=="Billy"
```

	gpa	hsgpa	income	hscomp
537.	1.03	2	2	High

We have information on Billy's high school competitiveness, hscomp, and his parents' income. With this information, we could form counterfactuals about Billy. We could fix Billy's high school GPA at 2.00, and we could fix his high school GPA at 3.00. We will let margins give us the expected values for college GPA under these two counterfactuals.

```
. margins if name=="Billy", at(hsgpa=(2 3))
warning: prediction constant over observations.

Predictive margins                                        Number of obs = 1
Model VCE: OIM

Expression: Average structural function mean, predict()
1._at: hsgpa = 2
2._at: hsgpa = 3
```

	Margin	Delta-method std. err.	z	P>|z|	[95% conf. interval]	
_at						
1	1.044564	.039223	26.63	0.000	.9676881	1.12144
2	2.280432	.1214604	18.78	0.000	2.042374	2.51849

When we set Billy's high school GPA to 2.00, Billy's expected college GPA is 1.04. Because we did not specify values for the other covariates, margins took them from the observation for Billy. So, more completely and correctly, this is the expected GPA for anyone whose high school GPA is 2.00 and whose parents' income is $20,000 and whose level of high school competitiveness is high.

Why do we care about high school competitiveness when it is not in the main equation? As discussed in [ERM] **Intro 7**, if you want to make inferences that have a structural interpretation with respect to the population (or data-generating process), you must include the level of endogeneity from the equation for high school GPA. In this case, that is an adjustment for Billy's unobserved ability.

When we fix Billy's high school GPA to 3.00, while keeping his parents' income constant at $20,000 and also keeping the adjustment for Billy's unobserved ability constant, we see that Billy's expected college GPA rises to 2.28.

As a sidebar, we note that our first counterfactual of high school GPA at 2.00 is really more of a factual than a counterfactual. Billy's observed GPA in the data is 2.00.

Let's take the next step and estimate the resulting difference in expected college GPA for our two counterfactuals. We just need to add contrast(at(r)) to our margins command.

```
. margins if name=="Billy", at(hsgpa=(2 3)) contrast(at(r) effects nowald)
warning: prediction constant over observations.
```

Contrasts of predictive margins   Number of obs = 1
Model VCE: OIM

Expression: Average structural function mean, predict()
1._at: hsgpa = 2
2._at: hsgpa = 3

		Delta-method				
	Contrast	std. err.	z	P>|z|	[95% conf. interval]	
_at						
(2 vs 1)	1.235868	.133686	9.24	0.000	.9738484	1.497888

The estimated effect, its standard error, and all other associated statistics are identical to the coefficient on `hsgpa` in our `eregress` output.

Would we see anything different if we averaged the effects over the sample to get estimates of the effects in the population? Just remove Billy from the command.

```
. margins, at(hsgpa=(2 3)) contrast(at(r) effects nowald)
```

Contrasts of predictive margins   Number of obs = 1,528
Model VCE: OIM

Expression: Average structural function mean, predict()
1._at: hsgpa = 2
2._at: hsgpa = 3

		Delta-method				
	Contrast	std. err.	z	P>|z|	[95% conf. interval]	
_at						
(2 vs 1)	1.235868	.133686	9.24	0.000	.9738484	1.497888

Not surprisingly, the estimated effect is still 1.24—the same value we have gotten every time, the same value as the coefficient on `hsgpa`. Perhaps more surprisingly, the standard error of the population-average estimate is also the same as the standard error of the coefficient. We do not gain or lose any information when we take an average over an estimate that is constant for all the observations.

In linear models without interactions, we have just seen that the effects are the same for many questions. In nonlinear models, the effects usually differ.

The models in the remaining two examples in this series, [ERM] **Example 1b** and [ERM] **Example 1c**, have exactly the same interpretation we gave to the model in this entry. Adding interval-censoring and endogenous sample selection does not affect the relevant questions or how they are answered.

Video example

Extended regression models: Endogenous covariates

Also see

Title

Example 1b — Interval regression with continuous endogenous covariate

Description

Continuing from [ERM] **Example 1a**, we now consider the case where the dependent variable is interval-censored. We fit this model using `eintreg`.

Remarks and examples

We now assume that, for reasons of confidentiality, the researchers conducting the study do not observe the actual college GPA for those with a GPA below 2.0. For the rest, they are given college GPA only in increments of 0.5 points. So the outcome has both left- and interval-censored observations. The model remains the same.

The lower and upper endpoints for college GPA are stored in `gpal` and `gpau`. Both variables contain a missing value for students who dropped out of college. Other than the change in command name and specification of the dependent variable, the command to fit the model is exactly the same.

```
. eintreg gpal gpau income, endogenous(hsgpa = income i.hscomp)

Iteration 0:  Log likelihood = -1716.9969
Iteration 1:  Log likelihood = -1716.9968
```

Extended interval regression

			Number of obs	=	1,528
			Uncensored =		0
			Left-censored =		150
			Right-censored =		0
			Interval-cens. =		1,378
			Wald chi2(2)	=	912.68
Log likelihood = -1716.9968			Prob > chi2	=	0.0000

	Coefficient	Std. err.	z	P>|z|	[95% conf. interval]	
income	.0551638	.0057859	9.53	0.000	.0438236	.066504
hsgpa	1.111672	.1407083	7.90	0.000	.8358891	1.387456
_cons	-.8180699	.4032468	-2.03	0.042	-1.608419	-.0277207
hsgpa						
income	.0356351	.0019553	18.22	0.000	.0318027	.0394675
hscomp						
Moderate	-.1317151	.0136277	-9.67	0.000	-.1584249	-.1050052
High	-.2320803	.0233633	-9.93	0.000	-.2778715	-.186289
_cons	2.951568	.0164465	179.46	0.000	2.919333	2.983802
var(e.gpal)	.1354248	.0090267			.1188397	.1543245
var(e.hsgpa)	.0591594	.0021403			.0551097	.0635066
corr(e.hsgpa, e.gpal)	.2700108	.0897936	3.01	0.003	.0868241	.4355353

We again find that unobservable factors that increase high school GPA tend to increase college GPA. The parameter estimates here are interpreted just as we did in [ERM] **Example 1a**. In that example, the estimated coefficient on hsgpa was 1.24; here it is 1.11. Like the relationship between regress and intreg, the 1.24 and 1.11 estimate the same parameter, the relationship between hsgpa and the uncensored outcome.

We will not further interpret this model here. Instead we refer you to the interpretation in [ERM] **Example 1a**. The interval-censoring of the dependent variable demonstrated here makes no difference in what commands you would type to answer questions or in how you would interpret the results of those commands. In fact, we encourage you to run the commands discussed in [ERM] **Example 1a** on this model and compare the results.

Because interval regression is a generalization of tobit regression, you can also use eintreg to fit a tobit model with endogenous selection. However, you must convert your dependent variable into interval form. We illustrate how to do this in [ERM] **Intro 8**.

Also see

[ERM] **eintreg** — Extended interval regression

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

[ERM] **Intro 3** — Endogenous covariates features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

> **Example 1c —** Interval regression with endogenous covariate and sample selection

Description

In [ERM] **Example 1a** and [ERM] **Example 1b**, we ignored the observations that were dropped because of missing data on GPA. In this example, we show you how to fit a model that includes a continuous endogenous covariate, a censored outcome, and endogenous sample selection.

Remarks and examples

In the previous two examples, the researchers excluded students who dropped out of college because they are missing college GPA data on these students. So they were estimating parameters for the population of students who graduate from college. Let's suppose they are interested in expected college GPA for all students who enroll, even those who drop out. They suspect that unobserved ability affects both the decision to stay in school and college GPA and thus that they have an endogenously selected sample.

To model the selection, they need a covariate that affects the probability that they observe a student's GPA but does not affect the level of the student's GPA. They include an indicator for whether the student participated in a retention program and whether the student had a roommate who also went to State U. They expect that students with a roommate who went to the same college were more likely to remain in school because they felt more included in the college environment.

```
. eintreg gpal gpau income, endogenous(hsgpa = income i.hscomp)
> select(graduate = hsgpa income i.roommate i.program)
  (iteration log omitted)
```

Extended interval regression					Number of obs	=	2,500
					Selected =		1,528
					Nonselected =		972
					Uncensored =		0
					Left-censored =		150
					Right-censored =		0
					Interval-cens. =		1,378
					Wald chi2(2)	=	734.96
Log likelihood = -2851.3222					Prob > chi2	=	0.0000

	Coefficient	Std. err.	z	P>|z|	[95% conf. interval]	
income	.0338548	.0075484	4.49	0.000	.0190602	.0486495
hsgpa	1.19378	.1443563	8.27	0.000	.9108467	1.476713
_cons	-.7895643	.3908796	-2.02	0.043	-1.555674	-.0234543
graduate						
hsgpa	2.215481	.4411331	5.02	0.000	1.350876	3.080086
income	.1920393	.0162334	11.83	0.000	.1602224	.2238563
roommate						
Yes	.1547087	.0455906	3.39	0.001	.0653528	.2440645
1.program	.4858749	.0523443	9.28	0.000	.383282	.5884678
_cons	-7.524521	1.237529	-6.08	0.000	-9.950034	-5.099008
hsgpa						
income	.047866	.0016981	28.19	0.000	.0445377	.0511942
hscomp						
Moderate	-.1337635	.0115749	-11.56	0.000	-.1564499	-.1110771
High	-.2284481	.0190089	-12.02	0.000	-.2657049	-.1911914
_cons	2.793802	.0132125	211.45	0.000	2.767906	2.819698
var(e.gpal)	.1753568	.0085604			.1593564	.1929636
var(e.hsgpa)	.0685863	.0019399			.0648876	.0724958
corr(e.gra~e, e.gpal)	-.9124422	.0327448	-27.87	0.000	-.9583429	-.8205981
corr(e.hsgpa, e.gpal)	.0534114	.0937195	0.57	0.569	-.1300101	.2332982
corr(e.hsgpa, e.graduate)	.2747613	.0955172	2.88	0.004	.079342	.4498437

The coefficients from the main equation for hsgpa continue to be interpreted as in [ERM] **Example 1b**. Now, however, they are estimates for the population of all admitted students, not the population of all graduates. The estimated effect of high school GPA for this population is slightly higher, 1.19 compared with 1.11.

As with [ERM] **Example 1b**, we will not further interpret this model here. Instead we refer you to the interpretation performed in [ERM] **Example 1a**. The addition of endogenous sample selection makes no difference in what commands you would type to answer questions or to how you would interpret the results of those commands. In fact, we encourage you to run the commands discussed in [ERM] **Example 1a** on this model and compare the results. The only thing to keep in mind is that now the population we are making inferences about is all students admitted to school, not just those who graduate.

Also see

[ERM] **eintreg** — Extended interval regression

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

[ERM] **Intro 3** — Endogenous covariates features

[ERM] **Intro 4** — Endogenous sample-selection features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Description   Remarks and examples   Also see

Description

In this example, we show how to estimate and interpret the results of an extended regression model with a continuous outcome and endogenous binary covariate.

Remarks and examples

Suppose that we want to study the effect of having a college degree on wages. One way to approach the problem is to look at the coefficient on an indicator for whether an individual has a college degree. This gives us an idea of how different the average wage is for individuals with a college degree compared with those without one. However, as in [ERM] **Example 1a**, we suspect that unobserved factors such as ability affect both the probability of graduating from college and wage level. Thus, we need to account for the potential endogeneity of the indicator for having a college degree.

In our fictional study, we collect data on the hourly wages (`wage`) and educational attainment (`college`) of 6,000 adults. We believe that differences in job tenure (`tenure`) and age (`age`) may also affect wages. We can control for these covariates by specifying them in the main equation. We specify `college` in the `endogenous()` option, but this time we also include the `probit` suboption to indicate that the variable is binary. We model graduation as a function of the level of parental education (`peduc`), which we assume does not have a direct effect on wage.

```
. use https://www.stata-press.com/data/r18/wageed
(Wages for 20 to 74 year olds, 2015)

. eregress wage c.age##c.age tenure, endogenous(college = i.peduc, probit)
> vce(robust)

Iteration 0:  Log pseudolikelihood = -18063.148
Iteration 1:  Log pseudolikelihood =   -18060.2
Iteration 2:  Log pseudolikelihood = -18060.164
Iteration 3:  Log pseudolikelihood = -18060.164
```

Extended linear regression

Log pseudolikelihood = -18060.164

Number of obs = 6,000
Wald chi2(4) = 7584.74
Prob > chi2 = 0.0000

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
wage						
age	.4200372	.0163312	25.72	0.000	.3880286	.4520457
c.age#c.age	-.0033523	.0001759	-19.06	0.000	-.003697	-.0030075
tenure	.4921838	.0182788	26.93	0.000	.4563581	.5280095
college						
Yes	5.238087	.1721006	30.44	0.000	4.900776	5.575398
_cons	5.524288	.3428735	16.11	0.000	4.852268	6.196307
college						
peduc						
College	.8605996	.0361723	23.79	0.000	.7897032	.9314959
Graduate	1.361257	.0490862	27.73	0.000	1.26505	1.457465
Doctorate	1.583818	.119513	13.25	0.000	1.349577	1.818059
_cons	-.9731264	.0294779	-33.01	0.000	-1.030902	-.9153508
var(e.wage)	8.99487	.2465919			8.524314	9.491402
corr(e.col~e, e.wage)	.5464027	.0286061	19.10	0.000	.4879055	.600014

The estimated correlation between the errors from the main and auxiliary equations is 0.55 and is significantly different from 0. We conclude that having a college degree is endogenous and that unobservable factors that increase the probability of graduating from college tend to also increase wages.

We find that graduating from college increases the expected wage by $5.24 given a person's age and employment tenure. This estimate is different than comparing the average wages for college graduates and noncollege graduates.

```
. tabulate college, summarize(wage)
```

Indicator for college degree	Summary of Hourly wage Mean	Std. dev.	Freq.
No	17.768516	3.0674174	3,766
Yes	25.520703	5.045888	2,234
Total	20.654913	5.4248886	6,000

The difference in the average wages is \$7.75, but unlike our regression coefficient, that value does not adjust for the different distribution of ages and tenures among college graduates and noncollege graduates.

Another approach to this problem is the potential-outcomes framework. With this approach, we consider the expected wage for each individual without a college degree versus the expected wage for each individual with a college degree. Specifically, we might like to know the average expected change in wages for those who complete college. This is called the average treatment effect on the treated. We consider this approach in [ERM] **Example 2b** and [ERM] **Example 2c**.

[ERM] **Example 2c** also includes an interpretation of how the expected level of income varies by age, tenure, and whether one graduates from college. That analysis could also be applied to this model.

Also see

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

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

[ERM] **estat teffects** — Average treatment effects for extended regression models

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Example 2b — Linear regression with exogenous treatment

Description   Remarks and examples   Also see

Description

In this example, we show how to estimate and interpret the results of an extended regression model with a continuous outcome and exogenous binary treatment.

Remarks and examples

In [ERM] **Example 2a**, we analyzed the effect of having a college degree on wages as a binary endogenous covariate. Now suppose that we approach our research question instead in the potential-outcomes framework. With this approach, we consider the expected wage for each individual without a college degree versus the expected wage for each individual with a college degree. Specifically, we might like to know the average expected change in wages for those who complete college, the average treatment effect on the treated (ATET).

As before, we use `wageed.dta` with educational attainment data on 6,000 adults. We control for differences in job tenure (`tenure`) and age (`age`) by specifying them in the main equation. For the time being, we consider the treatment (`college`) to be exogenous. We want to make inferences about the average effect of a college degree on the wages of all individuals who complete college, not just the subjects in our study sample, so we specify `vce(robust)`. This will allow us to estimate the standard errors of the ATET accounting for the fact the variables in our sample represent just one draw from the population.

```
. use https://www.stata-press.com/data/r18/wageed
(Wages for 20 to 74 year olds, 2015)

. eregress wage c.age##c.age tenure, extreat(college) vce(robust)

Iteration 0:  Log pseudolikelihood = -13989.589
Iteration 1:  Log pseudolikelihood = -13989.586
Iteration 2:  Log pseudolikelihood = -13989.586
```

Extended linear regression

```
                                         Number of obs  =      6,000
                                         Wald chi2(8)   = 439363.91
Log pseudolikelihood = -13989.586        Prob > chi2    =     0.0000
```

wage	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
college# c.age						
No	.2454534	.0180052	13.63	0.000	.2101638	.280743
Yes	.7042756	.0225386	31.25	0.000	.6601007	.7484505
college# c.age#c.age						
No	-.0018998	.0001935	-9.82	0.000	-.002279	-.0015206
Yes	-.0054223	.000243	-22.31	0.000	-.0058986	-.0049459
college# c.tenure						
No	.3206065	.0207164	15.48	0.000	.2800031	.36121
Yes	.4935213	.0257599	19.16	0.000	.4430329	.5440097
college						
No	9.851871	.3701276	26.62	0.000	9.126435	10.57731
Yes	4.384709	.4654545	9.42	0.000	3.472435	5.296983
var(e.wage)	6.20477	.1152627			5.982922	6.434843

Because we specified the command as a treatment-effects model, `eregress` automatically interacts the `college` variable with all other covariates in the model, thus essentially creating separate models for those who graduate from college and those who do not. There is nothing wrong with interpreting the coefficients. This is, after all, just a regression. The coefficients labeled `no` are the estimates of the parameters of the wage model for those who are not college graduates. The coefficients labeled `yes` are the estimates of the parameters for the model of those who are college graduates. Tenure in the company has a larger effect for college graduates than nongraduates. It is 49 cents an hour per tenure year for college graduates and 32 cents for nongraduates. The effect of age is more difficult to interpret because of the quadratic term. The effect of age is clearly different between the groups, but the pattern of that difference is not obvious. See [ERM] **Example 2c** for some tools you could apply to this model that would make that pattern obvious. The effect of college graduation is harder still to see. For any person, it would be the difference of the values predicted by the two models. Again, see [ERM] **Example 2c** for ways to visualize the effect.

If we are interested only in the average effect, we can estimate that using the `estat teffects` command.

