Syntax

heckoprobit  depvar  indepvars  [if]  [in]  [weight],
  select([depvar_s =]  varlist_s  [,  noconstant  offset(varname_o)])  [options]

options  Description

Model
  *select()  specify selection equation: dependent and independent
           variables; whether to have constant term and offset variable
  offset(varname)  include varname in model with coefficient constrained to 1
  constraints(constraints)  apply specified linear constraints
  collinear  keep collinear variables

SE/Robust
  vce(vcetype)  vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife

Reporting
  level(#)  set confidence level; default is level(95)
  first  report first-step probit estimates
  noheader  do not display header above parameter table
  nofootnote  do not display footnotes below parameter table
  noconsreport  do not display constraints
  display_options  control column formats, row spacing, line width, display of omitted
                   variables and base and empty cells, and factor-variable labeling

Maximization
  maximize_options  control the maximization process; seldom used
  coeflegend  display legend instead of statistics

*select() is required.

The full specification is select([depvar_s =]  varlist_s  [,  noconstant  offset(varname_o)]).

indepvars and varlist_s may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, depvar_s, and varlist_s may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce(), first, and weights are not allowed with the svy prefix; see [SVY] svy.
pweights, fweights, and iweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.

See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.
heckoprobit — Ordered probit model with sample selection

Menu
Statistics > Sample-selection models > Ordered probit model with selection

Description

heckoprobit fits maximum-likelihood ordered probit models with sample selection.

Options

Model

```bash
select([ depvar_s = ] varlist_s [, noconstant offset(varname_o) ]). specifies the variables and options for the selection equation. It is an integral part of specifying a selection model and is required. The selection equation should contain at least one variable that is not in the outcome equation.

If depvar_s is specified, it should be coded as 0 or 1, 0 indicating an observation not selected and 1 indicating a selected observation. If depvar_s is not specified, observations for which depvar is not missing are assumed selected, and those for which depvar is missing are assumed not selected.

noconstant suppresses the selection constant term (intercept).

offset(varname_o) specifies that selection offset varname_o be included in the model with the coefficient constrained to be 1.

offset(varname), constraints(constraints), collinear; see [R] estimation options.
```

SE/Robust

```bash
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

```bash
level(#); see [R] estimation options.

first specifies that the first-step probit estimates of the selection equation be displayed before estimation.

noheader suppresses the header above the parameter table, the display that reports the final log-likelihood value, number of observations, etc.

nofootnote suppresses the footnotes displayed below the parameter table.

