

[Description](#)
[Options](#)
[References](#)[Quick start](#)
[Remarks and examples](#)
[Also see](#)[Menu](#)
[Stored results](#)[Syntax](#)
[Methods and formulas](#)

Description

`xheckman` fits a random-effects linear regression model with endogenous sample selection.

Quick start

Random-effects model of `y` on `x1` using `xtset` data, with selection indicated by binary variable `selected` and predicted by `v1` and `x1`

```
xheckman y x1, select(selected = v1 x1)
```

Same as above, but constraining random effects to be independent

```
xheckman y x1, select(selected = v1 x1) norecorrelation
```

Same as above, but omit random effects from selection model

```
xheckman y x1, select(selected = v1 x1, none)
```

Menu

Statistics > Longitudinal/panel data > Sample-selection models > Linear regression with sample selection (RE)

Syntax

```
xheckman depvar [indepvars] [if] [in],  
      select(depvars = varlists [, sel_options]) [options]
```

<i>options</i>	Description
<hr/>	
Model	
* <u>select</u> ()	specify selection equation: dependent and independent variables; whether to have constant term and offset variable or include random effect
<u>noconstant</u>	suppress constant term
<u>norecorrelation</u>	constrain the random effects to be independent
<u>offset</u> (<i>varname</i> _o)	include <i>varname</i> _o in model with coefficient constrained to 1
<u>constraints</u> (<i>numlist</i>)	apply specified linear constraints
<hr/>	
SE/Robust	
<u>vce</u> (<i>vcetype</i>)	<i>vcetype</i> may be <u>oim</u> , <u>robust</u> , <u>cluster</u> <i>clustvar</i> , <u>opg</u> , <u>bootstrap</u> , or <u>jackknife</u>
<hr/>	
Reporting	
<u>level</u> (#)	set confidence level; default is <u>level</u> (95)
<u>nocnsreport</u>	do not display constraints
<u>display_options</u>	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
<hr/>	
Integration	
<u>intmethod</u> (<i>intmethod</i>)	integration method for random effects; <i>intmethod</i> may be <u>mvaghermite</u> (the default) or <u>ghermite</u>
<u>intpoints</u> (#)	set the number of integration (quadrature) points for random-effects integration; default is <u>intpoints</u> (7)
<hr/>	
Maximization	
<u>maximize_options</u>	control the maximization process; seldom used
<u>collinear</u>	keep collinear variables
<u>coeflegend</u>	display legend instead of statistics

<i>sel_options</i>	Description
<hr/>	
Model	
<u>noconstant</u>	suppress constant term
<u>nore</u>	do not include random effects in selection model
<u>offset</u> (<i>varname</i> _o)	include <i>varname</i> _o in model with coefficient constrained to 1

*select() is required.

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, collect, jackknife, and statsby are allowed; see [U] 11.1.10 Prefix commands.

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

`select(depvars = varlists [, sel_options])` specifies a random-effects 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). `select()` is required.

sel_options are the following:

`noconstant` suppresses the constant term (intercept) in the selection model.

`nore` specifies that a random effect not be included in the selection equation.

`offset(varnameo)` specifies that *varname*_{*o*} be included in the selection model with the coefficient constrained to 1.

`noconstant`; see [R] [Estimation options](#).

`norecorrelation` constrains the random effects in the outcome and selection equations to be independent.

`offset(varnameo)`, `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 [XT] [vce_options](#).

Reporting

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

display_options: `nocl`, `nopvalues`, `noomitted`, `vsquish`, `noemptycells`, `baselevels`, `allbaselevels`, `nofvlabel`, `fvwrap(#)`, `fvwrapon(style)`, `cformat(%fmt)`, `pformat(%fmt)`, `sformat(%fmt)`, and `nolstretch`; see [R] [Estimation options](#).

Integration

`intmethod(intmethod)` and `intpoints(#)` control how the integration of random effects is numerically calculated.

`intmethod()` specifies the integration method. The default method is mean-variance adaptive Gauss–Hermite quadrature, `intmethod(mvaghermite)`. We recommend this method. `intmethod(ghermite)` specifies that nonadaptive Gauss–Hermite quadrature be used. This method is less computationally intensive and less accurate. It is sometimes useful to try `intmethod(ghermite)` to get the model to converge and then perhaps use the results as initial values specified in `from` when fitting the model using the more accurate `intmethod(mvaghermite)`. See [Methods and formulas](#) for more details.

