heckpoisson — Poisson regression with sample selection

Description

`heckpoisson` fits a Poisson regression model with endogenous sample selection. This is sometimes called nonignorability of selection, missing not at random, or selection bias. Unlike the standard Poisson model, there is no assumption of equidispersion.

Quick start

Poisson model of `y` on `x1` with `z1` predicting selection when binary variable `selected` indicates selection status

```
heckpoisson y x1, select(selected = z1)
```

Add categorical variable `a` using factor-variables syntax

```
heckpoisson y x1 i.a, select(selected = z1 i.a)
```

Report results as incidence-rate ratios

```
heckpoisson y x1 i.a, select(selected = z1 i.a) irr
```

Add robust standard errors

```
heckpoisson y x1 i.a, select(selected = z1 i.a) vce(robust)
```

Include exposure variable `expose` to account for different exposure levels

```
heckpoisson y x1 i.a, select(selected = z1 i.a) exposure(expose)
```

Menu

Statistics > Sample-selection models > Poisson model with sample selection

Syntax

`heckpoisson` *depvar* *indepvars* [*if*] [*in*] [*weight*] ,
 `select(` [*depvar*$_s$ `=`] *indepvars*$_s$ [`, noconstant offset(`*varname*$_{os}$`)`] `)` [*options*]

options	Description
Model	
* `select()`	specify selection equation: dependent and independent variables; whether to have constant term and offset variable
`noconstant`	suppress constant term
`exposure(`*varname*$_e$`)`	include ln(*varname*$_e$) in model with coefficient constrained to 1
`offset(`*varname*$_o$`)`	include *varname*$_o$ 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)`
`irr`	report incidence-rate ratios
`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; default is `intpoints(25)`
Maximization	
maximize_options	control the maximization process; seldom used
`coeflegend`	display legend instead of statistics

*`select( )` is required.
 The full specification is `select(` [*depvar*$_s$ `=`] *indepvars*$_s$ [`, noconstant offset(`*varname*$_{os}$`)`] `)`.

indepvars and *indepvars*$_s$ may contain factor variables; see [U] **11.4.3 Factor variables**.

indepvars and *indepvars*$_s$ 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()` and weights are not allowed with the `svy` prefix; see [SVY] **svy**.

`fweight`s, `iweight`s, and `pweight`s 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.

Options

Model

select([*depvar*$_s$ =] *indepvars*$_s$ [, noconstant offset(*varname*$_{os}$)]) specifies the variables and options for the selection equation. It is an integral part of specifying a sample-selection model and is required.

If *depvar*$_s$ is specified, it should be coded as 0 or 1, with 0 indicating an observation not selected and 1 indicating a selected observation. If *depvar*$_s$ is not specified, then 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*$_{os}$) specifies that selection offset *varname*$_{os}$ be included in the model with the coefficient constrained to be 1.

noconstant, exposure(*varname*$_e$), offset(*varname*$_o$), constraints(*constraints*), collinear; 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(#); see [R] **estimation options**.

irr reports estimated coefficients transformed to incidence-rate ratios, that is, e^{β_i} rather than β_i. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results.

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(#) specifies the number of integration points to use for quadrature. The default is intpoints(25), which means that 25 quadrature points are used. The maximum number of allowed integration points is 128.

The more integration points, the more accurate the approximation to the log likelihood. However, computation time increases with the number of quadrature points and is roughly proportional to the number of points used.

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.

The following option is available with `heckpoisson` but is not shown in the dialog box:

`coeflegend`; see [R] **estimation options**.

Remarks and examples

When analyzing observational data, we must consider the possibility that we cannot treat the observations for which we have data as if they were selected at random. Suppose we are interested in the number of after-school tutoring sessions a child attends. If unobservable variables that affect which students attend the sessions, for example, family stability, also affect the number of visits we observe, then a condition known as endogenous sample selection is present. This phenomenon is sometimes simply referred to as sample selection or called missing not at random, nonignorability of selection, or selection bias. When endogenous sample selection occurs, conventional estimation techniques are not appropriate. Cameron and Trivedi (2010, 556–562) and Greene (2018, 950–957) provide good introductions to the concept of endogenous sample selection.

