Title

stata.com

xtcloglog — Random-effects and population-averaged cloglog models

Description

`xtcloglog` fits population-averaged and random-effects complementary log–log (cloglog) models for a binary dependent variable. Complementary log–log models are typically used when one of the outcomes is rare relative to the other.

Quick start

Random-effects complementary log–log regression of `y` on `x1` and `x2` using `xtset` data

```
xtcloglog y x1 x2
```

Add indicators for levels of categorical variable `a` and interact `x1` with `x2`

```
xtcloglog y x1 x2 c.x1#c.x2 i.a
```

Same as above, but suppress iteration log

```
xtcloglog y x1 x2 c.x1#c.x2 i.a, nolog
```

Population-averaged model with an exchangeable correlation structure

```
xtcloglog y x1 x2 c.x1#c.x2 i.a, pa
```

Random-effects model with bootstrap standard errors

```
xtcloglog y x1 x2 c.x1#c.x2 i.a, vce(bootstrap)
```

Population-averaged model with an autoregressive correlation structure of order 1

```
xtcloglog y x1 x2 c.x1#c.x2 i.a, pa corr(ar 1)
```

Menu

Statistics > Longitudinal/panel data > Binary outcomes > Complementary log–log regression (RE, PA)

Syntax

Random-effects (RE) model

```
xtcloglog depvar [indepvars] [if] [in] [weight] [, re RE_options]
```

Population-averaged (PA) model

```
xtcloglog depvar [indepvars] [if] [in] [weight], pa [PA_options]
```

RE_options	Description
Model	
noconstant	suppress constant term
re	use random-effects estimator; the default
offset(*varname*)	include *varname* in model with coefficient constrained to 1
constraints(*constraints*)	apply specified linear constraints
asis	retain perfect predictor variables
SE/Robust	
vce(*vcetype*)	*vcetype* may be oim, robust, cluster *clustvar*, bootstrap, or jackknife
Reporting	
level(#)	set confidence level; default is level(95)
lrmodel	perform the likelihood-ratio model test instead of the default Wald test
eform	report exponentiated coefficients
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	
intmethod(*intmethod*)	integration method; *intmethod* may be mvaghermite (the default) or ghermite
intpoints(#)	use # quadrature points; default is intpoints(12)
Maximization	
maximize_options	control the maximization process; seldom used
collinear	keep collinear variables
coeflegend	display legend instead of statistics

PA_options	Description
Model	
noconstant	suppress constant term
pa	use population-averaged estimator
offset(*varname*)	include *varname* in model with coefficient constrained to 1
asis	retain perfect predictor variables
Correlation	
corr(*correlation*)	within-panel correlation structure
force	estimate even if observations unequally spaced in time
SE/Robust	
vce(*vcetype*)	*vcetype* may be conventional, robust, bootstrap, or jackknife
nmp	use divisor $N - P$ instead of the default N
scale(*parm*)	overrides the default scale parameter; *parm* may be x2, dev, phi, or #
Reporting	
level(#)	set confidence level; default is level(95)
eform	report exponentiated coefficients
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Optimization	
optimize_options	control the optimization process; seldom used
coeflegend	display legend instead of statistics

correlation	Description
exchangeable	exchangeable; the default
independent	independent
unstructured	unstructured
fixed *matname*	user-specified
ar #	autoregressive of order #
stationary #	stationary of order #
nonstationary #	nonstationary of order #

A panel variable must be specified. For xtcloglog, pa, correlation structures other than exchangeable and independent require that a time variable also be specified. Use xtset; see [XT] **xtset**.

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

by, collect, mi estimate, and statsby are allowed; see [U] **11.1.10 Prefix commands**. fp is allowed for the random-effects model.

vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] **mi estimate**.

iweights, fweights, and pweights are allowed for the population-averaged model, and iweights are allowed for the random-effects model; see [U] **11.1.6 weight**. Weights must be constant within panel.

collinear and coeflegend do not appear in the dialog box.

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

Options for RE model

Model

`noconstant`; see [R] **Estimation options**.

`re` requests the random-effects estimator, which is the default.

`offset(`*varname*`)`, `constraints(`*constraints*`)`; see [R] **Estimation options**.