```
. estat teffects, atet
```

Predictive margins   Number of obs   = 6,000
                                                      Subpop. no. obs = 2,234

	Margin	Unconditional std. err.	z	P>|z|	[95% conf. interval]	
ATET						
college						
(Yes vs No)	7.62719	.0863465	88.33	0.000	7.457954	7.796426

The average wage is estimated to be \$7.63 higher per hour for the population of college graduates than the wage would have been if those same individuals had not completed college.

We have ignored several potential complications in this example. One of which is that unobserved factors such as ability that influence whether individuals complete college could also influence their wage. In that case, the treatment assignment (obtaining a college degree) would be endogenous. If the treatment were endogenous, we would model its coefficients and the correlation between the treatment assignment errors and the outcome errors. See [ERM] **Example 2c** for an example with an endogenous treatment.

Also see

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

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

[ERM] **estat teffects** — Average treatment effects for extended regression models

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Example 2c — Linear regression with endogenous treatment

Description

Continuing from [ERM] **Example 2b**, we now consider the case where the treatment is endogenous and the variance and correlation parameters differ by treatment group.

Remarks and examples

In [ERM] **Example 2b**, we assumed that graduating from college was an exogenous treatment. However, unobserved factors such as ability may affect whether individuals graduate from college and also affect their wage. Thus, it may be more appropriate for us to treat having a college degree as an endogenous treatment. We found endogeneity in [ERM] **Example 2a**, which analyzes the treatment instead as a binary endogenous covariate. You may want to compare the result of this example with the results from [ERM] **Example 2b**.

Because college graduation is now assumed to be endogenous, we must specify a model for `college`. We model graduation as a function of the level of parental education (`peduc`), which we further assume does not have a direct effect on wage. The endogenous treatment equation is specified in option `entreat()`.

```
. eregress wage c.age##c.age tenure, entreat(college = i.peduc) vce(robust)

Iteration 0:  Log pseudolikelihood = -17382.446
Iteration 1:  Log pseudolikelihood = -17381.922
Iteration 2:  Log pseudolikelihood =  -17381.92
```

Extended linear regression

```
                                              Number of obs =       6,000
                                              Wald chi2(8)  = 348743.60
Log pseudolikelihood = -17381.92              Prob > chi2   =      0.0000
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
wage						
college# c.age						
No	.2338084	.0176633	13.24	0.000	.199189	.2684279
Yes	.6777385	.0219827	30.83	0.000	.6346531	.7208239
college# c.age#c.age						
No	-.0018611	.00019	-9.79	0.000	-.0022335	-.0014887
Yes	-.0052533	.0002372	-22.14	0.000	-.0057183	-.0047883
college# c.tenure						
No	.3948863	.0207452	19.04	0.000	.3542263	.4355462
Yes	.5883544	.0257213	22.87	0.000	.5379415	.6387673
college						
No	10.86301	.3675208	29.56	0.000	10.14268	11.58333
Yes	3.184255	.4612019	6.90	0.000	2.280316	4.088194
college						
peduc						
College	.849575	.0356419	23.84	0.000	.7797181	.9194318
Graduate	1.347272	.0491996	27.38	0.000	1.250843	1.443701
Doctorate	1.541025	.1174797	13.12	0.000	1.310769	1.771281
_cons	-.973061	.0292791	-33.23	0.000	-1.030447	-.9156749
var(e.wage)	7.629807	.2245651			7.202122	8.082889
corr(e.col~e, e.wage)	.623109	.0267317	23.31	0.000	.5679046	.6727326

As in [ERM] **Example 2b**, we can interpret the coefficients in the wage equation as coefficients in separate models for the two potential outcomes—the models for those with and without a college degree. The estimated correlation between the errors from the main and auxiliary equations is 0.62. We could use the z statistic for the correlation to test for endogeneity. We could also use the estat teffects and margins commands to answer questions related to the entire population or specific subpopulations. However, we will not interpret the results of this model any further because we will first extend it.

Above, we assumed that the relationship between the unobserved factors that affect wage and the unobserved factors that affect whether individuals graduate from college was the same for those individuals with a college degree and those without. We do not have a good reason to believe that these will be the same, so we specify the suboption pocorrelation within the option entreat() to model separate correlation parameters for the two potential outcomes. We also assumed that the unobserved factors affecting wage were equally variable for those who had a college degree and

those who did not. We can relax this assumption and model different variances for the two potential outcomes by specifying the suboption `povariance` within the option `entreat()`.

```
. eregress wage c.age##c.age tenure,
> entreat(college = i.peduc, povariance pocorrelation) vce(robust)

Iteration 0:  Log pseudolikelihood = -17382.446
Iteration 1:  Log pseudolikelihood = -17381.327
Iteration 2:  Log pseudolikelihood = -17381.319
Iteration 3:  Log pseudolikelihood = -17381.319

Extended linear regression                         Number of obs  =      6,000
                                                   Wald chi2(8)   = 104887.19
Log pseudolikelihood = -17381.319                  Prob > chi2    =     0.0000
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]
wage						
college#						
c.age						
No	.234277	.0176793	13.25	0.000	.1996261	.2689278
Yes	.6759938	.0220455	30.66	0.000	.6327854	.7192021
college#						
c.age#c.age						
No	-.0018627	.00019	-9.80	0.000	-.0022351	-.0014902
Yes	-.0052427	.0002376	-22.07	0.000	-.0057084	-.0047771
college#						
c.tenure						
No	.3917974	.0211184	18.55	0.000	.350406	.4331887
Yes	.5951107	.0264841	22.47	0.000	.5432027	.6470187
college						
No	10.82487	.3712505	29.16	0.000	10.09723	11.55251
Yes	3.097338	.4678998	6.62	0.000	2.180271	4.014405
college						
peduc						
College	.8482632	.0356294	23.81	0.000	.7784309	.9180955
Graduate	1.343223	.0493492	27.22	0.000	1.2465	1.439945
Doctorate	1.538188	.1162237	13.23	0.000	1.310393	1.765982
_cons	-.9715507	.0292856	-33.18	0.000	-1.028949	-.9141521
var(e.wage)						
college						
No	7.46846	.2657898			6.965275	8.007997
Yes	7.98125	.3990003			7.236315	8.802871
corr(e.col~e,						
e.wage)						
college						
No	.6057846	.0374579	16.17	0.000	.5271994	.6740954
Yes	.6518029	.0359868	18.11	0.000	.5755573	.7168138

We see separate variance and correlation parameters for those with a college degree and those without. The estimated correlation between the errors from the main and auxiliary equation is 0.61 for individuals without a college degree and 0.65 for those with a college degree. The z statistics may be used for Wald tests of the null hypothesis that there is no endogenous treatment. For both treatment groups, we reject this hypothesis and conclude that having a college degree is an endogenous

treatment. Because the estimates are positive, we conclude that unobserved factors that increase the chance of having a college degree also tend to increase wage.

We can use `estat teffects` to estimate the average effect of a college degree on wage. We use the `atet` option to estimate the ATET.

```
. estat teffects, atet
Predictive margins                              Number of obs   = 6,000
                                                Subpop. no. obs = 2,234
```

	Margin	Unconditional std. err.	z	P>|z|	[95% conf. interval]	
ATET						
college						
(Yes vs No)	5.238589	.2047014	25.59	0.000	4.837382	5.639797

We estimate that the average wage for those who graduated from college is $5.24 higher than it would have been had those same individuals not graduated from college. This is $2.39 less than the result from our model in [ERM] **Example 2b** that did not account for the endogeneity of college graduation. We said "same individuals" to emphasize that $5.24 is a treatment effect on those who chose to attend college and graduated. More formally, it is our estimate of what the average increase in wage is in the whole population for everyone who chose to attend college and graduated.

But we do not need to stop there. We can analyze how college education affects different subpopulations. Usually, when we zoom into the different subpopulations, we can see effects that otherwise would be averaged out over the population. Before I do that though, I am going to change the data slightly to get nicer output:

```
. label drop tengrp
. label define tengrp 0 "0-3 years tenure" 4 "3-7 years tenure"
>                    8 "Over 7 years tenure"
. label values tenuregrp tengrp
```

Now, I am ready to use `margins` to zoom into some interesting subgroups of my data.

```
. margins college, over(agegrp tenuregrp) subpop(if college==1 & peduc==1)
> vce(unconditional)
  (output omitted)
```

Here we have narrowed our focus on individuals with a college degree whose parents have a college education, using the `subpop()` option, for different subpopulations defined by our age and tenure groupings, using the `over()` option. Also, by using the default average structural function prediction `asf`, we condition on the unobservable factors that increase the probability of graduating from college.

If you run the `margins` command, you will see that it takes a few seconds and that it produces a lot of output. Let's graph the results,

```
. marginsplot, by(college) byopts(style(altleg))
```


Predictive margins of college with 95% CIs
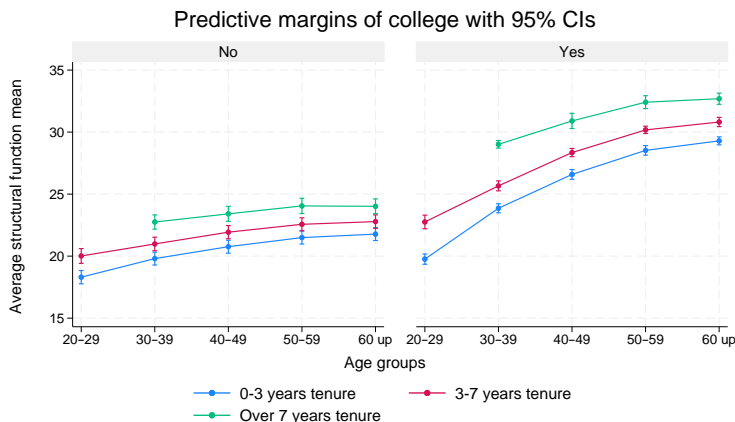
The age–earnings profiles on the left, where we have taken the degrees away from our college graduates, are distinctly different from those on the right, where they get to retain their degrees. We see that tenure does have an effect, and if we look closely, it has a larger effect on college graduates: the profiles are further apart on the right. What do the points on this graph represent? Each point in the panel on the right is the expected wage for someone who graduated from college, whose parents graduated from college, and who has the age and tenure shown on the graph. Each point on the left is a counterfactual where we assume those same people did not graduate from college.

Seeing that, we have to ask, what are the profiles of the effect of college? To find those, we just add an `r.` to `college` on our previous `margins` command.

```
. margins r.college, over(agegrp tenuregrp) subpop(if college==1 & peduc==1)
> vce(unconditional)
  (output omitted)
```

Again, the output is long, so we graph the results.

. marginsplot
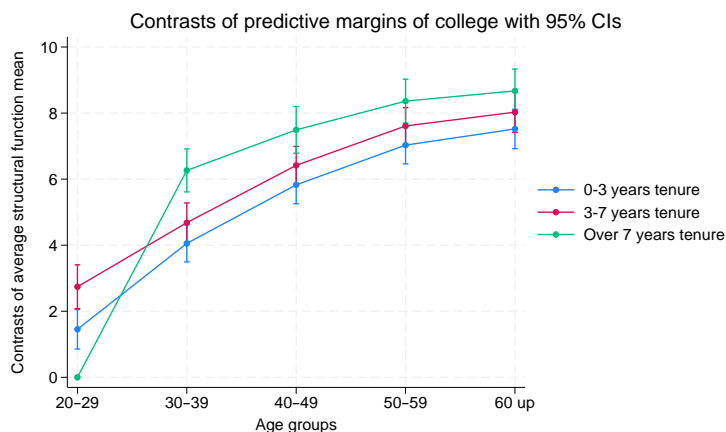
College affects wages the least when people are young and have no tenure. The largest effects are seen for those older than 50 and even more so when they also have long tenure. Each point represents the expected increase in wages due to graduating from college among those who chose to attend college and graduated. So each is an average treatment effect on the treated (ATET). Unlike overall average ATETs, these are conditioned on being at a specific age and having a specific tenure. Each point is bracketed by a pointwise 95% confidence interval. The confidence intervals reveal that we have pretty tight estimates for each of the ATETs. Note that the previous graph also displayed 95% confidence intervals. They were just so narrow that they are difficult to see.

As we saw above, margins allows us to answer numerous questions depending on the counterfactuals and subpopulations we defined.

See *Treatment* under *Methods and formulas* in [ERM] **eregress** and *Estimating treatment effects with margins* in [R] **margins, contrast** for additional information about calculating the ATET.

Video example

Extended regression models: Nonrandom treatment assignment

Also see

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

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

[ERM] **estat teffects** — Average treatment effects for extended regression models

[ERM] **Intro 9** — Conceptual introduction via worked example

Example 3a — Probit regression with continuous endogenous covariate

Description

In this example, we show how to estimate and interpret the results of an extended regression model with a binary outcome and continuous endogenous covariate.

Remarks and examples

In [ERM] **Example 1a** through [ERM] **Example 1c**, we showed how researchers at the fictional State University might approach an investigation of the relationship between the high school grade point average (GPA) of the students the university admits and their final college GPA. Suppose instead that they would like to know how the probability of college graduation is related to high school GPA. They again suspect that high school GPA is endogenous in a model of the probability of college graduation.

Their model for graduation includes parental income in \$10,000s and whether the student had a roommate who also went to State U. The State U researchers expect that the effect of high school competitiveness on the probability of graduating from college is negligible once the other covariates are controlled for. So they use the ranking of the high school (hscomp) as the instrumental variable for high school GPA. They also include parental income in the auxiliary model for high school GPA.

We want to make inferences about how our covariates affect graduation rates in the population, not just in our sample. We add vce(robust) so that subsequent calls to estat teffects and margins will be able to consider our sample as a draw from the population.

```
. use https://www.stata-press.com/data/r18/class10
(Class of 2010 profile)

. eprobit graduate income i.roommate, endogenous(hsgpa = income i.hscomp)
> vce(robust)

Iteration 0:  Log pseudolikelihood = -1418.5008
Iteration 1:  Log pseudolikelihood = -1418.4414
Iteration 2:  Log pseudolikelihood = -1418.4414
```

```
Extended probit regression                        Number of obs =   2,500
                                                  Wald chi2(3)  =  326.79
Log pseudolikelihood = -1418.4414                 Prob > chi2   =  0.0000
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
graduate						
income	.1597677	.0158826	10.06	0.000	.1286384	.1908969
roommate						
Yes	.2636312	.0563563	4.68	0.000	.1531748	.3740876
hsgpa	1.01877	.4324788	2.36	0.018	.1711273	1.866413
_cons	-3.647166	1.204728	-3.03	0.002	-6.008389	-1.285943
hsgpa						
income	.047859	.0016461	29.07	0.000	.0446327	.0510853
hscomp						
Moderate	-.135734	.0114717	-11.83	0.000	-.158218	-.1132499
High	-.225314	.0195055	-11.55	0.000	-.2635441	-.1870838
_cons	2.794711	.0127943	218.43	0.000	2.769634	2.819787
var(e.hsgpa)	.0685893	.0019597			.064854	.0725398
corr(e.hsgpa, e.graduate)	.3687006	.0919048	4.01	0.000	.1765785	.5337596

The estimate of the correlation between the errors of our two equations is 0.37 and is significantly different from zero, so we have endogeneity. Because the correlation is positive, we conclude that the unobservable factors that increase high school GPA also increase the probability of graduation.

The results for the main equation are interpreted as you would those from `probit`. We can obtain directions but not effect sizes from the coefficients in the main equation. For example, we see that family income and high school GPA are positively associated with the probability that a student graduates.

Let's ask something more interesting. What if we could increase each student's high school GPA by one point, moving a 2.0 to a 3.0, a 2.5 to a 3.5, and so on? We obviously cannot increase anyone's GPA by one point if he or she is already above a 3.0; so we restrict our population of interest to students with a GPA at or below 3.0. `margins` will give us the population-average expected graduation rate given each student's current GPA if we specify `at(hsgpa=generate(hsgpa))`. It will also give us the population-average expected graduation rate with an additional point in each student's GPA if we specify `at(hsgpa=generate(hsgpa+1))`. We want to hold each student's unobservable characteristics to be those that are implied by his or her current data by using the default average structural function prediction.

```
. margins, at(hsgpa=generate(hsgpa)) at(hsgpa=generate(hsgpa+1))
> subpop(if hsgpa <= 3) vce(unconditional)
```

```
Predictive margins                                    Number of obs   = 2,500
                                                      Subpop. no. obs = 1,430
```

Expression: Average structural function probability, predict()
1._at: hsgpa =   hsgpa
2._at: hsgpa = hsgpa+1

	Margin	Unconditional std. err.	z	P>|z|	[95% conf. interval]
_at					
1	.4315243	.0125571	34.37	0.000	.4069129 .4561357
2	.7737483	.1057771	7.31	0.000	.5664289 .9810677

For students with a high school GPA at or below 3.0, the expected graduation rate is 43%. If those same students are given an additional point in their GPA, the graduation rate rises to 77%.

By adding contrast(at(r)) to our margins command, we can difference those two counterfactuals and estimate the average effect of giving an additional point of GPA. We also added effects to add test statistics and nowald to clean up the output.

```
. margins, at(hsgpa=generate(hsgpa)) at(hsgpa=generate(hsgpa+1))
> subpop(if hsgpa <= 3) contrast(at(r) nowald effects) vce(unconditional)
```

```
Contrasts of predictive margins                      Number of obs   = 2,500
                                                     Subpop. no. obs = 1,430
```

Expression: Average structural function probability, predict()
1._at: hsgpa =   hsgpa
2._at: hsgpa = hsgpa+1

	Contrast	Unconditional std. err.	z	P>|z|	[95% conf. interval]
_at					
(2 vs 1)	.342224	.1070061	3.20	0.001	.1324959 .5519521

Giving students an additional point in their GPA increased graduation rates by just over 34 percentage points, with a 95% confidence interval from 13 to 55 percentage points.