`intpoints()` 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 [Methods and formulas](#) for more details.

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 default technique 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 `xheckman` but are not shown in the dialog box:

`collinear`, `coeflegend`; see [R] [Estimation options](#).

Remarks and examples

`xheckman` fits a panel-data model with endogenous sample selection. Endogenous sample selection is sometimes called nonignorability of selection, missing not at random, or selection bias. Within-panel correlation is accounted for by using panel-level random effects.

The outcome of interest y_{it} is modeled as

$$y_{it} = \mathbf{x}_{it}\beta + \nu_{1i} + \epsilon_{1it}$$

where \mathbf{x}_{it} are the covariates modeling the outcome, ν_{1i} is the panel-level random effect, and ϵ_{1it} is the observation-level error.

We model the selection process for the outcome by

$$s_{it} = 1 (\mathbf{z}_{it}\alpha + \nu_{2i} + \epsilon_{2it} > 0)$$

where $s_{it} = 1$ if we observe y_{it} and 0 otherwise, \mathbf{z}_{it} are the covariates modeling selection, ν_{2i} is the panel-level random effect for selection, and ϵ_{2it} is the observation-level selection error.

The random effects ν_{1i} and ν_{2i} are bivariate normal with mean 0 and variance

$$\begin{bmatrix} \sigma_{1\nu}^2 & \rho_{\nu}\sigma_{1\nu}\sigma_{2\nu} \\ \rho_{\nu}\sigma_{1\nu}\sigma_{2\nu} & \sigma_{2\nu}^2 \end{bmatrix}$$

The observation-level errors ϵ_{1it} and ϵ_{2it} are bivariate normal, with mean 0 and variance

$$\begin{bmatrix} \sigma_1^2 & \rho\sigma_1 \\ \rho\sigma_1 & 1 \end{bmatrix}$$

These observation-level errors are independent of the random effects.

Using the Heckman estimator ([heckman](#)) for this model will provide inefficient estimates because it ignores the within-panel correlation. Instead, we use maximum likelihood to model both the selection and outcome equations and account for the panel structure of the data. This random-effects estimator is used by `xtereg` and was discussed in [Rabe-Hesketh, Skrondal, and Pickles \(2002\)](#). There is no parametric fixed-effects estimator for panel data from an endogenously selected sample. See [Honoré, Kyriazidou, and Powell \(2000\)](#) and [Kyriazidou \(1997\)](#) for semiparametric fixed-effects estimators of panel-data endogenous sample-selection models.

► Example 1

Suppose that we wish to study the relationship between wage, job tenure, and age for college-educated adults. We have fictional data on 600 adults observed from 2013 to 2016. We use these data to model hourly wage as a function of age, age squared, and job tenure. However, an individual's wage is observed only if he or she works, and not everyone was employed on the dates the data were collected. We are not interested in modeling only the subpopulation of individuals who were employed at the time. We are also interested in the relationship of job tenure and age with the wage an individual would have received if he or she had been employed.

We suspect that the unobserved factors that affect an individual's wage are related to the unobserved factors that affect employment status. These unobserved factors could include person-level characteristics like ability and time-varying factors like an individual's family situation. We suspect that we have an endogenously selected sample. We have data on the local job market conditions (`market`). This variable is used with age and tenure to model the employment status of an individual.

Before we can fit a random-effects model to our data, 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 2013 to 2016.

```
. use https://www.stata-press.com/data/r19/wagework
(Wages for 20 to 77 year olds, 2013–2016)
. xtset personid year
Panel variable: personid (strongly balanced)
Time variable: year, 2013 to 2016
Delta: 1 unit
```

We are now ready to fit our model.

```
. xheckman wage c.age##c.age tenure, select(working = c.age##c.age market)
(setting technique to bhhh)
Iteration 0: Log likelihood = -5384.5076
Iteration 1: Log likelihood = -5377.4625
Iteration 2: Log likelihood = -5376.4805
Iteration 3: Log likelihood = -5376.4505
Iteration 4: Log likelihood = -5376.4464
Iteration 5: Log likelihood = -5376.4454
Iteration 6: Log likelihood = -5376.4451
Iteration 7: Log likelihood = -5376.4451
Iteration 8: Log likelihood = -5376.445
Iteration 9: Log likelihood = -5376.445
(switching technique to nr)
Iteration 10: Log likelihood = -5376.445

Random-effects regression with selection

Number of obs      =    2,400
       Selected     =    1,928
       Nonselected   =     472

Group variable: personid

Number of groups    =     600
Obs per group:
    min =         4
    avg =        4.0
    max =         4

Integration method: mvaghermite

Integration pts.    =         7

Wald chi2(3)        =   2827.78
Prob > chi2         =    0.0000

Log likelihood = -5376.445
```

