Title

stata.com

Intro 3 — Endogenous covariates features

Description Remarks and examples Also see

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

stata.com

Remarks are presented under the following headings:

What are endogenous and exogenous covariates?

Consider the model

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

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:

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

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

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

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 $\texttt{w1}^2$. Years of schooling after high school would increase y by $\beta_2\texttt{w1} + \beta_3\texttt{w1}^2$.

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