`asis` forces retention of perfect predictor variables and their associated, perfectly predicted observations and may produce instabilities in maximization; see [R] **probit**.

SE/Robust

`vce(`*vcetype*`)` specifies the type of standard error reported, which includes types that are derived from asymptotic theory (`oim`), 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***.

Specifying `vce(robust)` is equivalent to specifying `vce(cluster` *panelvar*`)`; see *xtcloglog, re and the robust VCE estimator* in *Methods and formulas*.

Reporting

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

`eform` displays the exponentiated coefficients and corresponding standard errors and confidence intervals.

`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

`intmethod(`*intmethod*`)`, `intpoints(#)`; see [R] **Estimation 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**. These options are seldom used.

The following options are available with `xtcloglog` but are not shown in the dialog box:

`collinear`, `coeflegend`; see [R] **Estimation options**.

Options for PA model

Model

`noconstant`; see [R] **Estimation options**.

`pa` requests the population-averaged estimator.

`offset(`*varname*`)`; see [R] **Estimation options**

asis forces retention of perfect predictor variables and their associated, perfectly predicted observations and may produce instabilities in maximization; see [R] **probit**.

Correlation

corr(*correlation*) specifies the within-panel correlation structure; the default corresponds to the equal-correlation model, corr(exchangeable).

When you specify a correlation structure that requires a lag, you indicate the lag after the structure's name with or without a blank; for example, corr(ar 1) or corr(ar1).

If you specify the fixed correlation structure, you specify the name of the matrix containing the assumed correlations following the word fixed, for example, corr(fixed myr).

force specifies that estimation be forced even though the time variable is not equally spaced. This is relevant only for correlation structures that require knowledge of the time variable. These correlation structures require that observations be equally spaced so that calculations based on lags correspond to a constant time change. If you specify a time variable indicating that observations are not equally spaced, the (time dependent) model will not be fit. If you also specify force, the model will be fit, and it will be assumed that the lags based on the data ordered by the time variable are appropriate.

SE/Robust

vce(*vcetype*) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (conventional), that are robust to some kinds of misspecification (robust), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [XT] ***vce_options***.

vce(conventional), the default, uses the conventionally derived variance estimator for generalized least-squares regression.

nmp, scale(x2 | dev | phi | #); see [XT] ***vce_options***.

Reporting

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

eform displays the exponentiated coefficients and corresponding standard errors and confidence intervals.

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**.

Optimization

optimize_options control the iterative optimization process. These options are seldom used.

iterate(#) specifies the maximum number of iterations. When the number of iterations equals #, the optimization stops and presents the current results, even if convergence has not been reached. The default is iterate(100).

tolerance(#) specifies the tolerance for the coefficient vector. When the relative change in the coefficient vector from one iteration to the next is less than or equal to #, the optimization process is stopped. tolerance(1e-6) is the default.

log and nolog specify whether to display the iteration log. The iteration log is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] ***set iter***.

trace specifies that the current estimates be printed at each iteration.

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

coeflegend; see [R] **Estimation options**.

Remarks and examples

stata.com

xtcloglog may be used to fit a population-averaged model or a random-effects complementary log–log (cloglog) model. There is no command for a conditional fixed-effects model, as there does not exist a sufficient statistic allowing the fixed effects to be conditioned out of the likelihood. Unconditional fixed-effects cloglog models may be fit with cloglog with indicator variables for the panels. However, unconditional fixed-effects estimates are biased. We do not discuss fixed-effects further in this entry.

By default, the population-averaged model is an equal-correlation model; that is, xtcloglog, pa assumes corr(exchangeable). Thus, xtcloglog, pa is a shortcut command for fitting the population-averaged model using xtgee; see [XT] **xtgee**. Typing

```
. xtcloglog ..., pa ...
```

is equivalent to typing

```
. xtgee ..., ... family(binomial) link(cloglog) corr(exchangeable)
```

Also see [XT] **xtgee** for information about xtcloglog.

