xtgee — GEE population-averaged panel-data models

Description

xtgee fits population-averaged panel-data models. In particular, xtgee fits generalized linear models and allows you to specify the within-group correlation structure for the panels.

Quick start

Population-averaged linear regression of y on x1 and x2

```
xtgee y x1 x2
```

Same as above, but estimate time-varying intragroup correlations

```
xtgee y x1 x2, corr(unstructured)
```

Same as above, but estimate a common second-order autoregression structure for the within-panel correlation

```
xtgee y x1 x2, corr(ar 2)
```

Population-averaged negative binomial regression of y2 on x3 and x4 equivalent to xtnbreg, pa

```
xtgee y2 x3 x4, family(nbinomial 1)
```

Population-averaged logistic regression of y3 on x5 and x6 when y3 is the number of events out of 10 trials

```
xtgee y3 x5 x6, family(binomial 10)
```

Menu

Statistics > Longitudinal/panel data > Generalized estimating equations (GEE) > Generalized estimating equations (GEE)

Syntax

xtgee *depvar* [*indepvars*] [*if*] [*in*] [*weight*] [, *options*]

options	Description
Model	
family(*family*)	distribution of *depvar*
link(*link*)	link function
Model 2	
exposure(*varname*)	include ln(*varname*) in model with coefficient constrained to 1
offset(*varname*)	include *varname* in model with coefficient constrained to 1
noconstant	suppress constant term
asis	retain perfect predictor variables
force	estimate even if observations unequally spaced in time
Correlation	
corr(*correlation*)	within-group correlation structure
SE/Robust	
vce(*vcetype*)	*vcetype* may be conventional, robust, bootstrap, or jackknife
nmp	use divisor $N - P$ instead of the default N
rgf	multiply the robust variance estimate by $(N-1)/(N-P)$
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
nodisplay	suppress display of header and coefficients
coeflegend	display legend instead of statistics

A panel variable must be specified. 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**.

depvar and *indepvars* may contain time-series operators; see [U] **11.4 varname and varlists**.

by, collect, mfp, mi estimate, and statsby are allowed; see [U] **11.1.10 Prefix commands**.

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

iweights, fweights, and pweights are allowed; see [U] **11.1.6 weight**. Weights must be constant within panel.

nodisplay and coeflegend do not appear in the dialog box.

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

family	Description
`gaussian`	Gaussian (normal); `family(normal)` is a synonym
`igaussian`	inverse Gaussian
`binomial`[#|*varname*]	Bernoulli/binomial
`poisson`	Poisson
`nbinomial`[#]	negative binomial
`gamma`	gamma

link	Link function/definition
`identity`	identity; $y = y$
`log`	log; $\ln(y)$
`logit`	logit; $\ln\{y/(1-y)\}$, natural log of the odds
`probit`	probit; $\Phi^{-1}(y)$, where $\Phi(\,)$ is the normal cumulative distribution
`cloglog`	cloglog; $\ln\{-\ln(1-y)\}$
`power`[#]	power; y^k with $k = \#$; $\# = 1$ if not specified
`opower`[#]	odds power; $[\{y/(1-y)\}^k - 1]/k$ with $k = \#$; $\# = 1$ if not specified
`nbinomial`	negative binomial; $\ln\{y/(y+\alpha)\}$
`reciprocal`	reciprocal; $1/y$

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

Options

Model

`family(`*family*`)` specifies the distribution of *depvar*; `family(gaussian)` is the default.

`link(`*link*`)` specifies the link function; the default is the canonical link for the `family()` specified (except for `family(nbinomial)`).

Model 2

`exposure(`*varname*`)` and `offset(`*varname*`)` are different ways of specifying the same thing. `exposure()` specifies a variable that reflects the amount of exposure over which the *depvar* events were observed for each observation; ln(*varname*) with coefficient constrained to be 1 is entered into the regression equation. `offset()` specifies a variable that is to be entered directly into the log-link function with its coefficient constrained to be 1; thus, exposure is assumed to be $e^{varname}$. If you were fitting a Poisson regression model, `family(poisson) link(log)`, for instance, you would account for exposure time by specifying `offset()` containing the log of exposure time.

noconstant specifies that the linear predictor has no intercept term, thus forcing it through the origin on the scale defined by the link function.

asis forces retention of perfect predictor variables and their associated, perfectly predicted observations and may produce instabilities in maximization; see [R] **probit**. This option is only allowed with option family(binomial) with a denominator of 1.

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.

Correlation

corr(*correlation*) specifies the within-group 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).

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.

vce(robust) specifies that the Huber/White/sandwich estimator of variance is to be used in place of the default conventional variance estimator (see *Methods and formulas* below). Use of this option causes xtgee to produce valid standard errors even if the correlations within group are not as hypothesized by the specified correlation structure. Under a noncanonical link, it does, however, require that the model correctly specifies the mean. The resulting standard errors are thus labeled "semirobust" instead of "robust" in this case. Although there is no vce(cluster *clustvar*) option, results are as if this option were included and you specified clustering on the panel variable.

nmp; see [XT] ***vce_options***.

rgf specifies that the robust variance estimate is multiplied by $(N-1)/(N-P)$, where N is the total number of observations and P is the number of coefficients estimated. This option can be used only with family(gaussian) when vce(robust) is either specified or implied by the use of pweights. Using this option implies that the robust variance estimate is not invariant to the scale of any weights used.