The venerable Heckman estimator handles endogenous sample selection when the outcome of interest is modeled by linear regression; see [R] **heckman**. However, the Heckman estimator is not appropriate for count outcomes because its linear model for the outcome could produce negative predicted values and does not restrict the predicted values to integers.

There are different methods for estimating the parameters of a count-data model with endogenous sample selection. `heckpoisson` implements the maximum likelihood estimator derived in Terza (1998); see also Cameron and Trivedi (2013, chap. 10) for a discussion of this estimator.

The model consists of one equation for the count outcome, y, and one equation for a binary selection indicator, s. The indicator s is always observed and takes values of 0 or 1. But the outcome y is observed only if $s = 1$, that is, we have complete information about the covariates of interest and selection status. However, the value of the primary outcome of interest, y, is sometimes unknown.

More formally, the count outcome y is assumed to have a Poisson distribution, conditional on the covariates, with conditional mean

$$E(y_j|\mathbf{x}_j, \epsilon_{1j}) = \exp(\mathbf{x}_j\boldsymbol{\beta} + \epsilon_{1j}) \qquad \textit{Poisson regression equation}$$

However, we only observe y for observation j if $s_j = 1$:

$$s_j = \begin{cases} 1, & \text{if } \mathbf{w}_j\gamma + \epsilon_{2j} > 0 \\ 0, & \text{otherwise} \end{cases} \qquad \textit{selection equation}$$

where

$$\epsilon_1 \sim N(0, \sigma)$$

$$\epsilon_2 \sim N(0, 1)$$

$$\text{corr}(\epsilon_1, \epsilon_2) = \rho$$

When $\rho \neq 0$, standard Poisson regression based on the observed y yields biased estimates. `heckpoisson` provides consistent, asymptotically efficient estimates for the parameters in such models.

Unlike the standard Poisson regression, the Poisson model with sample selection allows underdispersion and overdispersion.

▷ Example 1: Poisson model with sample selection

Suppose we want to know the effect of research and development (R&D) expenditures on the number of patents obtained by a firm in the last two years. The `patent` dataset contains fictional data on the number of patents (`npatents`) of 10,000 firms in different sectors. After reading in the data, we tabulate the frequencies of `npatents` against an indicator for whether a firm applied for patents (`applied`).

```
. use http://www.stata-press.com/data/r15/patent
(Fictional data on patents and R&D)

. tabulate npatents applied, missing
```

Number of patents (last 2 yrs)	Applied for patent Not Apply	Apply	Total
0	0	1,127	1,127
1	0	1,455	1,455
2	0	1,131	1,131
3	0	710	710
4	0	479	479
5	0	266	266
6	0	126	126
7	0	98	98
8	0	66	66
9	0	42	42
10	0	19	19
11	0	24	24
12	0	5	5
13	0	7	7
14	0	5	5
15	0	10	10
17	0	1	1
18	0	1	1
19	0	2	2
22	0	1	1
.	4,425	0	4,425
Total	4,425	5,575	10,000

The output shows that `npatents` is missing for about half of the sample because some firms did not apply for any patents. Some firms prefer to keep their discoveries as trade secrets instead of applying for patents. The sample selection will be endogenous if the unobservable variables that affect which firms apply for patents also affect the number of patents obtained. Therefore, we do not want to use a standard Poisson model for these data.