By default or when re is specified, xtcloglog fits, via maximum likelihood, the random-effects model

$$\Pr(y_{it} \neq 0 | \mathbf{x}_{it}) = P(\mathbf{x}_{it}\boldsymbol{\beta} + \nu_i)$$

for $i = 1, \ldots, n$ panels, where $t = 1, \ldots, n_i$, ν_i are i.i.d., $N(0, \sigma_\nu^2)$, and $P(z) = 1 - \exp\{-\exp(z)\}$.

Underlying this model is the variance-components model

$$y_{it} \neq 0 \iff \mathbf{x}_{it}\boldsymbol{\beta} + \nu_i + \epsilon_{it} > 0$$

where ϵ_{it} are i.i.d. extreme-value (Gumbel) distributed with the mean equal to Euler's constant and variance $\sigma_\epsilon^2 = \pi^2/6$, independently of ν_i. The nonsymmetric error distribution is an alternative to logit and probit analysis and is typically used when the positive (or negative) outcome is rare.

▷ Example 1

Suppose that we are studying unionization of women in the United States and are using the `union` dataset; see [XT] **xt**. We wish to fit a random-effects model of union membership:

```
. use https://www.stata-press.com/data/r18/union
(NLS Women 14-24 in 1968)
. xtcloglog union age grade not_smsa south##c.year
  (output omitted)
Random-effects complementary log-log model          Number of obs     = 26,200
Group variable: idcode                              Number of groups  =  4,434
Random effects u_i ~ Gaussian                       Obs per group:
                                                                  min =      1
                                                                  avg =    5.9
                                                                  max =     12
Integration method: mvaghermite                     Integration pts.  =     12
                                                    Wald chi2(6)      = 248.58
Log likelihood = -10535.928                         Prob > chi2       = 0.0000
```

union	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
age	.0128659	.0119004	1.08	0.280	-.0104586	.0361903
grade	.06985	.0138135	5.06	0.000	.042776	.096924
not_smsa	-.198416	.0647943	-3.06	0.002	-.3254104	-.0714215
1.south	-2.047645	.488965	-4.19	0.000	-3.005999	-1.089291
year	-.0006432	.0123569	-0.05	0.958	-.0248623	.0235759
south#c.year						
1	.0164259	.006065	2.71	0.007	.0045387	.0283132
_cons	-3.269158	.659029	-4.96	0.000	-4.560831	-1.977485
/lnsig2u	1.24128	.0461705			1.150787	1.331772
sigma_u	1.860118	.0429413			1.77783	1.946214
rho	.677778	.0100834			.6577057	.6972152

```
LR test of rho=0: chibar2(01) = 6009.36                  Prob >= chibar2 = 0.000
```

The output includes the additional panel-level variance component, which is parameterized as the log of the standard deviation, $\ln\sigma_\nu$ (labeled `lnsig2u` in the output). The standard deviation σ_ν is also included in the output, labeled `sigma_u`, together with ρ (labeled `rho`),

$$\rho = \frac{\sigma_\nu^2}{\sigma_\nu^2 + \sigma_\epsilon^2}$$

which is the proportion of the total variance contributed by the panel-level variance component.

When `rho` is zero, the panel-level variance component is not important, and the panel estimator is no different from the pooled estimator (`cloglog`). A likelihood-ratio test of this is included at the bottom of the output, which formally compares the pooled estimator with the panel estimator.

As an alternative to the random-effects specification, you might want to fit an equal-correlation population-averaged cloglog model by typing

```
. xtcloglog union age grade not_smsa south##c.year, pa
Iteration 1:  Tolerance = .11878399
Iteration 2:  Tolerance = .01424628
Iteration 3:  Tolerance = .00075278
Iteration 4:  Tolerance = .00003195
Iteration 5:  Tolerance = 1.661e-06
Iteration 6:  Tolerance = 8.308e-08
GEE population-averaged model                   Number of obs    = 26,200
Group variable: idcode                          Number of groups =  4,434
Family: Binomial                                Obs per group:
Link:   Complementary log-log                                min =      1
Correlation: exchangeable                                    avg =    5.9
                                                             max =     12
                                                Wald chi2(6)     = 234.66
Scale parameter = 1                             Prob > chi2      = 0.0000
```

union	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
age	.0153737	.0081156	1.89	0.058	-.0005326	.03128
grade	.0549518	.0095093	5.78	0.000	.0363139	.0735897
not_smsa	-.1045232	.0431082	-2.42	0.015	-.1890138	-.0200326
1.south	-1.714868	.3384558	-5.07	0.000	-2.378229	-1.051507
year	-.0115881	.0084125	-1.38	0.168	-.0280763	.0049001
south#c.year						
1	.0149796	.0041687	3.59	0.000	.0068091	.0231501
_cons	-1.488278	.4468005	-3.33	0.001	-2.363991	-.6125652