Does this effect differ across any of our other covariates? Our dataset has a grouping variable for family income incomegrp, so let's estimate the effect within each income grouping. We just add over(incomegrp) to our prior margins command.

```
. margins, at(hsgpa=generate(hsgpa)) at(hsgpa=generate(hsgpa+1))
> subpop(if hsgpa <= 3) contrast(at(r) nowald effects) noatlegend
> vce(unconditional) over(incomegrp)
```

```
Contrasts of predictive margins                    Number of obs   = 2,500
                                                   Subpop. no. obs = 1,430
```

```
Expression: Average structural function probability, predict()
Over:       incomegrp
```

	Contrast	Unconditional std. err.	z	P>|z|	[95% conf. interval]	
_at@						
incomegrp						
(2 vs 1)						
< 20K	.3690987	.1367297	2.70	0.007	.1011134	.6370839
(2 vs 1)						
20–39K	.3698609	.1254644	2.95	0.003	.1239552	.6157667
(2 vs 1)						
40–59K	.3516159	.1026905	3.42	0.001	.1503463	.5528855
(2 vs 1)						
60–79K	.3094611	.0798654	3.87	0.000	.1529277	.4659944
(2 vs 1)						
80–99K	.255203	.0572386	4.46	0.000	.1430174	.3673887
(2 vs 1)						
100–119K	.1829494	.038826	4.71	0.000	.1068519	.2590469
(2 vs 1)						
120–139K	.1238027	.0344788	3.59	0.000	.0562255	.19138
(2 vs 1)						
140K up	.0485429	.0139112	3.49	0.000	.0212775	.0758083

The effect is largest for the low-income groups and declines as income goes up. It becomes almost negligible for students from households whose income is above $140,000.

We can see this relationship more clearly if we graph the results.

```
. marginsplot
```
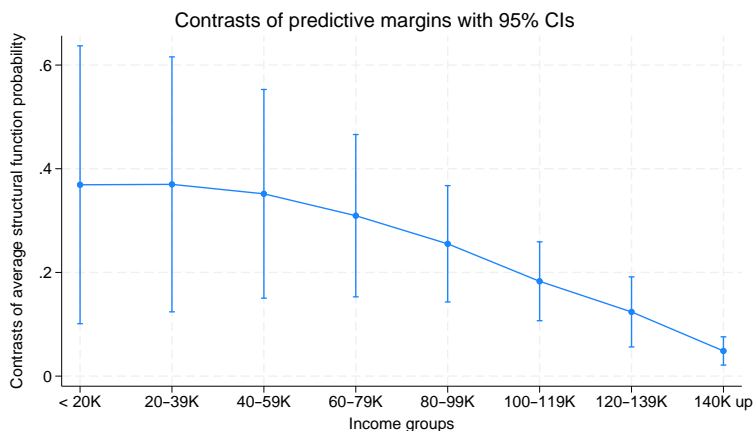

Figure 1.

Our point estimates of the effect on the probability of graduating are near 0.4 for the lowest-income groups and fall below 0.2 for incomes over $100,000.

So we can examine subpopulation averages and effects and make inferences about their values.

Let's see whether we can observe the effects of the unobservables that are affecting both graduation probability and high school GPA. We will restrict our attention to those with very low and very high incomes. We do not want any confounding from the individual's roommate status, so we will also include only those who have a roommate. Before we start, we modify our data to simplify our analysis.

```
. generate smpl = roommate==1 & (income < 3 | income > 10)
. generate byte hlincome = 1 if income < 3
(1,870 missing values generated)
. replace hlincome = 2 if income > 10
(216 real changes made)
. label define hiloinc 1 "Income < $30,000" 2 "Income > $100,000"
. label values hlincome hiloinc
```

We would like to know the expected graduation rates for those with high and low income over the groupings of high school GPA.

```
. margins, subpop(smpl) over(hsgpagrp hlincome) vce(unconditional)
(output omitted )
. marginsplot
(output omitted )
```


Figure 2.

Clearly, those with high incomes have much higher expected graduation rates.

What if we could level the playing field and give everyone the same family income level? We will give everyone $100,000. That is a substantial increase for those in the low-income group and somewhat of a reduction for most in the high-income group. We form these counterfactuals by adding `at(income=10)` to our `margins` command and then graph them.

```
. margins, subpop(smpl) over(hsgpagrp hlincome) at(income=10) vce(unconditional)
  (output omitted )
. marginsplot
  (output omitted )
```
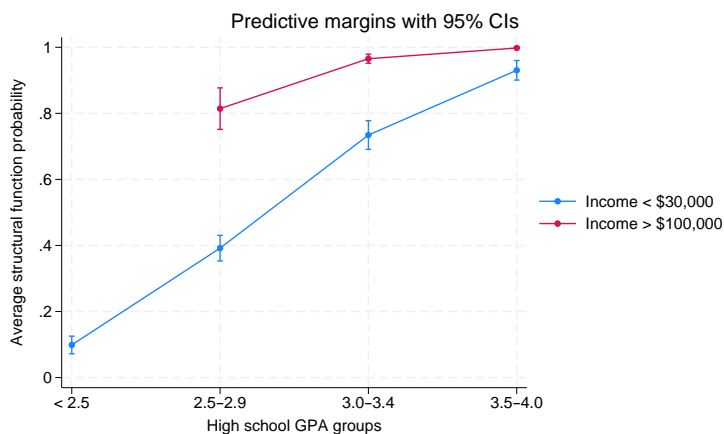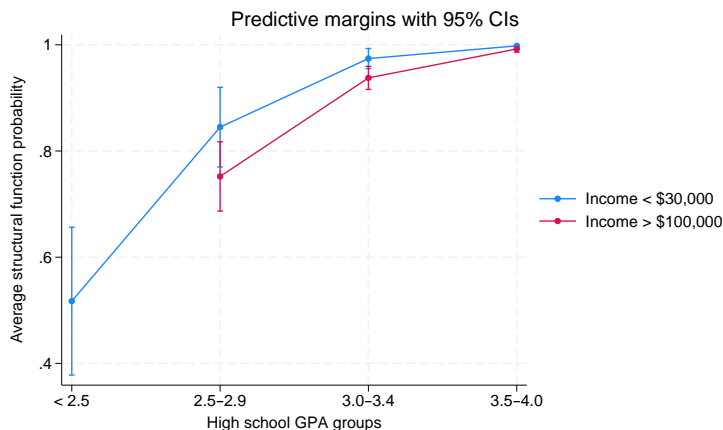

Figure 3.

Now, we have given everyone $100,000 of family income, but each person has retained his or her unobservable characteristics. The red line for those who originally had incomes over $100,000 is still pretty much where it was before. The line for those with incomes less than $30,000 is far higher than it was. It is now above the line for the high income students. For any level of GPA, the low-income group now has a higher graduation rate than the high-income group. We have locked up all the regressors in the main model by either putting them in groups or assigning them a counterfactual. So any differences that we see must be attributable to the individual's unobserved characteristics. If these were real data, the results would not be surprising. One way to view it is that any level of high school GPA is more difficult to obtain for individuals in a low-income family. That would mean that their unobservables will also tend to increase their graduation rate.

The results validate our previous conclusions. The important message is that we can analyze fully conditional counterfactuals and make complex inferences. These inferences account for not only observable characteristics but also unobservable traits and thus have a structural interpretation.

Also see

[ERM] **eprobit** — Extended probit regression

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

[ERM] **Intro 3** — Endogenous covariates features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Description   Remarks and examples   Also see

Description

We model a binary outcome that depends on a continuous endogenous covariate and has an endogenous treatment by using `eprobit` with the `endogenous()` and `entreat()` options.

Remarks and examples

Continuing from [ERM] **Example 3a**, State U administrators have implemented a voluntary program to increase retention freshman year. Whether a student chose to participate is stored in the indicator variable `program`. They are concerned that unobservable factors that influence a student's decision to participate in the college retention program also influence the probability of graduation. For example, students who have higher self-motivation may be more likely to join and also more likely to graduate without the program. Thus, they are concerned that participation in the program may be an endogenously chosen treatment. Further, they would like to control for the possibility that the unobserved factors affecting graduation have different relationships with the unobserved factors that affect participation and high school GPA for those who participated and those who did not.

The researchers believe the program was easier to access for students who lived on campus freshman year. They also think students who had scholarships may have been more motivated to attend the program. However, they do not believe either of these variables independently affects the probability of graduation after controlling for other covariates in the model. They use an indicator for on-campus residence during the freshman year (`campus`), having a scholarship of any kind (`scholar`), and parents' income in the treatment assignment model.

```
. eprobit graduate income i.roommate, endogenous(hsgpa = income i.hscomp)
> entreat(program = i.campus i.scholar income, pocorrelation) vce(robust)

Iteration 0:  Log pseudolikelihood = -2793.4696
Iteration 1:  Log pseudolikelihood = -2792.8365
Iteration 2:  Log pseudolikelihood = -2792.7434
Iteration 3:  Log pseudolikelihood = -2792.7433
```

Extended probit regression

Log pseudolikelihood = -2792.7433

Number of obs = 2,500
Wald chi2(8) = 335.99
Prob > chi2 = 0.0000

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]
graduate						
program# c.income						
0	.1824158	.0238431	7.65	0.000	.1356842	.2291475
1	.1865878	.0245008	7.62	0.000	.1385672	.2346084
roommate# program						
Yes#0	.3099365	.0827593	3.75	0.000	.1477313	.4721418
Yes#1	.2436647	.076438	3.19	0.001	.093849	.3934805
program# c.hsgpa						
0	1.083248	.6284794	1.72	0.085	-.1485491	2.315045
1	1.004868	.5841352	1.72	0.085	-.1400159	2.149752
program						
0	-4.201051	1.779367	-2.36	0.018	-7.688547	-.7135555
1	-3.590705	1.623489	-2.21	0.027	-6.772685	-.4087256
program						
campus						
Yes	.7437785	.0734259	10.13	0.000	.5998663	.8876906
scholar						
Yes	.8963839	.058676	15.28	0.000	.7813811	1.011387
income	-.0798981	.008895	-8.98	0.000	-.097332	-.0624643
_cons	-.3806292	.0859392	-4.43	0.000	-.5490669	-.2121916
hsgpa						
income	.0478622	.0016462	29.08	0.000	.0446358	.0510886
hscomp						
Moderate	-.1351312	.0115348	-11.72	0.000	-.1577391	-.1125233
High	-.226768	.0194135	-11.68	0.000	-.2648178	-.1887181
_cons	2.794476	.0128195	217.99	0.000	2.769351	2.819602
var(e.hsgpa)	.0685876	.0019597			.0648522	.0725381
corr(e.pro~m, e.graduate)						
program						
0	.3223659	.1492073	2.16	0.031	.0079293	.5787898
1	.4280942	.1358716	3.15	0.002	.1307496	.6547793

	Margin	std. err.	z	P>|z|	[95% conf. interval]	
corr(e.hsgpa, e.graduate) program						
0	.4241328	.1274031	3.33	0.001	.1471666	.6394236
1	.3792206	.1220983	3.11	0.002	.1190782	.5906426
corr(e.hsgpa, e.program)	-.0206714	.0264813	-0.78	0.435	-.0724717	.03124

The main equation output is slightly different from that in [ERM] **Example 3a**. Because program was specified as a treatment, it was automatically interacted with each of the other covariates in the graduate equation.

We specified the pocorrelation suboption in entreat() so that we estimate separate correlation parameters for the two potential outcomes—for those who participated and those who did not. In the treated group, the correlation of the errors from the graduation equation and those from the program participation equation corr(e.program,e.graduate) is estimated to be 0.43 and is significantly different from zero. The researchers conclude that unobservable factors that increase the chance of participating in the program also increase the chance of graduating among the individuals that participate in the program.

Now, we use estat teffects to estimate the ATE of program participation on college graduation. We specified vce(robust) when we fit the model, so estat teffects reports standard errors and tests for the population ATE.

```
. estat teffects
```
Predictive margins   Number of obs = 2,500

		Unconditional				
	Margin	std. err.	z	P>|z|	[95% conf. interval]	
ATE program (1 vs 0)	.1053155	.0491882	2.14	0.032	.0089083	.2017226

We estimate that the ATE is 0.11. In other words, the average probability of graduating increases by 0.11 when all students participate in the program versus when no students participate in the program.

We might be interested if those students who self-selected into the program increased their graduation probability by more than 0.11. We estimate the average treatment effect on the treated (ATET).

```
. estat teffects, atet
```
Predictive margins   Number of obs   = 2,500
                                                      Subpop. no. obs = 1,352

|		Unconditional				
	Margin	std. err.	z	P>|z|	[95% conf. interval]	
ATET program (1 vs 0)	.1255127	.0674688	1.86	0.063	-.0067236	.2577491

In this case, the program is only a little more effective on average for those who chose to participate than it would have been for everyone. The ATET is 0.13, only 0.02 higher than the ATE.

Those are the overall averages. Do graduation rates for participants and nonparticipants differ by high school GPA and parents' income? Our dataset has grouping variables, so we can let `margins` estimate graduation rates for subpopulations defined by all three covariates.

```
. margins, over(program incomegrp hsgpagrp) vce(unconditional)
  (output omitted)
```

The output is copious. You can type the command and see it if you like. The patterns are easier to see on a `marginsplot`.

```
. marginsplot, plot(program) xlabels(0 4 8 12)
```


Predictive margins with 95% CIs
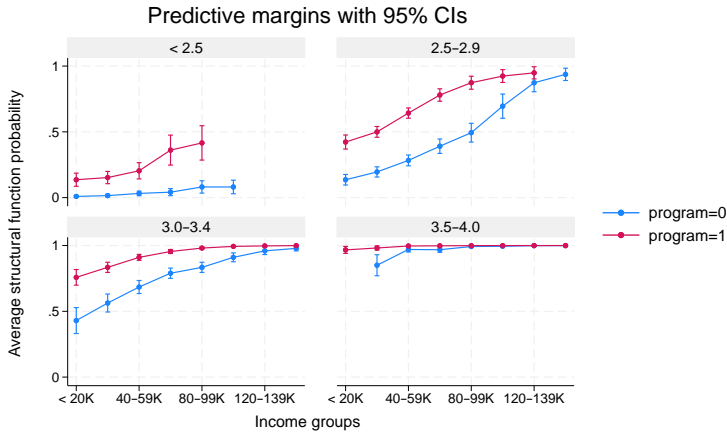
The red line shows expected graduation rates for those who participated in the program. The blue line shows rates for nonparticipants. Clearly, the differences between the groups in the program and those out of the program differ dramatically across GPA and family income. For GPAs at or above 3.5, the graduation rates are so high that there was no room for differences. For those with GPAs below 2.5, we see differences, with participation graduation rates being higher than nonparticipation, but lots of variation as income increases. For the other groups, the graduation rates are estimated to be substantially higher among those who participated.

We were careful not to call the comparisons above effects or attribute them directly to the program. They are indeed expected rates for the groups, but the students self-selected into program participation groups. If we want to compare graduation rates assuming all students do not participate and then assuming all students do participate, we type

```
. margins r.program, over(incomegrp hsgpagrp) vce(unconditional)
> contrast(nowald)
  (output omitted)
```

The output is again long, so we leave you to see it for yourself. The graphs reveal the patterns across groups.

```
. marginsplot, by(hsgpagrp) xlabels(0 4 8 12)
```


Contrasts of predictive margins of program with 95% CIs
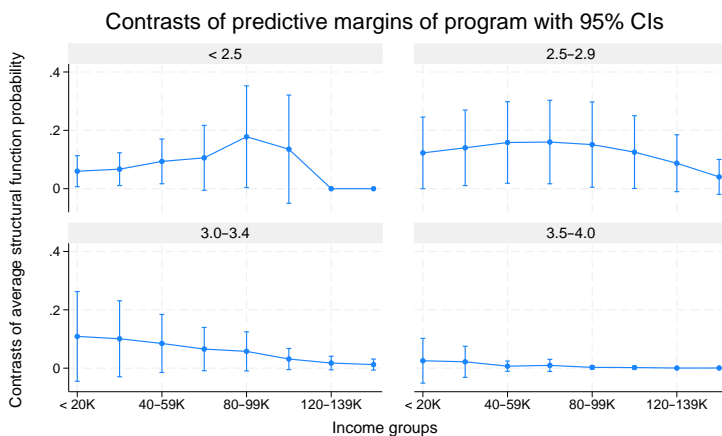
These differences are close to what we would have seen had we differenced the red and blue lines of the first graph. In this graph, each point is an estimate of the average treatment effect for a subpopulation defined by a range of GPAs and a range of family income. We note that the confidence intervals, as represented by the capped lines, are fairly wide.

Also see

[ERM] **eprobit** — Extended probit regression

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

[ERM] **estat teffects** — Average treatment effects for extended regression models

[ERM] **Intro 3** — Endogenous covariates features

[ERM] **Intro 5** — Treatment assignment features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Description   Remarks and examples   Also see

Description

In this example, we show how to estimate and interpret the results of an extended regression model with a binary outcome and endogenous sample selection.

Remarks and examples

We are interested in whether regular exercise and body mass index (BMI) influence the chance of having a subsequent heart attack. In our fictional study, we collected data on 625 men who had a heart attack when they were between the ages of 50 and 55. Some men withdrew from the study before it completed, and we believe their reasons for leaving are related to unobserved factors that also affect their chances of having a second heart attack. We did, however, observe all cases where a second heart attack was fatal.

To account for the endogenous sample selection, we specify an auxiliary model for selection using a covariate that belongs in the auxiliary model and is excluded from the main equation. We expect that the direct effect of whether a man had regular checkups before the study is negligible after we condition on other covariates.