We model `npatents` as a function of R&D expenditures (`expenditure`) and a categorical variable indicating whether the firm is in the information technology (IT) sector (`tech`). We model the selection indicator `applied` as a function of `expenditure`, `tech`, and firm size (`size`), which is excluded from the outcome model.

```
. heckpoisson npatents expenditure i.tech,
> select(applied = expenditure size i.tech)
initial:       log likelihood = -17442.266
rescale:       log likelihood = -17442.266
rescale eq:    log likelihood = -17442.266
(setting technique to bhhh)
Iteration 0:   log likelihood = -17442.266
Iteration 1:   log likelihood = -17441.444
Iteration 2:   log likelihood =  -17440.72
Iteration 3:   log likelihood = -17440.438
Iteration 4:   log likelihood = -17440.438
Poisson regression with endogenous selection    Number of obs     =     10,000
(25 quadrature points)                                Selected    =      5,575
                                                      Nonselected =      4,425

                                                Wald chi2(2)      =     443.90
Log likelihood = -17440.44                      Prob > chi2       =     0.0000

------------------------------------------------------------------------------
    npatents |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
npatents     |
 expenditure |    .497821   .0507866     9.80   0.000      .398281     .597361
             |
        tech |
  IT sector  |   .5833501   .0300366    19.42   0.000     .5244795    .6422207
       _cons |  -1.855143    .208204    -8.91   0.000    -2.263216   -1.447071
-------------+----------------------------------------------------------------
applied      |
 expenditure |   .1369954   .0447339     3.06   0.002     .0493185    .2246723
        size |   .2774201   .0469132     5.91   0.000     .1854718    .3693683
             |
        tech |
  IT sector  |   .2750208   .0277032     9.93   0.000     .2207236     .329318
       _cons |  -1.660778   .2631227    -6.31   0.000    -2.176489   -1.145066
-------------+----------------------------------------------------------------
     /athrho |   1.161677   .2847896     4.08   0.000     .6034999    1.719855
   /lnsigma |  -.3029685   .0499674    -6.06   0.000    -.4009028   -.2050342
-------------+----------------------------------------------------------------
         rho |   .8215857   .0925557                      .5395353    .9378455
       sigma |   .7386224    .036907                      .6697151    .8146195
------------------------------------------------------------------------------
Wald test of indep. eqns. (rho = 0): chi2(1) =    16.64   Prob > chi2 = 0.0000
```

The coefficient estimates reported by `heckpoisson` can be interpreted similarly to those reported by `poisson`. For example, the positive coefficient on `expenditure` tells us that increasing R&D expenditures is associated with an increasing number of patents. However, the magnitude of the effect cannot be directly determined by the coefficients. The best way to obtain interpretable effects is by using `margins`. See example 1 in [R] **heckpoisson postestimation** for more information.

The estimated correlation between the selection errors and outcome errors is 0.8, and the Wald test in the footer indicates that we can reject the null hypothesis of zero correlation. This positive and significant correlation estimate implies that unobservable factors that increase the number of patents a firm is awarded tend to occur with unobservable factors that also increase the chance of a firm being willing to apply for patents.

◁

❏ Technical note

In practice, we rely on the strength of the relationship between `size` and `applied` and the fact that `size` does not appear in the model for `npatents` to pin down the parameter estimates. Technically, we do not need this exclusion restriction, but identification from the functional form alone tends to be weak. For a discussion of this point, see Cameron and Trivedi (2010, 558–562).

❏

▷ Example 2: Obtaining incidence-rate ratios

In some cases, we may wish to view the parameters as incidence-rate ratios (IRRs). That is, we want to hold all the x's in the model constant except one, say, the ith. The IRR for a one-unit change in x_i is

$$\frac{e^{\ln(E)+\beta_1 x_1+\cdots+\beta_i(x_i+1)+\cdots+\beta_k x_k+e_1}}{e^{\ln(E)+\beta_1 x_1+\cdots+\beta_i x_i+\cdots+\beta_k x_k+e_1}} = e^{\beta_i}$$

For instance, we may want to know the relative incidence rate of patents as the expenditure changes or the relative incidence rate of patents as sectors change from non-IT to IT.