◁

▷ Example 2

In [R] **cloglog**, we showed these results and compared them with `cloglog, vce(cluster id)`. `xtcloglog` with the `pa` option allows a `vce(robust)` option so we can obtain the population-averaged cloglog estimator with the robust variance calculation by typing

```
. xtcloglog union age grade not_smsa south##c.year, pa vce(robust)
  (output omitted)
GEE population-averaged model                   Number of obs    = 26,200
Group variable: idcode                          Number of groups =  4,434
Family: Binomial                                Obs per group:
Link:   Complementary log-log                                min =      1
Correlation: exchangeable                                    avg =    5.9
                                                             max =     12
                                                Wald chi2(6)     = 157.24
Scale parameter = 1                             Prob > chi2      = 0.0000
                                 (Std. err. adjusted for clustering on idcode)
```

union	Coefficient	Semirobust std. err.	z	P>|z|	[95% conf.	interval]
age	.0153737	.0079446	1.94	0.053	-.0001974	.0309448
grade	.0549518	.0117258	4.69	0.000	.0319697	.077934
not_smsa	-.1045232	.0548598	-1.91	0.057	-.2120465	.0030001
1.south	-1.714868	.4864999	-3.52	0.000	-2.66839	-.7613455
year	-.0115881	.0085742	-1.35	0.177	-.0283932	.005217
south#c.year						
1	.0149796	.0060548	2.47	0.013	.0031124	.0268468
_cons	-1.488278	.4924738	-3.02	0.003	-2.453509	-.5230472

These standard errors are similar to those shown for `cloglog, vce(cluster id)` in [R] **cloglog**.

◁

❑ Technical note

The random-effects model is calculated using quadrature, which is an approximation whose accuracy depends partially on the number of integration points used. We can use the `quadchk` command to see if changing the number of integration points affects the results. If the results change, the quadrature approximation is not accurate given the number of integration points. Try increasing the number of integration points using the `intpoints()` option and run `quadchk` again. Do not attempt to interpret the results of estimates when the coefficients reported by `quadchk` differ substantially. See [XT] **quadchk** for details and [XT] **xtprobit** for an example.

Because the `xtcloglog` likelihood function is calculated by Gauss–Hermite quadrature, on large problems the computations can be slow. Computation time is roughly proportional to the number of points used for the quadrature.

❑

Stored results

`xtcloglog, re` stores the following in `e()`:

Scalars

`e(N)`	number of observations
`e(N_g)`	number of groups
`e(k)`	number of parameters
`e(k_aux)`	number of auxiliary parameters
`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(df_m)`	model degrees of freedom
`e(ll)`	log likelihood
`e(ll_0)`	log likelihood, constant-only model
`e(ll_c)`	log likelihood, comparison model
`e(chi2)`	χ^2
`e(chi2_c)`	χ^2 for comparison test
`e(N_clust)`	number of clusters
`e(rho)`	ρ
`e(sigma_u)`	panel-level standard deviation
`e(n_quad)`	number of quadrature points
`e(g_min)`	smallest group size
`e(g_avg)`	average group size
`e(g_max)`	largest group size
`e(p)`	p-value for model test
`e(rank)`	rank of `e(V)`
`e(rank0)`	rank of `e(V)` for constant-only model
`e(ic)`	number of iterations
`e(rc)`	return code
`e(converged)`	`1` if converged, `0` otherwise