The outcome of interest is whether the man had another heart attack within five years of his first heart attack (`attack`). We believe that the man's current age is also an important exogenous covariate along with BMI. We model the indicator for whether the man was observed for the full five years of the study (`full`) as a function of an indicator for having regular checkups along with the covariates from the main equation.

247

```
. use https://www.stata-press.com/data/r18/heartsm
(Heart attacks)

. eprobit attack age bmi i.exercise, select(full = age bmi i.checkup) vce(robust)

Iteration 0:   Log pseudolikelihood = -409.23137
Iteration 1:   Log pseudolikelihood = -408.78569
Iteration 2:   Log pseudolikelihood = -408.78452
Iteration 3:   Log pseudolikelihood = -408.78452
```

```
Extended probit regression                        Number of obs    =      625
                                                      Selected    =      458
                                                   Nonselected    =      167

                                                  Wald chi2(3)    =   142.85
Log pseudolikelihood = -408.78452                 Prob > chi2     =   0.0000
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
attack						
age	.2237091	.0351334	6.37	0.000	.1548489	.2925693
bmi	.1760896	.0298853	5.89	0.000	.1175155	.2346636
exercise						
Yes	-1.438937	.1515198	-9.50	0.000	-1.735911	-1.141964
_cons	-15.78445	2.105945	-7.50	0.000	-19.91202	-11.65687
full						
age	-.1599347	.032953	-4.85	0.000	-.2245214	-.095348
bmi	-.1146582	.0208896	-5.49	0.000	-.1556011	-.0737152
checkup						
Yes	2.306638	.1660248	13.89	0.000	1.981236	2.632041
_cons	11.66488	1.942686	6.00	0.000	7.857284	15.47247
corr(e.full, e.attack)	-.4537026	.1636665	-2.77	0.006	-.71301	-.0852183

We estimate that the correlation between the errors from the outcome equation and the errors from the selection equation is -0.45. This is significantly different from zero, so selection into the study is endogenous. Because the correlation is negative, we conclude that unobserved factors that increase the chance of staying in the study tend to occur with unobserved factors that decrease the chance of having a subsequent heart attack.

The results for the main outcome equation (`attack`) and auxiliary selection equation (`full`) are interpreted just as you would those from `heckprobit`. Which is to also say that the results for the main equation can be interpreted as you would those from a probit regression using `probit` on uncensored data. The goal of including a selection model is to estimate the parameters of the main equation as though there were no selection.

Age and BMI have increased the chances of having another heart attack, while regular exercise decreases the chances. However, the magnitude of the effect on the probability of another heart attack cannot be determined from the coefficient estimates themselves. We can use `margins` to examine the effect of different covariates on the probability of having a second heart attack. But first we want to investigate a possible further complication in our data: regular exercise may be an endogenous treatment. We explore this in [ERM] **Example 4b**.

Also see

Title

Example 4b — Probit regression with endogenous treatment and sample selection

Description

Continuing from [ERM] **Example 4a**, we show you how to estimate and interpret the results of a model for a binary outcome when the model includes an endogenous treatment and the data are subject to endogenous sample selection.

Remarks and examples

In [ERM] **Example 4a**, we ignored the possibility that regular exercise was an endogenous treatment. However, we suspect that unobserved factors that influence the choice to exercise may be correlated with the unobserved factors that affect the chance of having another heart attack.

We would like to know the average expected change in probability of having a subsequent heart attack for those who exercise. That is, we are interested in estimating the average treatment effect on the treated (ATET). We continue to include BMI and age in our outcome model, and to account for endogenous sample selection, we specify the same auxiliary model for selection we did in [ERM] **Example 4a**. We add a third equation to account for endogenous treatment assignment. Whether a man ever joined a gym is an instrumental variable predicting exercise that we do not expect to otherwise affect attack, so we include it in our model for regular exercise.

```
. eprobit attack age bmi, select(full = age bmi i.checkup)
> entreat(exercise = bmi i.gym) vce(robust)
  (iteration log omitted)
```

Extended probit regression					Number of obs	=	625
					Selected	=	458
					Nonselected	=	167
					Wald chi2(6)	=	111.78
Log pseudolikelihood = −711.90507					Prob > chi2	=	0.0000

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
attack						
exercise#						
c.age						
No	.2156634	.0550909	3.91	0.000	.1076872	.3236397
Yes	.221641	.0423742	5.23	0.000	.1385891	.3046928
exercise#						
c.bmi						
No	.1925833	.04278	4.50	0.000	.108736	.2764306
Yes	.2134441	.038381	5.56	0.000	.1382186	.2886696
exercise						
No	−16.07086	3.282712	−4.90	0.000	−22.50486	−9.636863
Yes	−17.84655	2.61864	−6.82	0.000	−22.97899	−12.71411
full						
age	−.1650386	.0321825	−5.13	0.000	−.228115	−.1019621
bmi	−.1143184	.0206726	−5.53	0.000	−.154836	−.0738008
checkup						
Yes	2.315167	.1639928	14.12	0.000	1.993747	2.636587
_cons	11.92957	1.898426	6.28	0.000	8.208727	15.65042
exercise						
bmi	−.1815549	.0211349	−8.59	0.000	−.2229786	−.1401313
gym						
Yes	1.517225	.1248316	12.15	0.000	1.27256	1.761891
_cons	3.941703	.5728064	6.88	0.000	2.819023	5.064383
corr(e.full, e.attack)	−.5338178	.1584217	−3.37	0.001	−.7737932	−.1598432
corr(e.exe~e, e.attack)	−.435728	.1467897	−2.97	0.003	−.676196	−.1113554
corr(e.exe~e, e.full)	.3212358	.0928654	3.46	0.001	.1293396	.4899396

The correlation between the errors that affect having a subsequent heart attack and the errors that affect staying in the study is estimated to be −0.53 and is significant. So we do have endogenous selection and conclude that unobservable factors that increase the chance of staying in the study also tend to decrease the chance of having a subsequent heart attack.

Increases in age and BMI increase the chance of having another heart attack. This is true both for those who exercise, coefficients marked yes, and for those who do not, coefficients marked no.

We use estat teffects to estimate the ATET of regular exercise on having a subsequent heart attack. We specified vce(robust) when we fit the model so that estat teffects will report unconditional standard errors for the population ATET rather than the sample ATET.

```
. estat teffects, atet
```

Predictive margins   Number of obs   = 625
                                                                Subpop. no. obs = 291

	Margin	Unconditional std. err.	z	P>|z|	[95% conf. interval]	
ATET						
exercise						
(Yes vs No)	-.2993399	.0773753	-3.87	0.000	-.4509927	-.1476871

The estimated ATET is −0.30. Thus, for those who exercise regularly, the average probability of having a subsequent heart attack is 0.30 lower than it would be if they did not exercise regularly.

Also see

[ERM] **eprobit** — Extended probit regression

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

[ERM] **estat teffects** — Average treatment effects for extended regression models

[ERM] **Intro 4** — Endogenous sample-selection features

[ERM] **Intro 5** — Treatment assignment features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Description   Remarks and examples   Also see

Description

We model a binary outcome that depends on an endogenous ordinal treatment by using `eprobit` with the `entreat()` option.

Remarks and examples

We are interested in estimating the average treatment effects (ATEs) of different levels of exercise intensity on the chance of having a subsequent heart attack. In our fictional study, we collected data on 625 men who had a heart attack when they were between the ages of 50 and 55. The outcome of interest is whether the man had another heart attack within five years of his first heart attack (`attack`). We believe that body mass index (BMI) and age are important covariates.

The `exintensity` variable records the intensity of exercise using the scale of 0 (no exercise), 1 (moderate), and 2 (heavy). We suspect that unobserved factors that influence the choice to exercise at a certain intensity level also affect the chance of having another heart attack, so we specify `exintensity` as an endogenous treatment. Whether an individual ever joined a gym is included as an instrumental covariate in the treatment model that we specify in `entreat()`.

```
. use https://www.stata-press.com/data/r18/heartsm
(Heart attacks)
. eprobit attack age bmi, entreat(exintensity = bmi i.gym) vce(robust)
  (iteration log omitted)
```

Extended probit regression   Number of obs =   625

                                                  Wald chi2(9)   = 152.33

Log pseudolikelihood = -728.6686   Prob > chi2   = 0.0000

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
attack						
exintensity#						
c.age						
None	.2118759	.0514612	4.12	0.000	.1110138	.312738
Moderate	.2338466	.0425341	5.50	0.000	.1504813	.3172119
Heavy	.2346887	.0805152	2.91	0.004	.0768818	.3924957
exintensity#						
c.bmi						
None	.1948171	.0386314	5.04	0.000	.119101	.2705332
Moderate	.2062276	.0405785	5.08	0.000	.1266952	.2857599
Heavy	.2155222	.0765592	2.82	0.005	.0654689	.3655755
exintensity						
None	-15.90911	3.043587	-5.23	0.000	-21.87444	-9.943793
Moderate	-18.2922	2.499325	-7.32	0.000	-23.19079	-13.39362
Heavy	-18.61821	5.395246	-3.45	0.001	-29.1927	-8.043721
exintensity						
bmi	-.1720462	.0204172	-8.43	0.000	-.2120632	-.1320292
gym						
Yes	1.518834	.1192361	12.74	0.000	1.285136	1.752532
/exintensity						
cut1	-3.677846	.5537938			-4.763262	-2.59243
cut2	-2.386538	.5372719			-3.439572	-1.333505
corr(e.exi~y,						
e.attack)	-.4722803	.1091789	-4.33	0.000	-.6575129	-.2332112

The estimated correlation between the errors in the main outcome and auxiliary treatment equations is −0.47. This is significantly different from zero, so we confirm that the choice of exercise intensity level is endogenous. Because it is negative, we conclude that unobservable factors that increase the intensity of exercising tend to decrease the chance of having a subsequent heart attack. The cutpoints for the ordered probit model for the endogenous treatment are shown just beneath the treatment model.

The coefficients for exintensity in the main equation indicate that both moderate and heavy exercise have a negative effect because they are smaller, more negative, than the coefficient for no exercise. BMI has a positive effect on the chance of having another heart attack, regardless of exercise level. In fact, the values of the three coefficients for bmi are so close that we might not need separate parameters for the three levels of exercise. The same could be said of the three coefficients on age.

Example 5 — Probit regression with endogenous ordinal treatment   255

The coefficients for the intercepts of heavy and moderate exercise are close in magnitude. To test whether these two coefficients are equal, we can use `test`.

```
. test 1.exintensity == 2.exintensity

 ( 1)  [attack]1.exintensity - [attack]2.exintensity = 0

          chi2(  1) =     0.00
        Prob > chi2 =     0.9557
```

We cannot reject that the coefficients are equal.

We also have separate coefficients on `age` and `bmi` for heavy and moderate exercise. To jointly test the equality of each coefficient associated with heavy exercise with the corresponding coefficient associated with moderate exercise, we type

```
. test (1.exintensity == 2.exintensity)
>      (1.exintensity#c.bmi == 2.exintensity#c.bmi)
>      (1.exintensity#c.age == 2.exintensity#c.age)

 ( 1)  [attack]1.exintensity - [attack]2.exintensity = 0
 ( 2)  [attack]1.exintensity#c.bmi - [attack]2.exintensity#c.bmi = 0
 ( 3)  [attack]1.exintensity#c.age - [attack]2.exintensity#c.age = 0

          chi2(  3) =     0.04
        Prob > chi2 =     0.9983
```

We do not have any evidence that heavy and moderate exercise have a different effect on the probability of a second heart attack.

That was some pretty tricky coefficient referencing in our `test` command. We suggest you type

```
. eprobit, coeflegend
```

to see how to reference coefficients in `test`, `nlcom`, and other postestimation commands.

What if every man in the population did not exercise? What if they all exercised moderately? What if they all exercised heavily? `estat teffects` can estimate the average probability of a second heart attack over the five years for each of those counterfactuals.

```
. estat teffects, pomean
```

Predictive margins   Number of obs = 625

		Unconditional				
	Margin	std. err.	z	P>|z|	[95% conf. interval]	
POmean						
exintensity						
None	.7918941	.0329342	24.04	0.000	.7273443	.856444
Moderate	.5419335	.0326336	16.61	0.000	.4779728	.6058942
Heavy	.5336232	.0767752	6.95	0.000	.3831466	.6840998

When no one in the population exercises, we estimate that 79% will have subsequent heart attacks. We are pretty confident in that number: the 95% confidence interval begins at 73% and ends at 86%. It does not matter much whether every man exercises moderately or heavily. Either intensity drops the expected rate of subsequent heart attacks to about 54%. These are the average potential-outcome means (POMs) under the three exercise-intensity regimes.

The difference between these POMs gives us estimates of the average treatment effects (ATEs) in the population. `estat teffects` will estimate those too.

```
. estat teffects
Predictive margins                                    Number of obs = 625
```

	Margin	Unconditional std. err.	z	P>|z|	[95% conf. interval]
ATE exintensity (Moderate vs None)	-.2499606	.0507776	-4.92	0.000	-.349483 -.1504383
(Heavy vs None)	-.2582709	.0965797	-2.67	0.007	-.4475637 -.0689781

We estimate that the ATE for heavy intensity compared with no exercise is -0.26. So the average probability of a subsequent heart attack is 26 percentage points lower when all men in the population exercise with heavy intensity versus when none of them exercise at all. The estimated ATE for moderate intensity versus none is -0.25. We again see no substantive difference between moderate and heavy exercise.

We used `vce(robust)` at estimation so that `estat teffects` would report standard errors that account for sampling variability in our covariates and are therefore valid for inference about the POMs, ATEs, and ATETs in the population from which our sample was drawn.

We have established that men who choose to exercise have unobserved attributes that tend to decrease their chance of another heart attack beyond the direct effect of exercising and beyond the effect of the other covariates. We can include the effect of these attributes for men who exercise by estimating the average treatment effect on the treated (ATET).

```
. estat teffects, atet
(subpopulation of first non-control treatment level assumed)
Predictive margins                                    Number of obs   = 625
                                                      Subpop. no. obs = 201
```

	Margin	Unconditional std. err.	z	P>|z|	[95% conf. interval]
ATET exintensity (Moderate vs None)	-.2992132	.0531572	-5.63	0.000	-.4033994 -.195027
(Heavy vs None)	-.309572	.100091	-3.09	0.002	-.5057467 -.1133973

The ATETs are both about 0.30, making them about 5 percentage points higher than the ATEs. We cannot, however, directly attribute that difference to the unobserved attributes. The ATETs are also averaged over subsamples and are therefore affected by any differences in the distribution of `age` or `bmi` in treated subsamples. The effect of those distributions could be either positive or negative.

With some care, we can extract just the effect of the unobserved attributes. It is a little tricky, both conceptually and syntactically. So continue reading only if you are truly interested.

Example 5 — Probit regression with endogenous ordinal treatment   257

Let's consider only the moderate exercisers. When we type

```
. margins r(0 1).exintensity, subpop(if exintensity == 1) vce(unconditional)
```

`margins` will produce the average difference for `exintensity` levels 0 and 1 (none and moderate). `subpop(if exintensity == 1)` restricts the average to men who exercised moderately.

`margins` would use the unobserved attributes associated with moderate exercise for both of the counterfactuals it requires to compute the contrast. Which is to say, it would use the true value of exercise intensity in the subpopulation we are averaging over. If you were to guess that this difference will be the ATET, you would be correct. For each man who chose moderate exercise, the ATET computation compares the man's expected probability of another attack using all the information on the man with that same man's expected probability if he instead did not choose to exercise. When we say "same man", we mean that he retains his original unobserved attributes when evaluating the counterfactual that he does not exercise. The ATET is then the average of that comparison over all those who exercise moderately.

We may also want to use [margins](#) to test whether the ATE for heavy exercise and the ATE for moderate exercise are equal. We specify two `predict()` options. On the first, we request treatment effects (`te`) for heavy exercisers (`tlevel(heavy)`). On the second, we request the treatment effects for moderate exercisers (`tlevel(moderate)`). We add `contrast(predict(r))` to request the difference between the predictions (their contrast). Finally, we use `vce(unconditional)` to request standard errors that account for sampling variability in the covariates and thus allow us to make inferences about the population.

```
. margins, predict(te tlevel(Heavy)) predict(te tlevel(Moderate))
> contrast(predict(r)) vce(unconditional)
```

Contrasts of predictive margins   Number of obs = 625

1._predict: treatment effect Pr(attack==Yes), exintensity: Heavy vs. None,
            predict(te tlevel(Heavy))
2._predict: treatment effect Pr(attack==Yes), exintensity: Moderate vs. None,
            predict(te tlevel(Moderate))

	df	chi2	P>chi2
_predict	1	0.01	0.9085

	Contrast	Unconditional std. err.	[95% conf. interval]	
_predict (2 vs 1)	.0083103	.0722814	−.1333587	.1499793

We cannot reject that the ATE for heavy exercise is equal to the ATE for moderate exercise. This result agrees with what we saw when we tested the coefficients for heavy and moderate exercise.

As we have seen repeatedly in the examples in the manual, most of the interesting questions are answered by `estat teffects` and `margins` and not by the parameter estimates themselves. This is particularly true of models estimated using [eprobit](#) and [eoprobit](#).

Also see

Title

Description   Remarks and examples   Also see

Description

In this example, we show how to estimate and interpret the results of an extended regression model with an ordinal outcome and endogenous treatment.

Remarks and examples

We are studying the effect of having health insurance on women's health status, which we measure with a health score from 1 (poor) to 5 (excellent). We want to estimate the average treatment effect (ATE) of insurance on the probability of having each of the five statuses. We suspect that our model needs to account for the health insurance being an endogenous treatment.

In our fictional study, we collect data on a sample of 6,000 women between the ages of 25 and 30. In addition to the insurance indicator, we include an indicator for whether the woman exercises regularly and the number of years of schooling she completed (grade) as exogenous covariates. For our treatment model, we use grade and an indicator for whether the woman is currently working or attending school (workschool), which is excluded from the outcome model.