nocnsreport; see [R] estimation options.
```

display_options: noduplicates, vsquish, noemptycells, baselevels, allbaselevels, nolabels, fvwrap(#), fvwrapon(style), cformat(%fmt), pformat(%fmt), sformat(%fmt), and nolstretch; see [R] estimation options.
heckoprobit — Ordered probit model with sample selection

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. These options are seldom used.

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

The following option is available with heckoprobit but is not shown in the dialog box: coeflegend; see [R] estimation options.

Remarks and examples

heckoprobit estimates the parameters of a regression model for an ordered categorical outcome from a nonrandom sample known as a selected sample. Selected samples suffer from “selection on unobservables” because the errors that determine whether a case is missing are correlated with the errors that determine the outcome.

For ordered categorical regression from samples that do not suffer from selection on unobservables, see [R] oprobit or [R] ologit. For regression of a continuous outcome variable from a selected sample, see [R] heckman.

Even though we are interested in modeling a single ordinal outcome, there are two dependent variables in the ordered probit sample-selection model because we must also model the sample-selection process. First, there is the ordinal outcome \( y_j \). Second, there is a binary variable that indicates whether each case in the sample is observed or unobserved. To handle the sample-selection problem, we model both dependent variables jointly. Both variables are categorical. Their categorical values are determined by the values of linear combinations of covariates and normally distributed error terms relative to certain cutpoints that partition the real line. The error terms used in the determination of selection and the ordinal outcome value may be correlated.

The probability that the ordinal outcome \( y_j \) is equal to the value \( v_h \) is given by the probability that \( x_j \beta + u_{1j} \) falls within the cutpoints \( \kappa_{h-1} \) and \( \kappa_h \),

\[
\Pr(y_j = v_h) = \Pr(\kappa_{h-1} < x_j \beta + u_{1j} \leq \kappa_h)
\]

where \( x_j \) is the outcome covariates, \( \beta \) is the coefficients, and \( u_{1j} \) is a random-error term. The observed outcome values \( v_1, \ldots, v_H \) are integers such that \( v_i < v_m \) for \( i < m \). \( \kappa_0 \) is taken as \( -\infty \) and \( \kappa_H \) is taken as \( +\infty \).

We model the selection process for the outcome by

\[
s_j = 1(z_j \gamma + u_{2j} > 0)
\]

where \( s_j = 1 \) if we observed \( y_j \) and 0 otherwise, \( z_j \) is the covariates used to model the selection process, \( \gamma \) is the coefficients for the selection process, \( 1(\cdot) \) denotes the indicator function, and \( u_{2j} \) is a random-error term.

\( (u_{1j}, u_{2j}) \) have bivariate normal distribution with mean zero and variance matrix

\[
\begin{bmatrix}
1 & \rho \\
\rho & 1
\end{bmatrix}
\]
When $\rho \neq 0$, standard ordered probit techniques applied to the outcome equation yield inconsistent results. `heckoprobit` provides consistent, asymptotically efficient estimates for all the parameters in such models.

De Luca and Perotti (2011) describe the maximum likelihood estimator used in `heckoprobit`.

### Example 1

We have a simulated dataset containing a sample of 5,000 women, 3,480 of whom work. The outcome of interest is a woman’s job satisfaction, and we suspect that unobservables that determine job satisfaction and the unobservables that increase the likelihood of employment are correlated. Women may make a decision to work based on how satisfying their job would be. We estimate the parameters of an ordered probit sample-selection model for the outcome of job satisfaction (`satisfaction`) with selection on employment (`work`). Age (`age`) and years of education (`education`) are used as outcome covariates, and we also expect that they affect selection. Additional covariates for selection are marital status (`married`) and the number of children at home (`children`).

Here we estimate the parameters of the model with `heckoprobit`. We use the factorial interaction of `married` and `children` in `select()`. This specifies that the number of children and marital status affect selection, and it allows the effect of the number of children to differ among married and nonmarried women. The factorial interaction is specified using factor-variable notation, which is described in [U] 11.4.3 Factor variables.
. use http://www.stata-press.com/data/r13/womensat
(Job satisfaction, female)
. heckoprobit satisfaction education age,
> select(work=education age i.married##c.children)

Fitting oprobit model:
Iteration 0: log likelihood = -3934.1474
Iteration 1: log likelihood = -3571.886
Iteration 2: log likelihood = -3570.2616
Iteration 3: log likelihood = -3570.2616

Fitting selection model:
Iteration 0: log likelihood = -3071.0775
Iteration 1: log likelihood = -2565.5092
Iteration 2: log likelihood = -2556.8369
Iteration 3: log likelihood = -2556.8237
Iteration 4: log likelihood = -2556.8237

Comparison: log likelihood = -6127.0853

Fitting full model:
Iteration 0: log likelihood = -6127.0853
Iteration 1: log likelihood = -6093.8868
Iteration 2: log likelihood = -6083.215
Iteration 3: log likelihood = -6083.0376
Iteration 4: log likelihood = -6083.0372

Ordered probit model with sample selection
Number of obs = 5000
Censored obs = 1520
Uncensored obs = 3480
Wald chi2(2) = 842.42
Log likelihood = -6083.037
Prob > chi2 = 0.0000

| Coef. | Std. Err. | z   | P>|z| | [95% Conf. Interval] |
|-------|-----------|-----|-----|----------------------|
| satisfaction | | | | |
| education | .1536381 | .0068266 | 22.51 | 0.000 | .1402583 | .1670179 |