We can use option `irr` to display the coefficient estimates transformed to IRRs. This option may be specified when we originally fit our model or on replay. Because we have already fit the model, we specify `irr` below using the replay syntax.

```
. heckpoisson, irr

Poisson regression with endogenous selection      Number of obs     =     10,000
(25 quadrature points)                                  Selected      =      5,575
                                                        Nonselected =      4,425

                                                  Wald chi2(2)      =     443.90
Log likelihood = -17440.44                        Prob > chi2       =     0.0000
```

npatents	IRR	Std. Err.	z	P>|z|	[95% Conf.	Interval]
npatents						
expenditure	1.645133	.0835508	9.80	0.000	1.489262	1.817316
tech						
IT sector	1.792032	.0538265	19.42	0.000	1.689579	1.900697
_cons	.1564305	.0325695	-8.91	0.000	.1040154	.2352583
applied						
expenditure	.1369954	.0447339	3.06	0.002	.0493185	.2246723
size	.2774201	.0469132	5.91	0.000	.1854718	.3693683
tech						
IT sector	.2750208	.0277032	9.93	0.000	.2207236	.329318
_cons	-1.660778	.2631227	-6.31	0.000	-2.176489	-1.145066
/athrho	1.161677	.2847896	4.08	0.000	.6034999	1.719855
/lnsigma	-.3029685	.0499674	-6.06	0.000	-.4009028	-.2050342
rho	.8215857	.0925557			.5395353	.9378455
sigma	.7386224	.036907			.6697151	.8146195

```
Note: Estimates are transformed only in the first equation.
Note: _cons estimates baseline incidence rate.
Wald test of indep. eqns. (rho = 0): chi2(1) =    16.64   Prob > chi2 = 0.0000
```

The IRR for IT is about 1.8, meaning that the expected number of patents in the IT sector is 1.8 times more than in the non-IT sector.

◁

Stored results

heckpoisson 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_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(N_clust)	number of clusters
e(chi2)	χ^2
e(chi2_c)	χ^2 for comparison, ρ=0 test
e(n_quad)	number of quadrature points
e(p)	significance of model test
e(p_c)	significance of comparison test
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)	heckpoisson
e(cmdline)	command as typed
e(depvar)	name of dependent variable
e(wtype)	weight type
e(wexp)	weight expression
e(title)	title in estimation output
e(title2)	secondary title in estimation output
e(clustvar)	name of cluster variable
e(offset1)	offset for regression equation
e(offset2)	offset for selection equation
e(chi2type)	Wald; type of model χ^2 test
e(chi2_ct)	Wald; type of comparison 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(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(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

Methods and formulas

`heckpoisson` implements Terza's maximum likelihood estimator for the parameters of a count-data model with endogenous sample selection (Terza 1998).

Suppose that the count outcome y_j has covariates $\mathbf{x}_j$ and that y_j has a Poisson distribution, conditional on $\mathbf{x}_j$, with conditional mean

$$E(y_j|\mathbf{x}_j, \epsilon_{1j}) = \mu_j = \exp(\mathbf{x}_j\boldsymbol{\beta} + \epsilon_{1j})$$

and

$$\Pr(Y = y_j|\mathbf{x}_j, \epsilon_{1j}) = \frac{\mu_j^{y_j} e^{-\mu_j}}{y_j!}$$

We only observe y_j when s_j, the selection outcome, which is the binary outcome from a latent-variable model with covariates $\mathbf{w}_j$, is equal to 1.

$$s_j = \begin{cases} 1, & \text{if } \mathbf{w}_j\gamma + \epsilon_{2j} > 0 \\ 0, & \text{otherwise} \end{cases}$$

The error terms ϵ_1 and ϵ_2 are assumed to have bivariate normal distribution with zero mean and covariance matrix

$$\begin{bmatrix} \sigma^2 & \sigma\rho \\ \sigma\rho & 1 \end{bmatrix}$$

where σ and ρ have their usual interpretation for the bivariate normal distribution. A nonzero ρ implies that the selected sample is not representative of the whole population and therefore that inference based on standard Poisson regression using the observed sample is incorrect.