```
. use https://www.stata-press.com/data/r18/womenhlth
(Women's health status)
. eoprobit health i.exercise grade, entreat(insured = grade i.workschool)
> vce(robust)
  (iteration log omitted)
```

Extended ordered probit regression   Number of obs =   6,000
                                                       Wald chi2(4) = 516.93
Log pseudolikelihood = -9105.4376   Prob > chi2   = 0.0000

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
health						
exercise# insured						
Yes#No	.5296149	.0619049	8.56	0.000	.4082835	.6509463
Yes#Yes	.5190249	.033872	15.32	0.000	.4526371	.5854127
insured# c.grade						
No	.1079014	.0250326	4.31	0.000	.0588383	.1569645
Yes	.1296456	.0107428	12.07	0.000	.10859	.1507012
insured						
grade	.3060024	.0100506	30.45	0.000	.2863036	.3257012
workschool						
Yes	.5387767	.0446794	12.06	0.000	.4512067	.6263466
_cons	-3.592452	.1348431	-26.64	0.000	-3.85674	-3.328165
/health						
insured# c.cut1						
No	.6282326	.2393499			.1591154	1.09735
Yes	-.7255086	.2470598			-1.209737	-.2412803
insured# c.cut2						
No	1.594089	.2300159			1.143266	2.044912
Yes	.4404531	.1986825			.0510426	.8298636
insured# c.cut3						
No	2.526424	.2241048			2.087186	2.965661
Yes	1.332514	.1845713			.9707608	1.694267
insured# c.cut4						
No	3.41748	.2356708			2.955574	3.879386
Yes	2.292828	.1760594			1.947758	2.637899
corr(e.ins~d, e.health)	.3414241	.0940374	3.63	0.000	.1460223	.5111858

The estimated correlation between the errors from the health status equation and the errors from the health insurance equation is 0.34. This is significantly different from zero, so the treatment choice of being insured is endogenous. Because it is positive, we conclude that unobserved factors that increase the chance of having health insurance tend to also increase the chance of being in a high health status.

We see estimates of both the coefficients and the cutpoints for two equations, one for insured women (`yes`) and one for uninsured (`no`). For both insured and uninsured, exercise and education have positive effects on health status.

We could use `estat teffects` to estimate the ATE of insurance on the probabilities of each health category.

```
. estat teffects
```

Feel free to run that command and see the results. We estimate and interpret other estimates of these ATEs in [ERM] **Example 6b** after adjusting for endogenous sample selection that is introduced in that example. The ATE estimates there are slightly different, but they estimate the same thing. Given a sufficiently large sample, the two sets of estimates would converge to the same values.

Also see

[ERM] **eoprobit** — Extended ordered probit regression

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

[ERM] **estat teffects** — Average treatment effects for extended regression models

[ERM] **Intro 5** — Treatment assignment features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

> **Example 6b —** Ordered probit regression with endogenous treatment and sample selection

Description

Continuing from [ERM] **Example 6a**, we show you how to estimate and interpret the results of a model for an ordinal outcome when the model includes an endogenous treatment and the data are subject to endogenous sample selection.

Remarks and examples

Suppose that we collected our data at doctors' offices and thus observe health score information only from women who visited their doctor in the study time frame ($\text{drvisit} = 1$). We suspect that unobserved factors that affect whether a woman visited the doctor are related to those that affect whether she has insurance and to those that affect her health status. Thus, we have an endogenously selected sample and an endogenously chosen treatment.

For our selection model, we use the endogenous treatment indicator for insurance status and regular checkups before the study (regcheck), which is excluded from the outcome model. Our command is otherwise exactly the same as specified in [ERM] **Example 6a**.

```
. eoprobit health i.exercise c.grade, entreat(insured = grade i.workschool)
> select(select = i.insured i.regcheck) vce(robust)
 (iteration log omitted)
Extended ordered probit regression              Number of obs  =   6,000
                                                     Selected  =   4,693
                                                  Nonselected  =   1,307

                                                Wald chi2(4)   =  367.30
Log pseudolikelihood = -9806.1189               Prob > chi2    =  0.0000
```

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
health						
exercise#						
insured						
Yes#No	.4169984	.0851131	4.90	0.000	.2501798	.583817
Yes#Yes	.5399986	.037546	14.38	0.000	.4664098	.6135874
insured#						
c.grade						
No	.1317866	.0342405	3.85	0.000	.0646765	.1988967
Yes	.1343324	.0129342	10.39	0.000	.1089818	.159683
select						
insured						
Yes	1.01669	.092325	11.01	0.000	.8357364	1.197644
regcheck						
Yes	.5374105	.0397297	13.53	0.000	.4595417	.6152793
_cons	-.1690644	.0743716	-2.27	0.023	-.3148301	-.0232987
insured						
grade	.3057852	.0100116	30.54	0.000	.2861628	.3254076
workschool						
Yes	.5314797	.0452607	11.74	0.000	.4427703	.6201891
_cons	-3.584315	.1348183	-26.59	0.000	-3.848554	-3.320077
/health						
insured#						
c.cut1						
No	.7262958	.3313472			.0768673	1.375724
Yes	-.5450451	.3181876			-1.168681	.0785912
insured#						
c.cut2						
No	1.719809	.3129056			1.106526	2.333093
Yes	.5683456	.2464686			.085276	1.051415
insured#						
c.cut3						
No	2.620793	.3056038			2.021821	3.219766
Yes	1.442022	.2227768			1.005387	1.878656
insured#						
c.cut4						
No	3.48945	.3158536			2.870389	4.108512
Yes	2.391497	.2090187			1.981828	2.801166
corr(e.sel~t,						
e.health)	.496699	.0990366	5.02	0.000	.2795869	.665485
corr(e.ins~d,						
e.health)	.4032487	.121518	3.32	0.001	.1421331	.6118937
corr(e.ins~d,						
e.select)	.2661948	.0555596	4.79	0.000	.1543216	.3713287

At both levels of the treatment, exercise and education still have positive effects on health status.

The correlation between the errors from the selection equation and the errors from the main equation is 0.497. This is significantly different from zero, so we confirm our suspicion of endogeneity. Because it is positive, we conclude that unobservable factors that increase the chance of being in the study also tend to increase the chance of being in a higher health status category.

What are the expected average probabilities of being in each health status if every woman had insurance? If every woman did not have insurance? We can answer those questions using estat teffects.

```
. estat teffects, pomean
Predictive margins                                      Number of obs = 6,000
POmean_Pr1: Pr(health=1=Poor)
POmean_Pr2: Pr(health=2=Not good)
POmean_Pr3: Pr(health=3=Fair)
POmean_Pr4: Pr(health=4=Good)
POmean_Pr5: Pr(health=5=Excellent)
```

	Margin	Unconditional std. err.	z	P>|z|	[95% conf. interval]	
POmean_Pr1						
insured						
No	.1028382	.0327177	3.14	0.002	.0387126	.1669637
Yes	.0058955	.0033611	1.75	0.079	-.0006921	.0124831
POmean_Pr2						
insured						
No	.2621517	.0479497	5.47	0.000	.1681719	.3561314
Yes	.0618234	.0116191	5.32	0.000	.0390504	.0845965
POmean_Pr3						
insured						
No	.3216819	.0259933	12.38	0.000	.270736	.3726278
Yes	.1759926	.0100741	17.47	0.000	.1562478	.1957374
POmean_Pr4						
insured						
No	.2144017	.0402798	5.32	0.000	.1354547	.2933488
Yes	.3237595	.009282	34.88	0.000	.3055672	.3419519
POmean_Pr5						
insured						
No	.0989265	.0521147	1.90	0.058	-.0032163	.2010694
Yes	.4325289	.0165829	26.08	0.000	.400027	.4650309

These are the estimates of the average potential-outcome means for the population. We can consider the values in this table to be either the expected proportions of all women being in a status category or the average probabilities of being in a status category. If we multiply by 100, we can talk about the expected percentage of all women being in a status category. The first pair of rows shows the probabilities of being in the first health status, poor. If all women are uninsured, the probability of having a poor health status is 0.10. If all women are insured, that probability falls to 0.01. At the other end of the spectrum, only 9.9% of women are expected to have excellent health if no women are insured. That number rises to 43.3% if all women are insured.

If we sum all the proportions labeled no, that sum is 1.0. The same is true of the proportions labeled yes. The sum of the proportions must be 1.0 because each woman can be in only one health status.

In any health status, if we subtract the potential-outcome mean when assuming all women are uninsured from the mean when assuming all women to be insured, we estimate the average treatment effect (ATE). This is the ATE that being insured has on the probability of being in the health status category. Let's do that.

```
. estat teffects
```

Predictive margins   Number of obs = 6,000

ATE_Pr1: Pr(health=1=Poor)
ATE_Pr2: Pr(health=2=Not good)
ATE_Pr3: Pr(health=3=Fair)
ATE_Pr4: Pr(health=4=Good)
ATE_Pr5: Pr(health=5=Excellent)

	Margin	Unconditional std. err.	z	P>|z|	[95% conf. interval]
ATE_Pr1 insured (Yes vs No)	-.0969427	.0333853	-2.90	0.004	-.1623767	-.0315086
ATE_Pr2 insured (Yes vs No)	-.2003283	.0552089	-3.63	0.000	-.3085358	-.0921207
ATE_Pr3 insured (Yes vs No)	-.1456893	.0322109	-4.52	0.000	-.2088216	-.082557
ATE_Pr4 insured (Yes vs No)	.1093578	.0437353	2.50	0.012	.0236382	.1950774
ATE_Pr5 insured (Yes vs No)	.3336024	.0637745	5.23	0.000	.2086066	.4585982

Looking at the last line, we see that the average probability of being in excellent health in the population of women aged 25 to 30 is 0.33 greater when all women have health insurance versus when no women have health insurance.

Because we specified vce(robust) at estimation, all of our estimates from estat teffects reported standard errors for the population ATE rather than standard errors that are conditional on the sample ATE.

Also see

Title

[Description](#)   [Remarks and examples](#)   [Reference](#)   [Also see](#)

Description

In this example, we show how to estimate and interpret the results of an extended regression model with a continuous outcome, a continuous endogenous covariate, and random effects.

Remarks and examples

We will use `nlswork.dta`, a subsample of the NLSY data (Center for Human Resource Research 1989) on young women aged 14–24 in 1968. These data are panel data; each individual was surveyed in multiple years ranging from 1968 to 1988.

Suppose that we want to study the relationship between the natural logarithm of wage (`ln_wage`) and the number of years at a job (`tenure`). We also model `ln_wage` with a quadratic effect of the individual's age (`age` and `c.age#c.age`), living in a metropolitan area (`not_smsa`), and whether the individual is African American (`2.race`). We suspect that the unobserved factors that influence the individual's job tenure are correlated with the unobserved factors that influence their wage, so we treat job tenure as an endogenous covariate. We use an individual's union status (`union`) and whether she lived in the southern United States (`south`) as instrumental covariates for tenure. Of course, these are not the instruments we would choose in real research, but they are useful for demonstrating how to use the commands below.

We also want to account for the within-panel correlation in our data, so we fit a random-effects model using `xteregress`. Before we can fit our model, we must use `xtset` to specify the panel identifier variable, in this case, `idcode`. Our data have already been `xtset`, so we type `xtset` to display the settings.

```
. use https://www.stata-press.com/data/r18/nlswork
(National Longitudinal Survey of Young Women, 14-24 years old in 1968)

. xtset
Panel variable: idcode (unbalanced)
 Time variable: year, 68 to 88, but with gaps
         Delta: 1 unit
```

We are now ready to fit our model. We want to make inferences about how our covariates affect the log wage in the population, not just in our sample. Therefore, we add the `vce(robust)` option so that subsequent calls to [margins](#) will consider our sample as a draw from the population.

By default, `xteregress` includes random effects for both `ln_wage` and `tenure` and allows these random effects to be correlated. Because of the complexity of this model, the command may take a few minutes to run.

```
. xteregress ln_wage age c.age#c.age i.not_smsa 2.race,
> endogenous(tenure = age c.age#c.age i.union 2.race i.south) vce(robust)
```
(*iteration log omitted*)

```
Extended linear regression                          Number of obs    =   19,007
Group variable: idcode                              Number of groups =    4,134

                                                    Obs per group:
                                                                 min =        1
                                                                 avg =      4.6
                                                                 max =       12
Integration method: mvaghermite                     Integration pts. =        7

                                                    Wald chi2(5)     =   384.25
Log pseudolikelihood = -53601.41                    Prob > chi2      =   0.0000
```

(Std. err. adjusted for 4,134 clusters in idcode)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
ln_wage						
age	.0161086	.0134428	1.20	0.231	-.0102388	.042456
c.age#c.age	-.0011178	.0002402	-4.65	0.000	-.0015887	-.000647
1.not_smsa	-.172498	.0122743	-14.05	0.000	-.1965552	-.1484408
race						
Black	-.2374388	.0254533	-9.33	0.000	-.2873263	-.1875513
tenure	.2300781	.0277646	8.29	0.000	.1756605	.2844957
_cons	1.690136	.2077606	8.14	0.000	1.282933	2.097339
tenure						
age	.0892847	.0599348	1.49	0.136	-.0281852	.2067547
c.age#c.age	.0033688	.0009943	3.39	0.001	.0014199	.0053176
1.union	.5584566	.0740956	7.54	0.000	.4132318	.7036814
race						
Black	.4691202	.1101411	4.26	0.000	.2532476	.6849929
1.south	-.4024058	.0628545	-6.40	0.000	-.5255983	-.2792132
_cons	-2.929734	.8800349	-3.33	0.001	-4.65457	-1.204897
var(e.ln_w~e)	.3654205	.0786259			.2396866	.5571114
var(e.tenure)	6.656475	.1285168			6.409292	6.913189
corr(e.ten~e, e.ln_wage)	-.9055589	.0213219	-42.47	0.000	-.9395846	-.8538145
var(ln_~e[idc~e])	.3314414	.0736048			.2144748	.5121973
var(ten~e[idc~e])	7.593483	.3027546			7.022688	8.210672
corr(ten~e[idc~e], ln_~e[idc~e])	-.8299334	.0421356	-19.70	0.000	-.8963409	-.7271053

The first two sections of the output provide the estimated coefficients in the equations for `ln_wage` and `tenure`. Because this is a linear regression, we can interpret the coefficients in the usual way. For example, we expect an increase of 0.23 in log wage for an additional year of job tenure.

Next, we see the estimates of the observation-level error variances and their correlation with the dependent variable. This is followed by estimates of the variances of the random effects and an estimate of their correlation with the dependent variable. If at least one of these correlations is significantly different from zero, we can conclude that `tenure` is endogenous. In our case, the correlation between the observation-level errors is −0.91, and the correlation between the random effects is −0.83. Because both are negative and significantly different from zero, we conclude that `tenure` is endogenous and that unobserved individual-level factors that increase job tenure tend to decrease log wage. Additionally, unobserved observation-level (time-varying) factors that increase job tenure tend to also decrease log wage.

We may also want to ask questions about specific groups in the population. Below, we consider how log wages differ by age group. We will study people between 18 and 40. As we mentioned in [ERM] **Intro 7**, the effects that `margins` computes by default have a causal interpretation if the model is correctly specified. The reason they do is that `margins` conditions on the level of endogeneity. We type

```
. margins, over(age) subpop(if (age>=18)*(age<=40)) vce(unconditional)
  (output omitted )
```

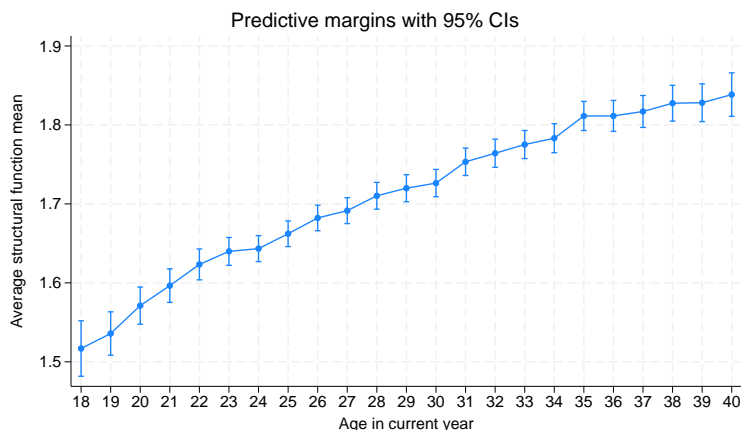and then graph the effects

```
. marginsplot
```


Figure 1.

What if we had not accounted for the level of endogeneity? This will not matter in cases where we average over the entire population and the effect of the unobservable becomes zero. It matters, however, when we look at effects over subpopulations. Below, we use `margins` to compute effects using the linear prediction, $x_{it}\beta$, by adding the option `predict(xb)`. The linear prediction is not conditioning on endogeneity.