Macros

`e(cmd)`	`xtcloglog`
`e(cmdline)`	command as typed
`e(depvar)`	name of dependent variable
`e(ivar)`	variable denoting groups
`e(model)`	`re`
`e(wtype)`	weight type
`e(wexp)`	weight expression
`e(title)`	title in estimation output
`e(clustvar)`	name of cluster variable
`e(offset)`	linear offset variable
`e(chi2type)`	`Wald` or `LR`; type of model χ^2 test
`e(chi2_ct)`	`Wald` or `LR`; type of model χ^2 test corresponding to `e(chi2_c)`
`e(vce)`	*vcetype* specified in `vce()`
`e(vcetype)`	title used to label Std. err.
`e(intmethod)`	integration method
`e(distrib)`	`Gaussian`; the distribution of the random effect
`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(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(Cns)`	constraints matrix
`e(ilog)`	iteration log
`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.

xtcloglog, pa stores the following in e():

Scalars

e(N)	number of observations
e(N_g)	number of groups
e(df_m)	model degrees of freedom
e(chi2)	χ^2
e(p)	p-value for model test
e(df_pear)	degrees of freedom for Pearson χ^2
e(chi2_dev)	χ^2 test of deviance
e(chi2_dis)	χ^2 test of deviance dispersion
e(deviance)	deviance
e(dispers)	deviance dispersion
e(phi)	scale parameter
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(tol)	target tolerance
e(dif)	achieved tolerance
e(rc)	return code

Macros

e(cmd)	xtgee
e(cmd2)	xtcloglog
e(cmdline)	command as typed
e(depvar)	name of dependent variable
e(ivar)	variable denoting groups
e(tvar)	variable denoting time within groups
e(model)	pa
e(family)	binomial
e(link)	cloglog; link function
e(corr)	correlation structure
e(scale)	x2, dev, phi, or #; scale parameter
e(wtype)	weight type
e(wexp)	weight expression
e(offset)	linear offset variable
e(chi2type)	Wald; type of model χ^2 test
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. err.
e(nmp)	nmp, if specified
e(properties)	b V
e(predict)	program used to implement predict
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(R)	estimated working correlation matrix
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

xtcloglog, pa reports the population-averaged results obtained using xtgee, family(binomial) link(cloglog) to obtain estimates.

For the random-effects model, assume a normal distribution, $N(0, \sigma_\nu^2)$, for the random effects ν_i,

$$\Pr(y_{i1}, \ldots, y_{in_i} | \mathbf{x}_{i1}, \ldots, \mathbf{x}_{in_i}) = \int_{-\infty}^{\infty} \frac{e^{-\nu_i^2/2\sigma_\nu^2}}{\sqrt{2\pi}\sigma_\nu} \left\{ \prod_{t=1}^{n_i} F(y_{it}, \mathbf{x}_{it}\boldsymbol{\beta} + \nu_i) \right\} d\nu_i$$

where

$$F(y, z) = \begin{cases} 1 - \exp\{ - \exp(z)\} & \text{if } y \neq 0 \\ \exp\{ - \exp(z)\} & \text{otherwise} \end{cases}$$

The panel-level likelihood l_i is given by

$$l_i = \int_{-\infty}^{\infty} \frac{e^{-\nu_i^2/2\sigma_\nu^2}}{\sqrt{2\pi}\sigma_\nu} \left\{ \prod_{t=1}^{n_i} F(y_{it}, \mathbf{x}_{it}\boldsymbol{\beta} + \nu_i) \right\} d\nu_i$$

$$\equiv \int_{-\infty}^{\infty} g(y_{it}, x_{it}, \nu_i) d\nu_i$$

This integral can be approximated with M-point Gauss–Hermite quadrature

$$\int_{-\infty}^{\infty} e^{-x^2} h(x) dx \approx \sum_{m=1}^{M} w_m^* h(a_m^*)$$

This is equivalent to

$$\int_{-\infty}^{\infty} f(x) dx \approx \sum_{m=1}^{M} w_m^* \exp\left\{(a_m^*)^2\right\} f(a_m^*)$$

where the w_m^* denote the quadrature weights and the a_m^* denote the quadrature abscissas. The log likelihood, L, is the sum of the logs of the panel-level likelihoods l_i.