In maximum likelihood estimation, $\ln \sigma$ and $\operatorname{atanh} \rho$ are estimated rather than directly estimating σ and ρ.

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

The joint log likelihood is given by

$$\ln L(\boldsymbol{\theta}) = \sum_{i=1}^{N} [s_j \times \ln\{\Pr(y_j, s_j = 1)|\mathbf{x}_j, \mathbf{w}_j, \boldsymbol{\theta}\} + (1 - s_j) \times \ln\{\Pr(s_j = 0|\mathbf{w}_j, \boldsymbol{\theta})\}]$$

where $\boldsymbol{\theta}$ denotes $(\boldsymbol{\beta}, \gamma, \rho, \sigma)$ for notational simplicity.

The joint probability $\Pr(y_j, s_j = 1|\mathbf{x}_j, \mathbf{w}_j, \boldsymbol{\theta})$ can be obtained by integrating the conditional probability $\Pr(y_j, s_j = 1|\mathbf{x}_j, \mathbf{w}_j, \boldsymbol{\theta}, \epsilon_1)$ over ϵ_1. More precisely,

$$\Pr(y_j, s_j = 1|\mathbf{x}_j, \mathbf{w}_j, \boldsymbol{\theta}) = \int_{-\infty}^{\infty} \Pr(y_j|\mathbf{x}_j, \epsilon_1)\Phi\left(\frac{\mathbf{w}_j\gamma + \rho/\sigma\epsilon_1}{\sqrt{1-\rho^2}}\right)\phi(\epsilon_1/\sigma)d\epsilon_1 \qquad (1)$$

where $\phi(\cdot)$ is the standard normal density function and $\Phi(\cdot)$ is the standard normal cumulative density function. $\Pr(s_j = 0|\mathbf{w}_j, \boldsymbol{\theta})$ is similarly derived.

$$\Pr(s_j = 0|\mathbf{w}_j, \boldsymbol{\theta}) = \int_{-\infty}^{\infty} \Phi\left(-\frac{\mathbf{w}_j\gamma + \rho/\sigma\epsilon_1}{\sqrt{1-\rho^2}}\right)\phi(\epsilon_1/\sigma)d\epsilon_1 \qquad (2)$$

The integrations in (1) and (2) have no closed form and must be approximated using Gauss–Hermite quadrature.

`heckpoisson` supports the Huber/White/sandwich estimator of the variance and its clustered version using `vce(robust)` and `vce(cluster` *clustvar*`)`, respectively. See [P] **_robust**, particularly *Maximum likelihood estimators* and *Methods and formulas*.

`heckpoisson` also supports estimation with survey data. For details on VCEs with survey data, see [SVY] **variance estimation**.

References

Cameron, A. C., and P. K. Trivedi. 2010. *Microeconometrics Using Stata*. Rev. ed. College Station, TX: Stata Press.

——. 2013. *Regression Analysis of Count Data*. 2nd ed. New York: Cambridge University Press.

Greene, W. H. 2018. *Econometric Analysis*. 8th ed. New York: Pearson.

Terza, J. V. 1998. Estimating count data models with endogenous switching: Sample selection and endogenous treatment effects. *Journal of Econometrics* 84: 129–154.

Also see

[R] **heckpoisson postestimation** — Postestimation tools for heckpoisson

[R] **heckman** — Heckman selection model

[R] **heckoprobit** — Ordered probit model with sample selection

[R] **heckprobit** — Probit model with sample selection

[R] **poisson** — Poisson regression

[SVY] **svy estimation** — Estimation commands for survey data

[TE] **etpoisson** — Poisson regression with endogenous treatment effects

[U] 20 Estimation and postestimation commands