| age | .0334463 | .0024049 | 13.91 | 0.000 | .0287329 | .0381598 |

| work | | | | |
| education | .0512494 | .0068095 | 7.53 | 0.000 | .037903 | .0645958 |
| age | .0288084 | .0026528 | 10.86 | 0.000 | .023609 | .0340078 |
| 1.married | .6120876 | .0700055 | 8.74 | 0.000 | .4748794 | .7492958 |
| children | .5140995 | .0288529 | 17.82 | 0.000 | .4575489 | .5706501 |
| married##c.children | | | | |
| 1 | -.1337573 | .035126 | -3.81 | 0.000 | -.202603 | -.0649117 |
| _cons | -2.203036 | .125772 | -17.52 | 0.000 | -2.449545 | -1.956528 |

| /cut1 | 1.728757 | .1232063 | 14.03 | 0.000 | 1.487277 | 1.970237 |
| /cut2 | 2.64357 | .116586 | 22.67 | 0.000 | 2.415066 | 2.872075 |
| /cut3 | 3.642911 | .1178174 | 30.92 | 0.000 | 3.411993 | 3.873829 |
| /athrho | .7430919 | .0780998 | 9.51 | 0.000 | .5900191 | .8961646 |
| rho | .6310096 | .0470026 | .529093 | .7144252 |

LR test of indep. eqns. (rho = 0): chi2(1) = 88.10 Prob > chi2 = 0.0000

The output shows several iteration logs. The first iteration log corresponds to running the ordered probit model for those observations in the sample where we have observed the outcome. The second iteration log corresponds to running the selection probit model, which models whether we observe...
our outcome of interest. If $\rho = 0$, the sum of the log likelihoods from these two models will equal
the log likelihood of the ordered probit sample-selection model; this sum is printed in the iteration
log as the comparison log likelihood. The final iteration log is for fitting the full ordered probit
sample-selection model.

The Wald test in the header is highly significant, indicating a good model fit. All the covariates
are statistically significant. The likelihood-ratio test in the footer indicates that we can reject the null
hypothesis that the errors for outcome and selection are uncorrelated. This means that we should use
the ordered probit sample-selection model instead of the simple ordered probit model.

The positive estimate of 0.63 for $\rho$ indicates that unobservables that increase job satisfaction tend
to occur with unobservables that increase the chance of having a job.

### Stored results

`heckoprobit` stores the following in `e()`:

**Scalars**

- `e(N)` number of observations
- `e(N_cens)` number of censored observations
- `e(N_cd)` number of completely determined observations
- `e(k_cat)` number of categories
- `e(k)` number of parameters
- `e(k_eq)` number of equations in `e(b)`
- `e(k_eq_model)` number of equations in overall model test
- `e(k_aux)` number of auxiliary parameters
- `e(k_dv)` number of dependent variables
- `e(df_m)` model degrees of freedom
- `e(ll)` log likelihood
- `e(ll_c)` log likelihood, comparison model
- `e(N_clust)` number of clusters
- `e(chi2)` $\chi^2$
- `e(chi2_c)` $\chi^2$ for comparison test
- `e(p_c)` $p$-value for comparison test
- `e(p)` significance of comparison test
- `e(rho)` $\rho$
- `e(rank)` rank of `e(V)`
- `e(ic)` number of iterations
- `e(rc)` return code
- `e(converged)` 1 if converged, 0 otherwise
De Luca and Perotti (2011) provide an introduction to this model.

The ordinal outcome equation is

\[ y_j = \sum_{h=1}^{H} v_h 1 \left( \kappa_{h-1} < x_j \beta + u_{1j} \leq \kappa_h \right) \]

where \( x_j \) is the outcome covariates, \( \beta \) is the coefficients, and \( u_{1j} \) is a random-error term. The observed outcome values \( v_1, \ldots, v_H \) are integers such that \( v_i < v_m \) for \( i < m \). \( \kappa_1, \ldots, \kappa_{H-1} \) are real numbers such that \( \kappa_i < \kappa_m \) for \( i < m \). \( \kappa_0 \) is taken as \(-\infty\) and \( \kappa_H \) is taken as \(+\infty\).

The selection equation is

\[ s_j = 1 \left( z_j \gamma + u_{2j} > 0 \right) \]

where \( s_j = 1 \) if we observed \( y_j \) and 0 otherwise, \( z_j \) is the covariates used to model the selection process, \( \gamma \) is the coefficients for the selection process, and \( u_{2j} \) is a random-error term.

\( (u_{1j}, u_{2j}) \) have bivariate normal distribution with mean zero and variance matrix

\[
\begin{bmatrix}
1 & \rho \\
\rho & 1
\end{bmatrix}
\]
Let $a_j = z_j \gamma + \text{offset}_j^\gamma$ and $b_j = x_j \beta + \text{offset}_j^\beta$. This yields the log likelihood

$$
\ln L = \sum_{j \notin S} w_j \ln \{\Phi(-a_j)\} + \sum_{h=1}^H \sum_{j \in S} w_j \ln \{\Phi_2(a_j, \kappa_h - b_j, -\rho) - \Phi_2(a_j, \kappa_h - 1 - b_j, -\rho)\}
$$

where $S$ is the set of observations for which $y_j$ is observed, $\Phi_2(\cdot)$ is the cumulative bivariate normal distribution function (with mean $[0 0]'$), $\Phi(\cdot)$ is the standard cumulative normal, and $w_j$ is an optional weight for observation $j$.

In the maximum likelihood estimation, $\rho$ is not directly estimated. Directly estimated is $\text{atanh} \rho$:

$$
\text{atanh} \rho = \frac{1}{2} \ln \left( \frac{1 + \rho}{1 - \rho} \right)
$$

From the form of the likelihood, it is clear that if $\rho = 0$, the log likelihood for the ordered probit sample-selection model is equal to the sum of the ordered probit model for the outcome $y$ and the selection model. We can perform a likelihood-ratio test by comparing the log likelihood of the full model with the sum of the log likelihoods for the ordered probit and selection models.

References

Baum, C. F. 2006. *An Introduction to Modern Econometrics Using Stata*. College Station, TX: Stata Press.


Also see

[R] heckoprobit postestimation — Postestimation tools for heckoprobit

[R] heckman — Heckman selection model

[R] heckprobit — Probit model with sample selection

[R] oprobit — Ordered probit regression

[R] probit — Probit regression

[R] regress — Linear regression

[R] tobit — Tobit regression

[SVY] svy estimation — Estimation commands for survey data

[U] 20 Estimation and postestimation commands