The default approximation of the log likelihood is by adaptive Gauss–Hermite quadrature, which approximates the panel-level likelihood with

$$l_i \approx \sqrt{2}\widehat{\sigma}_i \sum_{m=1}^{M} w_m^* \exp\left\{(a_m^*)^2\right\} g(y_{it}, x_{it}, \sqrt{2}\widehat{\sigma}_i a_m^* + \widehat{\mu}_i)$$

where $\widehat{\sigma}_i$ and $\widehat{\mu}_i$ are the adaptive parameters for panel i. Therefore, with the definition of $g(y_{it}, x_{it}, \nu_i)$, the total log likelihood is approximated by

$$L \approx \sum_{i=1}^{n} w_i \log\left[\sqrt{2}\widehat{\sigma}_i \sum_{m=1}^{M} w_m^* \exp\left\{(a_m^*)^2\right\} \frac{\exp\left\{-(\sqrt{2}\widehat{\sigma}_i a_m^* + \widehat{\mu}_i)^2/2\sigma_\nu^2\right\}}{\sqrt{2\pi}\sigma_\nu} \prod_{t=1}^{n_i} F(y_{it}, x_{it}\boldsymbol{\beta} + \sqrt{2}\widehat{\sigma}_i a_m^* + \widehat{\mu}_i)\right]$$

where w_i is the user-specified weight for panel i; if no weights are specified, $w_i = 1$.

The default method of adaptive Gauss–Hermite quadrature is to calculate the posterior mean and variance and use those parameters for $\widehat{\mu}_i$ and $\widehat{\sigma}_i$ by following the method of Naylor and Smith (1982), further discussed in Skrondal and Rabe-Hesketh (2004). We start with $\widehat{\sigma}_{i,0} = 1$ and $\widehat{\mu}_{i,0} = 0$, and the posterior means and variances are updated in the kth iteration. That is, at the kth iteration of the optimization for l_i, we use

$$l_{i,k} \approx \sum_{m=1}^{M} \sqrt{2}\widehat{\sigma}_{i,k-1} w_m^* \exp\left\{a_m^*\right)^2\right\} g(y_{it}, x_{it}, \sqrt{2}\widehat{\sigma}_{i,k-1} a_m^* + \widehat{\mu}_{i,k-1})$$

Letting

$$\tau_{i,m,k-1} = \sqrt{2}\widehat{\sigma}_{i,k-1} a_m^* + \widehat{\mu}_{i,k-1}$$

$$\widehat{\mu}_{i,k} = \sum_{m=1}^{M} (\tau_{i,m,k-1}) \frac{\sqrt{2}\widehat{\sigma}_{i,k-1} w_m^* \exp\left\{(a_m^*)^2\right\} g(y_{it}, x_{it}, \tau_{i,m,k-1})}{l_{i,k}}$$

and

$$\widehat{\sigma}_{i,k} = \sum_{m=1}^{M} (\tau_{i,m,k-1})^2 \frac{\sqrt{2}\widehat{\sigma}_{i,k-1} w_m^* \exp\left\{(a_m^*)^2\right\} g(y_{it}, x_{it}, \tau_{i,m,k-1})}{l_{i,k}} - (\widehat{\mu}_{i,k})^2$$

and this is repeated until $\widehat{\mu}_{i,k}$ and $\widehat{\sigma}_{i,k}$ have converged for this iteration of the maximization algorithm. This adaptation is applied on every iteration until the log-likelihood change from the preceding iteration is less than a relative difference of 1e–6; after this, the quadrature parameters are fixed.

The log likelihood can also be calculated by nonadaptive Gauss–Hermite quadrature, the `intmethod(ghermite)` option, where $\rho = \sigma_\nu^2/(\sigma_\nu^2 + 1)$:

$$\begin{aligned} L &= \sum_{i=1}^{n} w_i \log\left\{ \Pr(y_{i1}, \ldots, y_{in_i} | \mathbf{x}_{i1}, \ldots, \mathbf{x}_{in_i}) \right\} \\ &\approx \sum_{i=1}^{n} w_i \log\left[\frac{1}{\sqrt{\pi}} \sum_{m=1}^{M} w_m^* \prod_{t=1}^{n_i} F\left\{ y_{it}, \mathbf{x}_{it}\boldsymbol{\beta} + a_m^* \left(\frac{2\rho}{1-\rho}\right)^{1/2} \right\} \right] \end{aligned}$$