```
. margins, over(age) predict(xb) subpop(if (age>=18)*(age<=40)) vce(unconditional)
  (output omitted )
. marginsplot
  (output omitted )
```
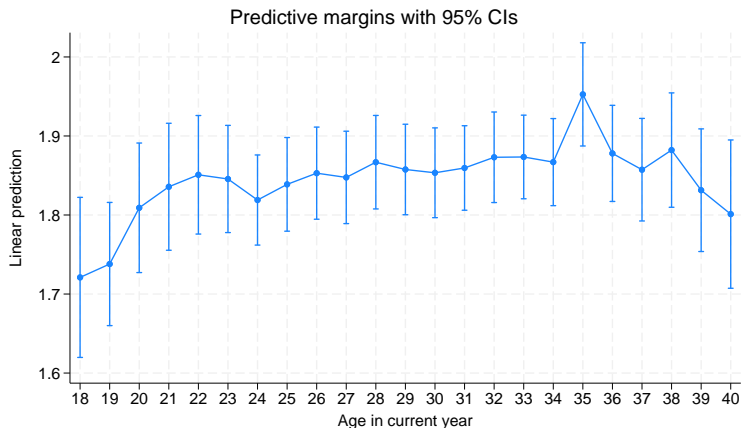

Figure 2.

Figure 1 shows wages increasing for each age group, whereas figure 2 shows wages decreasing after 35 years. Conditioning on the level of endogeneity to obtain structural effects matters even when we have linear models.

Reference

Center for Human Resource Research. 1989. *National Longitudinal Survey of Labor Market Experience, Young Women 14–24 years of age in 1968.* Columbus, OH: Ohio State University Press.

Also see

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

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

[ERM] **Intro 3** — Endogenous covariates features

[ERM] **Intro 6** — Panel data and grouped data model features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Example 8a — Random effects in one equation and endogenous covariate

Description

In this example, we show how to estimate and interpret the results of an extended regression model with a continuous outcome and a continuous endogenous covariate. We include random effects in the outcome equation but not in the equation for the continuous endogenous covariate.

Remarks and examples

In [ERM] **Example 1a**, we examined data from a fictional university that was studying the relationship between the high school grade point average (GPA) of its admitted students and their final college GPA.

Now suppose that 100 colleges have joined together in a study of the effect of high school GPA on the final college GPA of admitted students. Again, we suspect that unobserved ability affects both high school GPA and college GPA. So we treat high school GPA as an endogenous covariate. The researchers also believe that unobserved characteristics of the college are likely to affect college GPA but not high school GPA. Therefore, we allow for random effects in only the college GPA equation. Having random effects in only the main outcome equation is rare, but occasionally it corresponds to a model of interest.

Using data on the 2,000 students expected to graduate in 2010, the researchers model college GPA (`gpa`) as a function of high school GPA (`hsgpa`). In both cases, GPA is measured in 0.01 increments, and we ignore complications due to the boundary points. We also ignore that, unfortunately, the schools have a high dropout rate and that the college GPA is missing for these students, leaving the researchers with a sample of 1,372 students.

The researchers expect that the effect of high school competitiveness on college GPA is negligible once high school GPA is controlled for. So they include a ranking of the high school (`hscomp`) as an instrumental covariate for high school GPA. They include parental income measured in $10,000s, which they believe may also influence student performance, in the main model and in the model for high school GPA.

In our dataset, each observation represents one student. The variable `collegeid` uniquely identifies the 100 schools used in the study. Before we can fit a random-effects model to our data, we need to declare our grouping variable using `xtset`.

```
. use https://www.stata-press.com/data/r18/class10re
(Classes of 2010 profile)
. xtset collegeid
Panel variable: collegeid (balanced)
```

270

With the data `xtset`, we can now estimate the parameters of the model.

```
. xteregress gpa income, endogenous(hsgpa = income i.hscomp, nore)

(setting technique to bhhh)
Iteration 0:  Log likelihood =  44.332373
Iteration 1:  Log likelihood =  44.674349
Iteration 2:  Log likelihood =  44.688506
Iteration 3:  Log likelihood =  44.690548
Iteration 4:  Log likelihood =  44.691142
Iteration 5:  Log likelihood =  44.691359
Iteration 6:  Log likelihood =  44.691445
Iteration 7:  Log likelihood =  44.691483
Iteration 8:  Log likelihood =    44.6915
Iteration 9:  Log likelihood =  44.691507
(switching technique to nr)
Iteration 10: Log likelihood =  44.691511
```

Extended linear regression				Number of obs	=	1,372
Group variable: collegeid				Number of groups	=	100

				Obs per group:		
				min	=	3
				avg	=	13.7
				max	=	20

Integration method: mvaghermite				Integration pts.	=	7
				Wald chi2(2)	=	2916.69
Log likelihood = 44.691511				Prob > chi2	=	0.0000

	Coefficient	Std. err.	z	P>|z|	[95% conf. interval]	
gpa						
income	.0558709	.003765	14.84	0.000	.0484916	.0632502
hsgpa	.9390929	.0781538	12.02	0.000	.7859142	1.092272
_cons	-.5600512	.2346858	-2.39	0.017	-1.020027	-.1000755
hsgpa						
income	.0428695	.0019412	22.08	0.000	.0390649	.0466741
hscomp						
Moderate	-.1452852	.0140801	-10.32	0.000	-.1728817	-.1176887
High	-.2339232	.0235935	-9.91	0.000	-.2801656	-.1876809
_cons	3.083431	.0167567	184.01	0.000	3.050589	3.116274
var(e.gpa)	.0470832	.0024655			.0424907	.0521721
var(e.hsgpa)	.0572604	.0021862			.053132	.0617096
corr(e.hsgpa, e.gpa)	.1979973	.0870885	2.27	0.023	.0229883	.3612321
var(gpa[colle~d])	.0633532	.0095652			.0471252	.0851695

We suppressed the random effect from the equation for high school GPA by specifying `nore` within the `endogenous()` option. Therefore, no variance is reported for college random effects affecting a student's high school GPA. The variance of the random effects affecting college GPA is estimated to be 0.06.

To check for endogeneity, we need to examine only the correlation between the student-level errors in high school and college GPAs. The estimate of this correlation is 0.2, and the corresponding

test finds that it is significantly different from zero. The researchers conclude that the unobserved student-level factors that increase high school GPA tend to also increase college GPA.

Because this is a linear regression model, the coefficients can be directly interpreted. For example, the researchers expect the difference in college GPA is about 0.94 points for students with a difference of 1 point in high school GPA.

Also see

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

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

[ERM] **Intro 3** — Endogenous covariates features

[ERM] **Intro 6** — Panel data and grouped data model features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Description   Remarks and examples   Also see

Description

In [ERM] **Example 8a**, we ignored the observations that were dropped because of missing data on GPA. In this example, we show you how to fit a model with a continuous outcome, a continuous endogenous covariate, endogenous sample selection, and random effects.

Remarks and examples

In the last example, the researchers excluded students who dropped out of college because they are missing college GPA on these students. Thus they were estimating the parameters for only the population of students who graduate from college. Now let's suppose that the researchers are interested in the expected college GPA for all the students who enrolled, even those who dropped out. What would their GPA be if they had remained in school?

The researchers assumed that unobserved student ability affected both college and high school GPAs. They also suspect that unobserved ability affects the decision to stay in school, so they could have an endogenously selected sample. The researchers have data on whether the students have participated in a retention program (`program`) and whether they had a roommate from the same college (`roommate`). They use these variables in addition to high school GPA and parent's income to model whether the student graduates.

The researchers assumed that there are unobserved characteristics of the college that affects college GPA. They also assumed that unobserved college characteristics such as the availability and type of extracurricular activities and the rigor of the curriculum affect whether the students graduate. They account for these unobserved college-level factors that may affect the probability of graduating and the final college GPA of the students by including random effects in both of these equations.

```
. xteregress gpa income, endogenous(hsgpa = income i.hscomp, nore)
> select(graduate=hsgpa income i.roommate i.program)
(setting technique to bhhh)
Iteration 0:  Log likelihood = -750.88823
Iteration 1:  Log likelihood = -750.14312
Iteration 2:  Log likelihood = -750.09077
Iteration 3:  Log likelihood = -750.03772
Iteration 4:  Log likelihood = -750.03525
Iteration 5:  Log likelihood = -750.03163
Iteration 6:  Log likelihood = -750.03143
Iteration 7:  Log likelihood = -750.03082
Iteration 8:  Log likelihood = -750.03081
Iteration 9:  Log likelihood = -750.03069
(switching technique to nr)
Iteration 10: Log likelihood = -750.03068
Iteration 11: Log likelihood = -750.03067
```

```
Extended linear regression                    Number of obs    =    2,000
                                                  Selected =       1,372
                                               Nonselected =         628

Group variable: collegeid                     Number of groups =       100

                                              Obs per group:
                                                           min =         20
                                                           avg =       20.0
                                                           max =         20

Integration method: mvaghermite               Integration pts. =         7

                                              Wald chi2(2)     =   2498.14
Log likelihood = -750.03067                   Prob > chi2      =    0.0000
```

	Coefficient	Std. err.	z	P>|z|	[95% conf. interval]	
gpa						
income	.0580659	.0039428	14.73	0.000	.0503381	.0657938
hsgpa	.975956	.0719006	13.57	0.000	.8350334	1.116879
_cons	-.7193664	.2132662	-3.37	0.001	-1.13736	-.3013723
graduate						
hsgpa	1.59638	.5058428	3.16	0.002	.604946	2.587813
income	.2111094	.0256101	8.24	0.000	.1609145	.2613043
roommate						
Yes	1.16331	.0893901	13.01	0.000	.9881092	1.338512
1.program	.8719825	.0858947	10.15	0.000	.7036319	1.040333
_cons	-6.488787	1.512468	-4.29	0.000	-9.453169	-3.524405
hsgpa						
income	.0487467	.0016938	28.78	0.000	.0454269	.0520664
hscomp						
Moderate	-.1594138	.0121475	-13.12	0.000	-.1832225	-.1356051
High	-.2532709	.0195334	-12.97	0.000	-.2915557	-.2149862
_cons	3.018068	.0138501	217.91	0.000	2.990922	3.045214
var(e.gpa)	.0475351	.0024437			.0429789	.0525742
var(e.hsgpa)	.0602102	.0019041			.0565915	.0640602
corr(e.gra~e, e.gpa)	.2754647	.1003886	2.74	0.006	.0697401	.4587145
corr(e.hsgpa, e.gpa)	.1905273	.081385	2.34	0.019	.0273572	.3438079
corr(e.hsgpa, e.graduate)	.1534595	.1210009	1.27	0.205	-.0879677	.3778581
var(gpa[colle~d])	.0646465	.0097678			.0480764	.0869278
var(gra~e[col~d])	.9011305	.1745683			.6164413	1.317297
corr(gra~e[col~d], gpa[colle~d])	.2599483	.1069409	2.43	0.015	.0412395	.4548852

Now we see a random-effect variance parameter estimate for graduation and for college GPA and a correlation between these random effects. The student-level and college-level correlation parameters between the college GPA equation and graduation are significantly different from zero, so the researchers

conclude that there is endogenous sample selection. The student-level correlation between college GPA and high school GPA is also significantly different from zero, so they conclude that high school GPA is an endogenous covariate.

We can interpret the coefficients in the main equation as we did in [ERM] **Example 8a**, but now they are estimated for the population of admitted students, not the population of graduates. The estimated effect of high school GPA is slightly higher, 0.98 rather than 0.94.

Also see

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

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

[ERM] **Intro 3** — Endogenous covariates features

[ERM] **Intro 4** — Endogenous sample-selection features

[ERM] **Intro 6** — Panel data and grouped data model features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

Description

In this example, we show how to estimate and interpret the results of an extended regression model with an ordinal outcome, an endogenous treatment, and random effects.

Remarks and examples

In [ERM] **Example 6a**, we examined fictional data on the health scores of women between the ages of 25 and 30. Each woman was observed at one time point. Our outcome was an ordinal health status ranging from 1 (poor) to 5 (excellent). We estimated the average treatment effect of having health insurance on the probabilities of having each health status.

Now suppose that we conduct a fictional study where we have collected data on 1,800 women between the ages of 25 and 30 annually from 2010 to 2013. We have measured the women's health status in each year. We want to estimate the average treatment effect (ATE) of having insurance on the probability of each of the five statuses. We suspect that our model needs to account for health insurance being an endogenous treatment. We also believe that unobserved characteristics of the individual might affect both health status and whether the woman has insurance, so we include random effects in both equations.

In addition to the insurance indicator, we include an indicator for whether the woman exercises regularly and the number of years of schooling she completed (grade) as exogenous covariates in the model for health status. For our treatment model, we use grade and an indicator for whether the woman is currently working or attending school (workschool), which is excluded from the outcome model.

Before we can fit our random-effects model, we need to specify the panel structure of the data using xtset. Our panel variable is personid, the identification code for the individual. The time variable is year, and it ranges from 2010 to 2013.

```
. use https://www.stata-press.com/data/r18/womenhlthre
(Women's health status panel)

. xtset personid year
Panel variable: personid (strongly balanced)
 Time variable: year, 2010 to 2013
         Delta: 1 unit
```

With the data `xtset`, we can estimate the parameters of the model.

```
. xteoprobit health exercise grade,
> entreat(insured = grade i.workschool) vce(robust)

(setting technique to bhhh)
Iteration 0:  Log pseudolikelihood = -12272.723
Iteration 1:  Log pseudolikelihood = -12256.949
Iteration 2:  Log pseudolikelihood = -12256.539
Iteration 3:  Log pseudolikelihood = -12256.478
Iteration 4:  Log pseudolikelihood = -12256.468
Iteration 5:  Log pseudolikelihood = -12256.466
Iteration 6:  Log pseudolikelihood = -12256.465
Iteration 7:  Log pseudolikelihood = -12256.465
Iteration 8:  Log pseudolikelihood = -12256.465
Iteration 9:  Log pseudolikelihood = -12256.465
```

```
Extended ordered probit regression        Number of obs    =   7,200
Group variable: personid                  Number of groups =   1,800

                                          Obs per group:
                                                       min =       4
                                                       avg =     4.0
                                                       max =       4
Integration method: mvaghermite           Integration pts. =       7
                                          Wald chi2(4)     =  404.14
Log pseudolikelihood = -12256.465         Prob > chi2      =  0.0000
```

(Std. err. adjusted for 1,800 clusters in personid)

	Coefficient	Robust std. err.	z	P>|z|	[95% conf. interval]	
health						
insured#						
c.exercise						
No	.356811	.0521592	6.84	0.000	.2545809	.459041
Yes	.4929456	.0360086	13.69	0.000	.4223701	.5635211
insured#						
c.grade						
No	.0970783	.0198281	4.90	0.000	.0582159	.1359407
Yes	.130956	.0114576	11.43	0.000	.1084996	.1534124
insured						
grade	.29484	.0100943	29.21	0.000	.2750555	.3146245
workschool						
Yes	.5841205	.0638709	9.15	0.000	.4589358	.7093052
_cons	-3.502613	.1377291	-25.43	0.000	-3.772557	-3.232669

/health						
insured# c.cut1						
No	.4910109	.1864684			.1255395	.8564823
Yes	-.2650117	.2049759			-.6667571	.1367337
insured# c.cut2						
No	1.388273	.1810191			1.033482	1.743064
Yes	.5527565	.1908832			.1786323	.9268806
insured# c.cut3						
No	2.192588	.1794012			1.840968	2.544207
Yes	1.381288	.1806265			1.027267	1.73531
insured# c.cut4						
No	2.994727	.1873594			2.627509	3.361945
Yes	2.297709	.1731544			1.958333	2.637086
corr(e.ins~d, e.health)	.3783935	.0770755	4.91	0.000	.2183033	.5186513
var(hea~h[per~d])	.379062	.0284741			.3271676	.4391877
var(ins~d[per~d])	.2436723	.0354709			.1831887	.3241259
corr(ins~d[per~d], hea~h[per~d])	.3251756	.0721159	4.51	0.000	.1774673	.458556

The estimated correlation between the observation-level errors is 0.38. The estimated correlation between the individual-level random effects affecting health status and the individual-level random effects affecting insurance status is 0.33. Both are significantly different from zero. We conclude that insurance status is endogenous and that the unobserved person-specific factors that increase the chance of having health insurance also tend to increase the chance of being in a high health status. Additionally, the unobserved observation-level (time-varying) factors that increase the chance of having health insurance also tend to increase the chance of being in a high health status.

We see estimates of both the coefficients and the cutpoints for two equations, one for insured women (yes) and one for uninsured women (no). For both insured and uninsured, exercise and education have positive effects on health status.

We can use `estat teffects` to estimate the ATE of insurance on the probabilities of each health category.