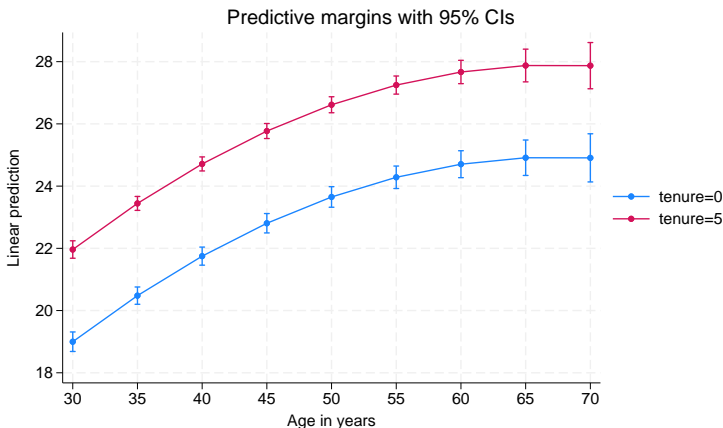
	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
wage						
age	.5722234	.0477613	11.98	0.000	.4786129	.6658339
c.age#c.age	-.0042448	.0005329	-7.97	0.000	-.0052893	-.0032003
tenure	.5927719	.0169866	34.90	0.000	.5594787	.626065
_cons	5.651812	1.038011	5.44	0.000	3.617347	7.686277
working						
age	.2305309	.0207988	11.08	0.000	.1897661	.2712958
c.age#c.age	-.0026832	.0002241	-11.97	0.000	-.0031225	-.0022439
market	.1894934	.019038	9.95	0.000	.1521796	.2268072
_cons	-3.276904	.4352836	-7.53	0.000	-4.130045	-2.423764
var(e.wage)	4.458219	.2235342			4.040939	4.918588
corr(e.wor~g, e.wage)	.4091115	.1391856	2.94	0.003	.1065022	.642359
var(wage[pers~d])	2.493737	.2547628			2.041226	3.046564
var(wor~g[per~d])	.3831411	.0830963			.250466	.5860961
corr(wor~g[per~d], wage[pers~d])	.6021096	.0845675	7.12	0.000	.4106863	.7426953

The first two sections of the output provide the estimated coefficients for the wage equation and the selection (`working`) equation. We can interpret the coefficients in the wage equation using the standard linear regression interpretation. For example, we expect an increase of \$0.59 per hour for an additional year of job tenure.

Next we see `var(e.wage)`, an estimate of the variance of the observation-level error for wage; this is followed by `corr(e.working,e.wage)`, an estimate of its correlation with the observation-level error for the selection model. The next section of the output reports estimates of the variances of the random effects, `var(wage[personid])` and `var(working[personid])`. The last section reports an estimate of the correlation of these random effects. If at least one of the correlations is significantly different from zero, we can conclude that we have endogenous sample selection. In our case, the correlation between the observation-level errors is 0.41, and the correlation between the random effects is 0.60. Because both are positive and significantly different from zero, we conclude that we have endogenous selection and that unobserved individual-level factors that increase the chance of being employed tend to increase wage. Additionally, unobserved observation-level (time-varying) factors that increase the chance of being employed tend to increase wage.

We estimated coefficients for age and age squared. We can use `margins` and `marginsplot` to gain a clearer understanding of the effect of the individuals' age on hourly wage. We use `margins` with `at()` to profile the expected wages for individuals between ages 30 and 70 and with 0 and 5 years of job tenure. Then, we use `marginsplot` to graph the estimates.

```
. margins, at(age=(30(5)70) tenure =(0 5))
. marginsplot
```



Based on this model, and assuming the data are from a random or otherwise representative sample, the plotted points represent the expected wage for individuals with the specified job tenure and age. We see that age has an increasing effect on expected wage until the mid-60s and then attenuates. Having 5 years of job tenure instead of none shifts the curve up by about \$3.00 per hour.