Both quadrature formulas require that the integrated function be well approximated by a polynomial of degree equal to the number of quadrature points. The number of periods (panel size) can affect whether

$$\prod_{t=1}^{n_i} F(y_{it}, \mathbf{x}_{it}\boldsymbol{\beta} + \nu_i)$$

is well approximated by a polynomial. As panel size and ρ increase, the quadrature approximation can become less accurate. For large ρ, the random-effects model can also become unidentified. Adaptive quadrature gives better results for correlated data and large panels than nonadaptive quadrature; however, we recommend that you use the `quadchk` command (see [XT] **quadchk**) to verify the quadrature approximation used in this command, whichever approximation you choose.

xtcloglog, re and the robust VCE estimator

Specifying `vce(robust)` or `vce(cluster` *clustvar*`)` causes the Huber/White/sandwich VCE estimator to be calculated for the coefficients estimated in this regression. See [P] **_robust**, particularly *Introduction* and *Methods and formulas*. Wooldridge (2020) and Arellano (2003) discuss this application of the Huber/White/sandwich VCE estimator. As discussed by Wooldridge (2020), Stock and Watson (2008), and Arellano (2003), specifying `vce(robust)` is equivalent to specifying `vce(cluster` *panelvar*`)`, where *panelvar* is the variable that identifies the panels.

Clustering on the panel variable produces a consistent VCE estimator when the disturbances are not identically distributed over the panels or there is serial correlation in ϵ_{it}.

The cluster–robust VCE estimator requires that there are many clusters and the disturbances are uncorrelated across the clusters. The panel variable must be nested within the cluster variable because of the within-panel correlation that is generally induced by the random-effects transform when there is heteroskedasticity or within-panel serial correlation in the idiosyncratic errors.

References

Arellano, M. 2003. *Panel Data Econometrics*. Oxford: Oxford University Press.

Liang, K.-Y., and S. L. Zeger. 1986. Longitudinal data analysis using generalized linear models. *Biometrika* 73: 13–22. https://doi.org/10.1093/biomet/73.1.13.

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.

Neuhaus, J. M. 1992. Statistical methods for longitudinal and clustered designs with binary responses. *Statistical Methods in Medical Research* 1: 249–273. https://doi.org/10.1177/096228029200100303.

Neuhaus, J. M., J. D. Kalbfleisch, and W. W. Hauck. 1991. A comparison of cluster-specific and population-averaged approaches for analyzing correlated binary data. *International Statistical Review* 59: 25–35. https://doi.org/10.2307/1403572.

Pendergast, J. F., S. J. Gange, M. A. Newton, M. J. Lindstrom, M. Palta, and M. R. Fisher. 1996. A survey of methods for analyzing clustered binary response data. *International Statistical Review* 64: 89–118. https://doi.org/10.2307/1403425.

Skrondal, A., and S. Rabe-Hesketh. 2004. *Generalized Latent Variable Modeling: Multilevel, Longitudinal, and Structural Equation Models*. Boca Raton, FL: Chapman and Hall/CRC.

Stock, J. H., and M. W. Watson. 2008. Heteroskedasticity-robust standard errors for fixed effects panel data regression. *Econometrica* 76: 155–174. https://doi.org/10.1111/j.0012-9682.2008.00821.x.

Wooldridge, J. M. 2020. *Introductory Econometrics: A Modern Approach*. 7th ed. Boston: Cengage.

Also see

[XT] **xtcloglog postestimation** — Postestimation tools for xtcloglog

[XT] **quadchk** — Check sensitivity of quadrature approximation

[XT] **xtgee** — GEE population-averaged panel-data models

[XT] **xtlogit** — Fixed-effects, random-effects, and population-averaged logit models

[XT] **xtprobit** — Random-effects and population-averaged probit models

[XT] **xtset** — Declare data to be panel data

[ME] **mecloglog** — Multilevel mixed-effects complementary log–log regression

[MI] **Estimation** — Estimation commands for use with mi estimate

[R] **cloglog** — Complementary log–log regression

[U] 20 Estimation and postestimation commands