```
. estat teffects
```

Predictive margins   Number of obs = 7,200

(Std. err. adjusted for 1,800 clusters in personid)

	Margin	Unconditional std. err.	z	P>|z|	[95% conf. interval]
ATE_Pr1 insured (Yes vs No)	-.1761541	.0279001	-6.31	0.000	-.2308372 -.1214709
ATE_Pr2 insured (Yes vs No)	-.1731894	.0227877	-7.60	0.000	-.2178525 -.1285264
ATE_Pr3 insured (Yes vs No)	-.0607013	.0127344	-4.77	0.000	-.0856602 -.0357424
ATE_Pr4 insured (Yes vs No)	.1145319	.0214062	5.35	0.000	.0725765 .1564874
ATE_Pr5 insured (Yes vs No)	.2955128	.0345022	8.57	0.000	.2278897 .3631359

We see that the treatment effect is negative on the probability of being in poor health. The treatment effect becomes more positive for each successive health status. Looking at the last line, we see that the average probability of being in excellent health in the population of women aged 25 to 30 is 0.30 greater when all women have health insurance versus when no women have health insurance.

Also see

[ERM] **eoprobit** — Extended ordered probit regression

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

[ERM] **estat teffects** — Average treatment effects for extended regression models

[ERM] **Intro 5** — Treatment assignment features

[ERM] **Intro 6** — Panel data and grouped data model features

[ERM] **Intro 9** — Conceptual introduction via worked example

Title

> **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 $\left[\,type\,\right]$ *newvar* $\left[\,if\,\right]$ $\left[\,in\,\right]$ [, *statistic treatstatistic asfmethod*
>
>   *counterfactual treatmodifier oprobitmodifier advanced*]

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

> predict $\left[\,type\,\right]$ { *stub* * | *newvarlist* } $\left[\,if\,\right]$ $\left[\,in\,\right]$ [, *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	
<u>eq</u>uation(*depvar*)	calculate results for specified dependent variable
<u>nooff</u>set	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 $E(y_i\|a < y_i < b)$, 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

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

Title

predict treatment — predict for treatment statistics

Description

`predict` has options to predict potential-outcome means, treatment effects, and treatment effects on the treated after models fit using the `entreat()` or `extreat()` option. The `predict` options are described below.

For standard use of `predict`, see

　　　　[ERM] **eregress predict**

　　　　[ERM] **eintreg predict**

　　　　[ERM] **eprobit predict**

　　　　[ERM] **eoprobit predict**

For advanced use of `predict`, see

　　　　[ERM] **predict advanced**

Also see [ERM] **estat teffects** for reports of average treatment statistics.

Syntax

You previously fit a model by using the `entreat()` or `extreat()` option,

```
eregress     y    x1 ..., ... entreat(treated = ...) ...
eintreg   yl yu  x1 ..., ... entreat(treated = ...) ...
eprobit      y    x1 ..., ... entreat(treated = ...) ...
eoprobit     y    x1 ..., ... entreat(treated = ...) ...
xteregress   y    x1 ..., ... entreat(treated = ...) ...
xteintreg yl yu  x1 ..., ... entreat(treated = ...) ...
xteprobit    y    x1 ..., ... entreat(treated = ...) ...
xteoprobit   y    x1 ..., ... entreat(treated = ...) ...

eregress     y    x1 ..., ... extreat(treated) ...
eintreg   yl yu  x1 ..., ... extreat(treated) ...
eprobit      y    x1 ..., ... extreat(treated) ...
eoprobit     y    x1 ..., ... extreat(treated) ...
xteregress   y    x1 ..., ... extreat(treated) ...
xteintreg yl yu  x1 ..., ... extreat(treated) ...
xteprobit    y    x1 ..., ... extreat(treated) ...
xteoprobit   y    x1 ..., ... extreat(treated) ...
```

In these cases, `predict` has extra features. `predict`'s extra syntax for these features is

　`predict` [*type*] *newvar* [*if*] [*in*], *treatstatistic* [*treatmodifier oprobitmodifier*]

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

> predict [*type*] { *stub** | *newvarlist* } [*if*] [*in*] , *treatstatistic* [*treatmodifier*
>
> *oprobitmodifier*]

treatstatistic	Description
<u>pomean</u>	potential-outcome mean (POM)
te	treatment effect (TE)
tet	treatment effect on the treated (TET)

treatmodifier	Description
<u>tl</u>evel(*#*)	treatment level for which *treatstatistic* is calculated

may be specified as a value recorded in variable treated, such as 1, 2, . . . or such as 1, 5, . . . , depending on the values recorded.

may also be specified as #1, #2, . . . , meaning the first, second, . . . values recorded in treated.

oprobitmodifier	Description
<u>outl</u>evel(*#*)	ordered outcome for which *treatstatistic* is calculated

When used after models fit with eoprobit or xteoprobit, *treatstatistic* is calculated for the specified outcome, or for the first outcome if you do not specify otherwise.

outlevel(*#*) specifies the outcome for which statistics are to be calculated. *#* is specified in the same way as with tlevel(), but the meaning is different. In the case of outlevel(), you are specifying the outcome, not the treatment level.

Options

The options for the statistic to be calculated—pomean, te, and tet—are mutually exclusive. You calculate one treatment statistic per predict command.

pomean calculates the POMs for each treatment level. The POMs are the expected value of y that would have been observed if everyone was assigned to each of the treatment levels.

If there were two treatment levels (a control and a treatment), you would type

```
. predict pom1 pom2, pomean
```

If there were three levels, you would type

```
. predict pom1 pom2 pom3, pomean
```

pomean can alternatively be used with tlevel() to produce individual POMs:

```
. predict pom1, pomean tlevel(#1)
. predict pom2, pomean tlevel(#2)
```

If you have fit the model using eoprobit or xteoprobit, the POMs calculated for the examples above would be for y's first outcome. You can change that. See *Predicting treatment effects after eoprobit and xteoprobit* in *Remarks and examples* below.

te calculates the TEs for each treatment level. The TEs are the differences in the POMs. For instance, if there were two treatment levels—a control and a treatment—there would be one treatment effect and it would be pom2-pom1. If there were three levels, there would be two treatment effects, pom2-pom1 and pom3-pom1.

If there were two treatment levels—a control and a treatment—you would type

```
. predict te2, te
```

If there were three levels, you would type

```
. predict te2 te3, te
```

te can alternatively be used with tlevel() to produce individual TEs:

```
. predict te2, te tlevel(#2)
. predict te3, te tlevel(#3)
```

If you have fit the model using eoprobit or xteoprobit, the TEs calculated for the examples above would be for y's first outcome. You can change that. See *Predicting treatment effects after eoprobit and xteoprobit* in *Remarks and examples* below.

tet calculates the TETs. The TETs are the differences in the POMs conditioned on treatment level.

If there were two treatment levels—a control and a treatment—you would type

```
. predict tet2, tet
```

If there were three levels, you would type

```
. predict tet2 tet3, tet
```

tet can alternatively be used with tlevel() to produce individual TETs:

```
. predict tet2, tet tlevel(#2)
. predict tet3, tet tlevel(#3)
```

If you have fit the model using eoprobit or xteoprobit, the TETs calculated for the examples above would be for y's first outcome. You can change that. See *Predicting treatment effects after eoprobit and xteoprobit* in *Remarks and examples* below.

tlevel(#) is optionally used with pomean, te, or tet. Its use is illustrated above.

outlevel(#) is optionally used with pomean, te, or tet with models fit by eoprobit and xteoprobit. See *Predicting treatment effects after eoprobit and xteoprobit* in *Remarks and examples* below.

Remarks and examples

For an example of predict with treatment effects, see [ERM] **Intro 9**.

Remarks are presented under the following headings:

Predicting treatment effects after eregress, eintreg, xteregress, and xteintreg

eregress, eintreg, xteregress, and xteintreg concern models with a continuous outcome variable. In eregress and xteregress models, y_i is observed. In eintreg and xteintreg models, y_i is not observed directly, but it is known that $yl_i \leq y_i \leq yu_i$.

Thus, the treatment statistics are expressed in the units of y. If y is blood pressure, the units are presumably mmHG. POMs are in mmHG. TEs and TETs are differences in blood pressure expressed in mmHG.

Predicting treatment effects after eprobit and xteprobit

eprobit and xteprobit concern models with binary outcomes, and predictions are in terms of the probability of a positive outcome. Thus, POMs are probabilities. TEs and TETs are differences in probabilities.

Predicting treatment effects after eoprobit and xteoprobit

eoprobit and xteoprobit concern models with ordinal outcome variables, and predictions are in terms of the probabilities—the probability of each outcome.

Treatment statistics are calculated on the basis of probabilities of outcomes. Thus, POMs are probabilities. TEs and TETs are differences in probabilities.

We want probabilities and differences in probabilities, but you need to specify which probability. The probability for the first outcome? The second?

If you do not specify which and simply type

```
. predict pom1 pom2 pom3, pomean
```

then the POMs are calculated for the first outcome, what eoprobit and xteoprobit call out-level(#1). If you wanted to obtain the POMs for outlevel(#2), you would type

```
. predict pom1 pom2 pom3, pomean outlevel(#2)
```

If you wanted them for outlevel(#3), you would type

```
. predict pom1 pom2 pom3, pomean outlevel(#3)
```

The same logic applies to calculating TE and TET with the te and tet options. outlevel(#1) is used unless you specify otherwise.

Methods and formulas

See *Methods and formulas* in [ERM] **eintreg**, [ERM] **eoprobit**, [ERM] **eprobit**, and [ERM] **eregress**.

Also see

Title

Triangularize — How to triangularize a system of equations

Description   Remarks and examples   Also see

Description

ERMs allow endogenous covariates, but they must form a triangular system, also known as a recursive system. Said differently, ERMs do not allow simultaneous causation. This was explained for simple cases in [ERM] **Intro 3**. How to triangularize complicated systems is described below.

Remarks and examples

The day will come when you try to fit a model and the ERM command responds with the following error:

```
. eregress y w1 w2 w3 x1 x2,
>           endogenous(w1 = w2  z1 z2 x1 x2 x5, nomain)
>           endogenous(w2 = w1  z1 z3 x1 x2 x5, nomain)
>           endogenous(w3 = w1  z4 z5 x1 x2 x5, nomain)
>           endogenous(z1 =     z5   x1 x2 x4, nomain)
 endogenous variables do not form a triangular system
    The problem may be fixable.  See triangularizing the system.
 r(459);
```

The error can even occur in simple models:

```
. eregress y w1 x2 x3, endogenous(w1 = y z1 x2, nomain)
 endogenous variables do not form a triangular system
     The problem may be fixable.  See triangularizing the system.
 r(459);
```

The error message says the problem may be fixable. We explain below how to find the problem, how to determine whether it is fixable, and how to fix it when it is.

Remarks are presented under the following headings:

What is a triangular system?
Triangularizing nontriangular systems
You can only triangularize linear equations
Options entreat(), select(), and tobitselect() also add endogenous variables
Workarounds involving the main equation
Why the above is a workaround and not a fix

What is a triangular system?

ERMs require that the endogenous variables in the model being fit form a triangular system. The endogenous variables include the dependent variable in the main equation and the dependent variables in the endogenous() options. In addition, the options entreat(), select(), and tobitselect() add endogenous variables, but we will cover those options later.

289

The endogenous variables are y, w1, w2, w3, and z1 in the model

```
. eregress y w1 w2 w3 x1 x2 x5,                               ///
           endogenous(w1 =    z1 z2 x1 x2 x5, nomain)  ///
           endogenous(w2 = w1  z1 z3 x1 x2 x5, nomain)  ///
           endogenous(w3 = w1  z4 z5 x1 x2 x5, nomain)  ///
           endogenous(z1 =    z5    x1 x2 x4, nomain)
```

The system that needs to be triangular is y, w1, w2, w3, and z1. That system is

Endogenous variable	which depends on the endogenous variable(s)
y	w1 w2 w3
w1	z1
w2	w1 z1
w3	w1
z1	*(none)*

A system is triangular when the dependencies can be ordered such that each endogenous variable is already defined before it is used as an explanatory variable. The system, in order, is

Endogenous variable	which depends on the endogenous variable(s)
z1	*(none)*
w1	z1
w3	w1
w2	w1 z1
y	w1 w2 w3

The system is in order and triangular because

1. Endogenous variable z1 depends on no other endogenous variables.

2. Endogenous variable w1 depends on z1, and z1's definition has already been listed.

3. Endogenous variable w3 depends on w1, and w1's definition has already been listed.

4. Endogenous variable w2 depends on w1 and z1, and their definitions have already been listed.

5. Endogenous variable y depends on w1, w2, and w3, and their definitions have already been listed.

When the system is triangular, ERMs can fit the model.

Triangularizing nontriangular systems

Consider the model

```
. eregress y w1 w2 w3 x1 x2 x5,
>           endogenous(w1 = w2  z1 z2 x1 x2 x5, nomain)
>           endogenous(w2 = w1  z1 z3 x1 x2 x5, nomain)
>           endogenous(w3 = w1  z4 z5 x1 x2 x5, nomain)
>           endogenous(z1 =    z5    x1 x2 x4, nomain)
 endogenous variables do not form a triangular system
     The problem may be fixable.  See triangularizing the system.
 r(459);
```

The ERM command has already told us that the system defined by this model is not triangular. Thus, if we try to order the definitions as we did above, we will not be successful. Where we run into difficulties, however, will tell us where the problem is.

The endogenous variables in this model are y, w1, w2, w3, and z1. Their definitions in the order in which they appear in the command are

Endogenous variable	which depends on the endogenous variable(s)
y	w1 w2 w3
w1	w2 z1
w2	w1 z1
w3	w1
z1	*(none)*

The definitions in as near to the correct order as we can get them are

Endogenous variable	which depends on the endogenous variable(s)
z1	*(none)*
w1	z1 w2 ← problem here
w2	w1 z1
w3	w1
y	w1 w2 w3

The problem appears in the second line where w1 is defined in terms of z1 and w2: w2 has not yet been defined. Obviously, we need to put its definition above that for w1. However, if we move the definition of w2 above that of w1, we still have a problem: w2 depends on z1 and w1, and now w1 has not yet been defined!

You might notice that there are three endogenous variables involved in the problem—w1, w2, and w3—but just focus on the first pair of definitions that cause the problem. It does not matter which two of the three they are. In our case, they are

```
endogenous(w1 = w2  z1 z2 x1 x2 x5, nomain)
endogenous(w2 = w1  z1 z3 x1 x2 x5, nomain)
```

As we said in [ERM] **Intro 3**, there is a workaround for the problem when both equations are linear, as they are in this case. The workaround is

> When the simultaneous-causation problem occurs in linear equations defined by endogenous() options, remove the endogenous variable from one equation and substitute for it all the variables from the removed variable's equation except, of course, the variable you just removed.

The workaround in this case either

1. Removes w2 from the first equation and substitutes "z1 z3 x1 x2 x5" for it.

2. Removes w1 from the second equation and substitutes "z1 z2 x1 x2 x5" for it.

It does not matter which we do.

To remind you, we are fixing the first equation:

```
endogenous(w1 = w2  z1 z2 x1 x2 x5, nomain)
```

When we remove w2 and substitute "z1 z3 x1 x2 x5", we obtain

```
                    z1 z3 x1 x2 x5    z1 z2 x1 x2 x5
```

Now, we need to remove the duplicates. Removing them, we have

```
             z3              z1 z2 x1 x2 x5
```

Thus, the first equation becomes

```
endogenous(w1 = z3 z1 z2 x1 x2 x5, nomain)
```

We can now try fitting the model again:

```
. eregress y w1 w2 w3 x1 x2 x5,                           ///
        endogenous(w1 =    z3 z1 z2 x1 x2 x5, nomain)  ///
        endogenous(w2 = w1  z1 z3 x1 x2 x5, nomain)    ///
        endogenous(w3 = w1  z4 z5 x1 x2 x5, nomain)    ///
        endogenous(z1 =    z5    x1 x2 x4, nomain)
```

When we try to fit the model, it will be successful or it will repeat the same error we saw earlier:

```
endogenous variables do not form a triangular system
    The problem may be fixable.  See triangularizing the system.
r(459);
```

In this case, the model will be successfully fit. If you do get the error, repeat the process. Remove the problems one at a time.

You can only triangularize linear equations

The rule is

> When the simultaneous-causation problem occurs in *linear* equations defined by endogenous() options, remove the endogenous variable from one equation and substitute for it all the variables from the removed variable's equation except, of course, the variable you just removed.

Triangularization involves a pair of equations that must both be linear. In the example above, both were linear:

```
endogenous(w1 = w2  z1 z2 x1 x2 x5, nomain)
endogenous(w2 = w1  z1 z3 x1 x2 x5, nomain)
```

They would not have both been linear if either had been fit by probit or oprobit. If one or both of the equations had been

```
endogenous(w1 = w2  z1 z2 x1 x2 x5, nomain probit)
endogenous(w2 = w1  z1 z3 x1 x2 x5)
```

or

```
endogenous(w1 = w2  z1 z2 x1 x2 x5, nomain)
endogenous(w2 = w1  z1 z3 x1 x2 x5, nomain oprobit)
```

there would have been no solving the simultaneous-causation problem.

This linearity requirement applies only to the two equations directly involved. Other equations can be nonlinear and there will be no issue. The workaround we outlined would have worked just as well had the model been

```
. eregress y w1 w2 w3 x1 x2 x5,                        ///
            endogenous(w1 = w2  z1 z2 x1 x2 x5, nomain)        ///
            endogenous(w2 = w1  z1 z3 x1 x2 x5, nomain)        ///
            endogenous(w3 = w1  z4 z5 x1 x2 x5, nomain probit)  ///
            endogenous(z1 =     z5    x1 x2 x4, nomain probit)
```

or even