Stored results

xthheckman stores the following in `e()`:

Scalars

<code>e(N)</code>	number of observations
<code>e(N_g)</code>	number of groups
<code>e(N_selected)</code>	number of selected observations
<code>e(N_nonselected)</code>	number of nonselected observations
<code>e(k)</code>	number of parameters
<code>e(k_eq)</code>	number of equations in <code>e(b)</code>
<code>e(k_eq_model)</code>	number of equations in overall model test
<code>e(k_aux)</code>	number of auxiliary parameters
<code>e(k_dv)</code>	number of dependent variables
<code>e(df_m)</code>	model degrees of freedom
<code>e(ll)</code>	log likelihood
<code>e(N_clust)</code>	number of clusters
<code>e(chi2)</code>	χ^2
<code>e(p)</code>	p -value for model test
<code>e(n_requad)</code>	number of integration points for random effects
<code>e(g_min)</code>	smallest group size
<code>e(g_avg)</code>	average group size
<code>e(g_max)</code>	largest group size
<code>e(rank)</code>	rank of <code>e(V)</code>
<code>e(ic)</code>	number of iterations
<code>e(rc)</code>	return code
<code>e(converged)</code>	1 if converged, 0 otherwise

Macros

<code>e(cmd)</code>	xthheckman
<code>e(cmdline)</code>	command as typed
<code>e(depvar)</code>	names of dependent variables
<code>e(ivar)</code>	variable denoting groups
<code>e(title)</code>	title in estimation output
<code>e(clustvar)</code>	name of cluster variable
<code>e(offset#)</code>	offset for the $\#$ th <i>depvar</i> , where $\#$ is determined by equation order in output
<code>e(chi2type)</code>	Wald; type of model χ^2 test
<code>e(vce)</code>	<i>vcetype</i> specified in <code>vce()</code>
<code>e(vcetype)</code>	title used to label Std. err.
<code>e(reintmethod)</code>	integration method for random effects
<code>e(opt)</code>	type of optimization
<code>e(which)</code>	max or min; whether optimizer is to perform maximization or minimization
<code>e(ml_method)</code>	type of ml method
<code>e(user)</code>	name of likelihood-evaluator program
<code>e(technique)</code>	maximization technique
<code>e(properties)</code>	b V
<code>e(estat_cmd)</code>	program used to implement <code>estat</code>
<code>e(predict)</code>	program used to implement <code>predict</code>
<code>e(marginsok)</code>	predictions allowed by <code>margins</code>
<code>e(marginsnotok)</code>	predictions disallowed by <code>margins</code>
<code>e(asbalanced)</code>	factor variables <code>fvset</code> as <code>asbalanced</code>
<code>e(asobserved)</code>	factor variables <code>fvset</code> as <code>asobserved</code>

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

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

`xheckman` fits a random-effects linear regression model with endogenous sample selection via maximum likelihood estimation. For details on maximum likelihood estimators, see the results in [Wooldridge \(2010, chap. 13\)](#) and [White \(1996\)](#).

The log-likelihood function maximized by `xheckman` is implied by the triangular structure of the model. Specifically, the joint distribution of the outcome and selection 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 [Amemiya \(1985, chap. 10\)](#); [Heckman \(1976, 1979\)](#); [Maddala \(1983, chap. 5\)](#); [Maddala and Lee \(1976\)](#); [Wooldridge \(2010, sec. 15.7.2, 15.7.3, 16.3.3, 17.5.2, and 19.7.1; 2014\)](#). [Roodman \(2011\)](#) and [Bartus and Roodman \(2014\)](#) used this result to derive the formulas discussed below.

We have panels $i = 1, \dots, N$ and observations $t = 1, \dots, N_i$. We model y_{it} as

$$y_{it} = \mathbf{x}_{it}\beta + \nu_{1i} + \epsilon_{1it}$$

where \mathbf{x}_{it} are the outcome covariates, ν_{1i} is the panel-level random effect, and ϵ_{1it} is the observation-level error.

The selection process for the outcome is modeled by

$$s_{it} = 1 (\mathbf{z}_{it}\alpha + \nu_{2i} + \epsilon_{2it} > 0)$$

where $s_{it} = 1$ if we observe y_{it} and 0 otherwise, \mathbf{z}_{it} are the selection covariates, ν_{2i} is the panel-level random effect for selection, and ϵ_{2it} is the observation-level selection error.