```
. eprobit y w1 w2 w3 x1 x2 x5,                         ///
            endogenous(w1 = w2  z1 z2 x1 x2 x5, nomain)        ///
            endogenous(w2 = w1  z1 z3 x1 x2 x5, nomain)        ///
            endogenous(w3 = w1  z4 z5 x1 x2 x5, nomain probit)  ///
            endogenous(z1 =     z5    x1 x2 x4, nomain probit)
```

Options entreat(), select(), and tobitselect() also add endogenous variables

The example above contained repeated uses of the `endogenous()` option. When you make the list of endogenous variables, you must also include the dependent variables *treated* and *selected* from the options

> `entreat(`*treated*`= . . . )`
>
> `select(`*selected*`= . . . )`
>
> `tobitselect(`*selected*`= . . . )`

The above options make *treated* and *selected* endogenous. Unlike with the `endogenous()` option, however, the variables are not automatically added to the main equation even if you do not specify `nomain`.

These three options are nonlinear. If the simultaneous-causation problem involves equations created by these options, then there is no workaround for the simultaneous-causation problem.

Workarounds involving the main equation

The example of the simultaneous-causation problem involved two equations defined by `endogenous()` options. The problem could also occur when one of the equations is the main equation. In [ERM] **Intro 3**, we discussed problems involving the main equation as if they were different from simultaneous causation, but they are not. It is the same problem that has the same workaround, but with an important difference.

In workarounds involving equations defined by `endogenous()` equations, the workaround may be applied to either equation.

In workarounds involving the main equation and an `endogenous()` equation, the workaround must be applied to the `endogenous()` equation.

> When the simultaneous-causation problem involves the main equation fit by `eregress` and an `endogenous()` linear equation, remove the dependent variable from the `endogenous()` equation and substitute for it all the variables from the main equation except, of course, the variable you just removed.

Also notice that this rule applies only to main equations fit by `eregress`. What about `eintreg`, `eprobit`, and `eoprobit`?

The simultaneous-causation problem does not arise in models fit by `eintreg`. There is no way you could include `eintreg`'s dependent variables as explanatory variables in another equation.

The simultaneous-causation problem can arise in models fit by `eprobit` and `eoprobit`, but those are nonlinear equations, and that means you cannot apply the workaround. The workaround requires that both equations be linear.

The main equation must be linear if it is one of the two equations involved in the simultaneous-causation problem. Otherwise, the main equation is not required to be linear.

Why the above is a workaround and not a fix

It is a detail, but you may have noticed that we provided a workaround and not a fix. The purpose of ERMs is to obtain valid estimates of the coefficients of the main equation—its structural parameters—in light of lots of complications. It so happens that ERMs produce estimates of structural parameters for all the other equations if the system is truly triangular. That is not important, but it is true.

When you triangularize a nontriangular system, ERMs no longer produce estimates of the structural parameters for the equations that you modify. They produce estimates of the reduced-form equation, and that is sufficient. Valid estimates of the reduced-form equation ensures that estimates of the coefficients in the main equation are estimates of its structural parameters.

Thus, what we provided is a workaround, not a fix. If you use the workaround, do not interpret any equations modified as estimates of their structural parameters.

Also see

[ERM] **Intro 3** — Endogenous covariates features

Glossary

average structural function. The average structural function (ASF) is used to calculate predicted values of ERMs.

The ASF averages out the heterogeneity caused by the endogeneity from a conditional mean or a conditional probability in a model with endogenous covariates. Applying the ASF to a conditional mean produces an average structural mean (ASM). Applying the ASF to a conditional probability produces an average structural probability (ASP). Contrasts of ASMs or ASPs produced by a covariate change define a causal structural effect. Blundell and Powell (2003, 2004) and Wooldridge (2005, 2014) are seminal papers that define and extend the ASF. See Wooldridge (2010, 22–24) for a textbook introduction.

average structural mean. The average structural mean (ASM) is the result of applying the average structural function to a conditional mean.

average structural probability. The average structural probability (ASP) is the result of applying the average structural function to a conditional probability.

average treatment effect. See *treatment effects*.

average treatment effect on the treated. See *treatment effects*.

average treatment effect on the untreated. See *treatment effects*.

binary variable. A binary variable is any variable that records two values, the two values representing false and true, such as whether a person is sick. We usually speak of the two values as being 0 and 1 with 1 meaning true, but Stata requires merely that 0 means false and nonzero and nonmissing mean true. Also see *continuous variable*, *categorical variable*, and *interval variable*.

categorical variable. A categorical variable is a variable that records the category number for, say, lives in the United States, lives in Europe, and lives in Asia. Categorical variables play no special role in this manual, but ordered categorical variables do. The example given is unordered. The categories United States, Europe, and Asia have no natural ordering. We listed the United States first only because the author of this manual happens to live in the United States.

The way we use the term, categorical variables usually record two or more categories, and the term binary variable is used for categorical variables having two categories.

We usually speak of categorical variables as if they take on the values 1, 2, ... Stata does not require that. However, the values do need to be integers.

censored, **left-censored**, **right-censored**, and **interval-censored**. Censoring involves not observing something but knowing when and where you do not observe it.

For instance, sometimes patients, subjects, or units being studied—observations in your dataset—have values equal to missing. Such observations are said to be censored when there is a reason they are missing. A variable is missing because a potential worker chooses not to work, because a potential patient chooses not to be a patient, because a potential subject was not prescribed the treatment, etc. Such censored outcomes cause difficulty when there is an unobserved component to the reason they are censored that is correlated with the outcome being studied. ERM option `select()` addresses these issues.

Another type of censoring—interval-censoring—involves not observing a value precisely but knowing its range. You do not observe blood pressure, but you know it is in the range 120 to 140. Or you know it is less than 120 or greater than 160. ERM command `eintreg` fits models in which the dependent variable is interval-censored.

Left-censoring is open-ended interval-censoring in which measurements below a certain value are unobserved. Blood pressure is less than 120.

Right-censoring is open-ended interval-censoring in which measurements above a certain value are unobserved. Blood pressure is above 160.

conditional mean. The conditional mean of a variable is the expected value based on a function of other variables. If y is a linear function of x_1 and x_2—$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 +$ noise—then the conditional mean of y for $x_1 = 2$ and $x_2 = 4$ is $\beta_0 + 2\beta_1 + 4\beta_2$.

confounding variable, confounder. A confounding variable is an omitted explanatory variable in a model that is correlated with variables included in the model. The fitted coefficients on the observed variables will include the effect of the variables, as intended, plus the effect of being correlated with the omitted variable.

Confounders are often omitted from the model because they are unobserved. See [ERM] **Intro 3**.

continuous variable. A continuous variable is a variable taking on any value on the number line. In this manual, however, we use the term to mean the variable is not a binary variable, not a categorical variable, and not an interval variable.

counterfactual. The result that would be expected from a thought experiment that assumes things counter to what are currently true. What would be the average income if everyone had one more year of schooling? What would be the effect of an experimental medical treatment if the treatment were made widely available? Stata's `margins` command produces statistical answers to these kinds of thought experiments and reports standard errors as well.

counterfactual predictions. Counterfactual predictions are used when you have endogenous covariates in your main equation and you wish to estimate either counterfactuals or the effect on the outcome of changing the values of covariates. They are obtained using `predict` options `base()` and `fix()`.

covariate. A covariate is a variable appearing on the right-hand side (RHS) of a model. Covariates can be exogenous or endogenous, but when the term is used without qualification, it usually means exogenous covariate. Covariates are also known as explanatory variables. Also see *endogenous covariate* and *exogenous covariate*.

cross-sectional data. Cross-sectional data refers to data collected over a set of individuals, such as households, firms, or countries sampled from a population at a given point in time.

dependent variable. A dependent variable is a variable appearing on the left-hand side of an equation in a model. It is the variable to be explained. Every equation of a model has a dependent variable. The term "the dependent variable" is often used in this manual to refer to the dependent variable of the main equation. Also see [ERM] **Intro 3**.

endogenous and exogenous treatment assignment. See *treatment assignment*.

endogenous covariate. An endogenous covariate is a covariate appearing in a model 1) that is correlated with omitted variables that also affect the outcome; 2) that is measured with error; 3) that is affected by the dependent variable; or 4) that is correlated with the model's error. See [ERM] **Intro 3**.

endogenous sample selection. Endogenous sample selection refers to situations in which the subset of the data used to fit a model has been selected in a way correlated with the model's outcome.

Mechanically, the subset used is the subset containing nonmissing values of variables used by the model. A variable is unobserved—contains missing values—because a potential worker chooses not to work, because a potential patient chooses not to be a patient, because a potential subject was not prescribed the treatment, etc. Such censored outcomes cause difficulty when there is an

unobserved component to the reason they are censored that is correlated with the outcome being studied.

ERM option `select()` can address these issues when the dataset contains observations for which the dependent variable was missing.

error. Error is the random component (residual) appearing at the end of the equations in a model. These errors account for the unobserved information explaining the outcome variable. Errors in this manual are written as *e.depvarname*, such as $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + e.y$.

exogenous covariate. An exogenous covariate is a covariate that is uncorrelated with the error term in the model. See [ERM] **Intro 3**.

explanatory variable. Explanatory variable is another word for covariate.

extended regression models. Extended regression models (ERMs) are generalized structural equation models that allow identity and probit links and Gaussian, binomial, and ordinal families for the main outcome. They extend interval regression, ordered probit, probit, and linear regression models by accommodating endogenous covariates, nonrandom and endogenous treatment assignment, endogenous sample selection, and random effects.

individual-level treatment effect. An individual-level treatment effect is the difference in the individuals outcome that would occur when given one treatment instead of another. It is the difference between two potential outcomes for the individual. The blood pressure after taking a pill minus the blood pressure were the pill not taken is the individual-level treatment effect of the pill on blood pressure.

informative missingness. See *missingness*.

instrument. Instrument is an informal word for instrumental variable.

instrumental variable. An instrumental variable is a variable that affects an endogenous covariate but does not affect the dependent variable. See [ERM] **Intro 3**.

interval measurement. Interval measurement is a synonym for interval-censored. See *censored*.

interval variable. An interval variable is actually a pair of variables that record the lower and upper bounds for a variable whose precise values are unobserved. `ylb` and `yub` might record such values for a variable y. Then it is known that, for each observation i, $ylb_i \leq y \leq yub_i$. ERM estimation command `eintreg` fits such models. Also see *censored*.

interval-censored. See *censored*.

left-hand-side (LHS) variable. A left-hand-side variable is another word for dependent variable.

longitudinal data. Longitudinal data is another term for panel data. See also *panel data*.

loss to follow-up. Subjects are lost to follow-up if they do not complete the course of the study for reasons unrelated to the event of interest. For example, loss to follow-up occurs if subjects move to a different area or decide to no longer participate in a study. Loss to follow-up should not be confused with administrative censoring. If subjects are lost to follow-up, the information about the outcome these subjects would have experienced at the end of the study, had they completed the study, is unavailable.

main equation. The main equation in an ERM is the first equation specified, the equation appearing directly after the `eregress`, `eintreg`, `eprobit`, `eoprobit`, `xteregress`, `xteintreg`, `xteprobit`, or `xteoprobit` command. The purpose of ERMs is to produce valid estimates of the coefficients in the main equation, meaning the structural coefficients, in the presence of complications such as endogeneity, selection, treatment assignment, or random effects.

measurement error, measured with error. A variable measured with error has recorded value equal to $x + \epsilon$, where x is the true value. The error is presumably uncorrelated with all other errors in the model. In that case, fitted coefficients will be biased toward zero. See [ERM] **Intro 3**.

missing at random (MAR). See *missingness*.

missing completely at random (MCAR). See *missingness*.

missing not at random (MNAR). See *missingness*.

missingness. Missingness refers to how missing observations in data occur. The categories are 1) missing not at random (MNAR), 2) missing at random (MAR), and 3) missing completely at random (MCAR).

In what follows we will refer to missing observations to mean not only observations entirely missing from a dataset but also the omitted observations because of missing values when fitting models.

MNAR observations refer to cases in which the missingness depends on the outcome under study. The solution in this case is to model that dependency. When observations are missing because of missing values, ERM option select() can be used to model the missingness.

MAR observation refer to cases in which the missingness does not depend on the outcome under study but does depend on other variables correlated with the outcome. The solution for some of the problems raised is to include those other variables as covariates in your model. Importantly, you do not need to model the reason for missingness.

MCAR observations are just that and obviously not a problem other than to cause loss of efficiency.

The MNAR and MAR cases are known jointly as informative missingness.

multivalued treatment. A multivalued treatment is a treatment with more than two arms. See *treatment arms*.

observational data. Observational data are data collected over which the researcher had no control. The opposite of observational data is experimental data. Use of observational data often introduces statistical issues that experimental data would not. For instance, in a treatment study based on observational data, researchers had no control over treatment assignment; thus the treatment assignment needs to be modeled.

omitted variables. Omitted variables is an informal term for covariates that should appear in the model but do not. They do not because they are unmeasured, because of ignorance or other reasons. Problems arise when the variables that are not omitted are correlated with the omitted variables.

ordered categorical variable. An ordered categorical variable is a categorical variable in which the categories can be ordered, such as healthy, sick, and very sick. Actually recorded in the variable are integers such as 1, 2, and 3. The integers need not be sequential, but they must reflect the ordering. Also see *binary variable* and *continuous variable*.

outcome variable. See *dependent variable*.

panel data. Panel data are data in which the same units were observed over multiple periods. The units, called panels, are often firms, households, or patients who were observed at several points in time. In a typical panel dataset, the number of panels is large, and the number of observations per panel is relatively small.

potential outcome. Potential outcome is a term used in the treatment-effects literature. It is the outcome an individual would have had if given a specific treatment. Individual in this case means conditional on the individual's covariates, which are in the main equation in models fit by ERMs. It is the outcome that would have been observed for that individual. For instance, each patient in

a study has one potential blood pressure after taking a pill and another had he or she not taken it. Also see *treatment effects*.

potential-outcome means. Potential-outcome means (POMs) is a term used in the treatment-effects literature. They are the means (averages) of potential outcomes. The average treatment effect (see treatment effects) is the difference between the potential-outcome mean for treated and untreated over the population.

random-effects model. A random-effects model for panel data treats the panel-specific errors for each equation as random variables drawn from a population with zero mean and constant variance. The regressors not distinctly specified as endogenous must be uncorrelated with the random effects for the estimates to be consistent.

recursive (structural) model. ERMs fit recursive models. A model is not recursive when one endogenous variable depends (includes its equation) on another endogenous variable that depends on the first. Said in symbols, when A depends on B, which depends on A. A model is also not recursive when A depends on B depends on C, which depends on A, and so on. See [ERM] **Triangularize**.

reverse causation and simultaneous causation. We use the term reverse causation in this manual when the dependent variable in the main equation of an ERM affects a covariate as well as when the covariate affects the dependent variable. Stressed persons may be physically unhealthy because they are stressed and further stressed because they are unhealthy. When a covariate suffers from reverse causation, the solution is to make it endogenous and find instruments for it.

Our use of the term reverse causation is typical of how it is used elsewhere. Reverse causation is a reason to make a variable endogenous. Reverse causation is discussed in [ERM] **Intro 3**.

The term simultaneous causation is sometimes used as a synonym for reverse causation elsewhere, but we draw a distinction. We use the term when two already endogenous variables affect each other. Simultaneous causation is discussed in [ERM] **Triangularize**.

right-hand-side (RHS) variable. A right-hand-side variable is another word for covariate.

sample selection. Sample selection is another term for endogenous sample selection.

selection. Selection is another term for endogenous sample selection.

selection on unobservables. Selection on unobservables is another term for endogenous sample selection.

simultaneous causation. See *recursive (structural) model*.

simultaneous system. A simultaneous system is a multiple-equation model in which dependent variables can affect each other freely. The equation for y1 could include y2, and the equation for y2 include y1. ERMs cannot fit simultaneous systems. Because the focus of ERMs is on one equation in particular—the main equation—you can substitute the covariates for y1 into the y2 equation to form the reduced-form result and still obtain estimates of the structural parameters of the y1 equation. In this manual, we discuss this issue using the terms reverse causation and recursive (structural) model. In the manual, it is discussed in [ERM] **Triangularize**.

strongly balanced. A longitudinal or panel dataset is said to be strongly balanced if each panel has the same number of observations and the observations for different panels were all made at the same times.

TE. See *treatment effect*.

tobit estimator. Tobit is an estimation technique for dealing with dependent variables that are censored. The classic tobit model dealt with left-censoring, in which the outcome variable was recorded as zero if it would have been zero or below. The estimator has since been generalized to dealing

with models in which observations can be left-censored, right-censored, or interval-censored. See *censored*.

treatment. A treatment is a drug, government program, or anything else administered to a patient, job seeker, etc., in hopes of improving an outcome.

treatment arms. Sometimes, experiments are run on more than one *treatment* simultaneously. Each different treatment is called an arm of the treatment. The controls (those not treated) are also an arm of the treatment.

treatment assignment. Treatment assignment is the process by which subjects are assigned to a *treatment arm*. That process can be endogenous or exogenous, meaning that the random component (error) in the assignment is correlated or is not correlated with the outcomes of the treatments. It is often endogenous because doctors assign subjects or subjects choose based in part on unobserved factors correlated with the treatment's outcome.

treatment effects. A treatment effect (TE) is the effect of a treatment in terms of a measured outcome such as blood pressure, ability to walk, likelihood of finding employment, etc. The statistical problem is to measure the effect of a treatment in the presence of complications such as censoring, treatment assignment, and so on.

ERMs fit treatment-effect models when one of the options `entreat()` or `extreat()` is specified for endogenous or exogenous treatment assignment. Meanwhile, the outcome model is specified in the main equation.

The TE is, for each person, the difference in the predicted outcomes based on the covariates in the main equation given that treatment is locked at treated or untreated.

The treatment effect on the treated (TET) is, for each person who was treated, the difference in the predicted outcomes based on the covariates in the main equation and the fact that they were assigned to or choose to be treated.

The treatment effect on the untreated (TEU) is, for each person who was not treated, the difference in predicted outcomes based on the covariates in the main equation and the fact that they were assigned to or choose not to be treated.

The average treatment effect (ATE) is an estimate of the average effect in a population after accounting for statistical issues.

The average effect on the treated (ATET) is an estimate of the average effect that would have been observed for those who were in fact treated in the data.

The average effect on the untreated (ATEU) is an estimate of the average effect that would have been observed for those who were in fact not treated in the data.

triangular system. See *recursive (structural) model*.

unbalanced data. A longitudinal or panel dataset is said to be unbalanced if each panel does not have the same number of observations. See also *weakly balanced* and *strongly balanced*.

weakly balanced. A longitudinal or panel dataset is said to be weakly balanced if each panel has the same number of observations but the observations for different panels were not all made at the same times.

References

Blundell, R. W., and J. L. Powell. 2003. Endogeneity in nonparametric and semiparametric regression models. In *Advances in Economics and Econometrics: Theory and Applications, Eighth World Congress*, ed. M. Dewatripont, L. P. Hansen, and S. J. Turnovsky, vol. 2, 312–357. Cambridge: Cambridge University Press.

——. 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. 2005. Unobserved heterogeneity and estimation of average partial effects. In *Identification and Inference for Econometric Models: Essays in Honor of Thomas Rothenberg*, ed. D. W. K. Andrews and J. H. Stock, 27–55. New York: Cambridge University Press.

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

——. 2014. Quasi-maximum likelihood estimation and testing for nonlinear models with endogenous explanatory variables. *Journal of Econometrics* 182: 226–234. https://doi.org/10.1016/j.jeconom.2014.04.020.

Subject and author index

See the combined subject index and the combined author index in the *Stata Index*.