The random effects ν_{1i} and ν_{2i} are bivariate normal with mean 0 and variance

$$\Sigma_\nu = \begin{bmatrix} \sigma_{1\nu}^2 & \rho_\nu \sigma_{1\nu} \sigma_{2\nu} \\ \rho_\nu \sigma_{1\nu} \sigma_{2\nu} & \sigma_{2\nu}^2 \end{bmatrix}$$

The observation-level errors ϵ_{1it} and ϵ_{2it} are independent of the random effects and are also bivariate normal, with mean 0 and variance

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho \sigma_1 \\ \rho \sigma_1 & 1 \end{bmatrix}$$

When we condition on the random effects ν_{1i} and ν_{2i} , we can write the joint density of the y_{it} and s_{it} using the conditional density of the selection error ϵ_{2it} on the outcome error ϵ_{1it} .

For the selection indicator s_{it} , we have lower limit l_{it} and upper limit u_{it} ,

$$l_{it} = \begin{cases} -\infty & s_{it} = 0 \\ -\mathbf{z}_{it}\boldsymbol{\alpha} - \nu_{2i} - \frac{\rho}{\sigma_1}(y_{it} - \mathbf{x}_{it}\boldsymbol{\beta} - \nu_{1i}) & s_{it} = 1 \end{cases} \quad u_{it} = \begin{cases} -\mathbf{z}_{it}\boldsymbol{\alpha} - \nu_{2i} & s_{it} = 0 \\ \infty & s_{it} = 1 \end{cases}$$

Then, the joint density of y_{it} and s_{it} conditional on the random effects is

$$f(y_{it}, s_{it} | \nu_{1i}, \nu_{2i}) = \begin{cases} \Phi\left(\frac{-l_{it}}{\sqrt{1-\rho^2}}\right) + \phi\left(\frac{y_{it}-\mathbf{x}_{it}\boldsymbol{\beta}}{\sigma_1}\right) & s_{it} = 1 \\ \Phi\left(\frac{u_{it}}{\sqrt{1-\rho^2}}\right) & s_{it} = 0 \end{cases}$$

Note that each panel has the same random effects for each observation. So the likelihood for panel i is

$$L_i = \int_{\mathbb{R}^2} \left[\prod_{t=1}^{N_i} f(y_{it}, s_{it} | \nu_{1i}, \nu_{2i}) \phi_2\{(\nu_{1i}, \nu_{2i}), \boldsymbol{\Sigma}_\nu\} d\nu_{1i} d\nu_{2i} \right] \quad (1)$$

This multivariate integral is generally not tractable. We can use a change-of-variables technique to transform it into a set of nested univariate integrals. Let \mathbf{L} be the Cholesky decomposition of $\boldsymbol{\Sigma}_\nu$; that is, $\boldsymbol{\Sigma}_\nu = \mathbf{L}\mathbf{L}'$. It follows that $(\nu_{1i}, \nu_{2i})' = \mathbf{L}\boldsymbol{\psi}_i$, where $\boldsymbol{\psi}_i$ is a vector of independent standard normal random variables.

We can rewrite (1) as

$$L_i = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left\{ \prod_{t=1}^{N_i} f(y_{it}, s_{it} | (\nu_{1i}, \nu_{2i})' = \mathbf{L}\boldsymbol{\psi}_i) \right\} \phi(\psi_{1i}) \phi(\psi_{2i}) d\psi_{1i} d\psi_{2i}$$

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^*) , where $k = 1, \dots, q$. The GHQ approximation is then

$$\int_{-\infty}^{\infty} f(x) \exp(-x^2) dx \approx \sum_{k=1}^q w_k^* f(a_k^*)$$

Consider a 2-dimensional quadrature grid containing q quadrature points in both dimensions. Let the vector of abscissas $\mathbf{a}_k = (a_{k_1}, a_{k_2})'$ be a point in this grid, and let $\mathbf{w}_k = (w_{k_1}, w_{k_2})'$ be the vector of corresponding weights.

The GHQ approximation to the likelihood for a given panel is

$$L_i = \sum_{k_1=1}^q \sum_{k_2=1}^q \left[\left\{ \prod_{t=1}^{N_i} f(y_{it}, s_{it} | (\nu_{1i}, \nu_{2i})' = \mathbf{L}\mathbf{a}_k) \right\} \left\{ \prod_{s=1}^2 w_{k_s} \right\} \right]$$

Rather than using regular GHQ, we can use mean–variance adaptive GHQ. Fixing the observed variables and model parameters in the integrand of (1), we see the posterior density for $\boldsymbol{\psi}_i$ is proportional to

$$\left\{ \prod_{t=1}^{N_i} f(y_{it}, s_{it} | (\nu_{1i}, \nu_{2i})' = \mathbf{L}\boldsymbol{\psi}_i) \right\} \phi(\boldsymbol{\psi}_i)$$

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 $\boldsymbol{\psi}_i$ as the weighting distribution in the integral, we can use our approximation for the posterior density,

$$L_i = \int_{\mathbb{R}^2} \frac{\left\{ \prod_{t=1}^{N_i} f(y_{it}, s_{it} | (\nu_{1i}, \nu_{2i})' = \mathbf{L}\boldsymbol{\psi}_i) \right\} \phi(\boldsymbol{\psi}_i)}{\phi(\boldsymbol{\psi}_i, \boldsymbol{\mu}_{vi}, \boldsymbol{\tau}_{vi})} \phi(\boldsymbol{\psi}_i, \boldsymbol{\mu}_{vi}, \boldsymbol{\tau}_{vi}) d\boldsymbol{\psi}_i$$

The likelihood is then approximated by

$$L_i = \sum_{k_1=1}^q \cdots \sum_{k_2=1}^q \left[\left\{ \prod_{t=1}^{N_i} f(y_{it}, s_{it} | (\nu_{1i}, \nu_{2i})' = \mathbf{L}\boldsymbol{\alpha}_k) \right\} \left\{ \prod_{s=1}^2 \omega_{k_s} \right\} \right]$$

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 $\boldsymbol{\psi}_i$. 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 GHQ approximation, starting with a 0 mean vector and identity variance matrix.

The log likelihood for all panels is then

$$\ln L = \sum_{i=1}^N \left(\ln \sum_{k_1=1}^q \cdots \sum_{k_2=1}^q \left[\left\{ \prod_{t=1}^{N_i} f(y_{it}, s_{it} | (\nu_{1i}, \nu_{2i})' = \mathbf{L}\boldsymbol{\alpha}_k) \right\} \left\{ \prod_{s=1}^2 \omega_{k_s} \right\} \right] \right)$$

The conditional mean of y_{it} is

$$E(y_{it} | \mathbf{x}_{it}) = \mathbf{x}_{it}\boldsymbol{\beta}$$

xtheckman results are obtained using `xteregress`; see [Methods and formulas](#) of `[ERM] eregress`.

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.
- 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”. In *Annals of Economic and Social Measurement*, edited by S. V. Berg, vol. 5: 475–492. Cambridge, MA: National Bureau of Economic Research.
- . 1979. Sample selection bias as a specification error. *Econometrica* 47: 153–161. <https://doi.org/10.2307/1912352>.
- Honoré, B. E., E. Kyriazidou, and J. L. Powell. 2000. Estimation of tobit-type models with individual specific effects. *Econometric Reviews* 19: 341–366. <https://doi.org/10.1080/07474930008800476>.
- Kyriazidou, E. 1997. Estimation of a panel data sample selection model. *Econometrica* 65: 1335–1364. <https://doi.org/10.2307/2171739>.
- Maddala, G. S. 1983. *Limited-Dependent and Qualitative Variables in Econometrics*. Cambridge: Cambridge University Press. <https://doi.org/10.1017/CBO9780511810176>.
- Maddala, G. S., and L.-F. Lee. 1976. “Recursive models with qualitative endogenous variables”. In *Annals of Economic and Social Measurement*, edited by S. V. Berg, vol. 5: 525–545. Cambridge, MA: National Bureau of Economic Research.
- Rabe-Hesketh, S., A. Skrondal, and A. Pickles. 2002. Multilevel selection models using gllamm. Presented at the 2002 Dutch and German Stata Users Group meeting, Maastricht, The Netherlands, May 23. <https://www.stata.com/meeting/2dutch/select.pdf>.
- 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. <https://doi.org/10.1017/CCOL0521252806>.
- 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

- [XT] **xheckman postestimation** — Postestimation tools for xheckman
- [XT] **xteregress** — Extended random-effects linear regression
- [XT] **xtreg** — Linear models for panel data
- [XT] **xtset** — Declare data to be panel data
- [ERM] **eregress** — Extended linear regression
- [R] **heckman** — Heckman selection model
- [R] **regress** — Linear regression
- [U] **20 Estimation and postestimation commands**

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.

For suggested citations, see the FAQ on [citing Stata documentation](#).