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 as described in [R] **Maximize**. For family(binomial) link(logit) (that is, logistic regression), exponentiation results in odds ratios; for family(poisson) link(log) (that is, Poisson regression), exponentiated coefficients are incidence-rate ratios.

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 options are available with xtgee but are not shown in the dialog box:

nodisplay is for programmers. It suppresses display of the header and coefficients.

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

Remarks and examples

For a thorough introduction to GEE in the estimation of GLM, see Hardin and Hilbe (2013). More information on linear models is presented in Nelder and Wedderburn (1972). Finally, there have been several illuminating articles on various applications of GEE in Zeger, Liang, and Albert (1988); Zeger and Liang (1986), and Liang (1987). Pendergast et al. (1996) surveys the current methods for analyzing clustered data in regard to binary response data. Our implementation follows that of Liang and Zeger (1986).

xtgee fits generalized linear models of y_{it} with covariates $\mathbf{x}_{it}$

$$g\{E(y_{it})\} = \mathbf{x}_{it}\boldsymbol{\beta}, \qquad y \sim F \text{ with parameters } \theta_{it}$$

for $i = 1, \dots, m$ and $t = 1, \dots, n_i$, where there are n_i observations for each group identifier i. $g(\,)$ is called the link function, and F is the distributional family. Substituting various definitions for $g(\,)$ and F results in a wide array of models. For instance, if y_{it} is distributed Gaussian (normal) and $g(\,)$ is the identity function, we have

$$E(y_{it}) = \mathbf{x}_{it}\boldsymbol{\beta}, \qquad y \sim N(\,)$$

yielding linear regression, random-effects regression, or other regression-related models, depending on what we assume for the correlation structure.

If $g(\cdot)$ is the logit function and y_{it} is distributed Bernoulli (binomial), we have

$$\text{logit}\{E(y_{it})\} = \mathbf{x}_{it}\boldsymbol{\beta}, \qquad y \sim \text{Bernoulli}$$

or logistic regression. If $g(\cdot)$ is the natural log function and y_{it} is distributed Poisson, we have

$$\ln\{E(y_{it})\} = \mathbf{x}_{it}\boldsymbol{\beta}, \qquad y \sim \text{Poisson}$$

or Poisson regression, also known as the log-linear model. Other combinations are possible.

You specify the link function with the `link()` option, the distributional family with `family()`, and the assumed within-group correlation structure with `corr()`.

The binomial distribution can be specified as case 1 `family(binomial)`, case 2 `family(binomial #)`, or case 3 `family(binomial` *varname*`)`. In case 2, # is the value of the binomial denominator N, the number of trials. Specifying `family(binomial 1)` is the same as specifying `family(binomial)`; both mean that y has the Bernoulli distribution with values 0 and 1 only. In case 3, *varname* is the variable containing the binomial denominator, thus allowing the number of trials to vary across observations.

The negative binomial distribution must be specified as `family(nbinomial #)`, where # denotes the value of the parameter α in the negative binomial distribution. The results will be conditional on this value.

You do not have to specify both `family()` and `link()`; the default `link()` is the canonical link for the specified `family()` (excluding `family(nbinomial)`):

Family	Default link
`family(binomial)`	`link(logit)`
`family(gamma)`	`link(reciprocal)`
`family(gaussian)`	`link(identity)`
`family(igaussian)`	`link(power -2)`
`family(nbinomial)`	`link(log)`
`family(poisson)`	`link(log)`

The canonical link for the negative binomial family is obtained by specifying `link(nbinomial)`. If you specify both `family()` and `link()`, not all combinations make sense. You may choose among the following combinations:

	Gaussian	Inverse	Binomial	Poisson	Negative	Gamma
		Gaussian			Binomial	
Identity	x	x	x	x	x	x
Log	x	x	x	x	x	x
Logit			x			
Probit			x			
C. log–log			x			
Power	x	x	x	x	x	x
Odds Power			x			
Neg. binom.					x	
Reciprocal	x		x	x		x

You specify the assumed within-group correlation structure with the `corr()` option.

For example, call **R** the working correlation matrix for modeling the within-group correlation, a square $\max\{n_i\} \times \max\{n_i\}$ matrix. `corr()` specifies the structure of **R**. Let $\mathbf{R}_{t,s}$ denote the t, s element.

The `independent` structure is defined as

$$\mathbf{R}_{t,s} = \begin{cases} 1 & \text{if } t = s \\ 0 & \text{otherwise} \end{cases}$$

The `corr(exchangeable)` structure (corresponding to equal-correlation models) is defined as

$$\mathbf{R}_{t,s} = \begin{cases} 1 & \text{if } t = s \\ \rho & \text{otherwise} \end{cases}$$

The `corr(ar` *g*`)` structure is defined as the usual correlation matrix for an AR(g) model. This is sometimes called multiplicative correlation. For example, an AR(1) model is given by

$$\mathbf{R}_{t,s} = \begin{cases} 1 & \text{if } t = s \\ \rho^{|t-s|} & \text{otherwise} \end{cases}$$

The `corr(stationary` *g*`)` structure is a stationary(g) model. For example, a stationary(1) model is given by

$$\mathbf{R}_{t,s} = \begin{cases} 1 & \text{if } t = s \\ \rho & \text{if } |t-s| = 1 \\ 0 & \text{otherwise} \end{cases}$$

The `corr(nonstationary` *g*`)` structure is a nonstationary(g) model that imposes only the constraints that the elements of the working correlation matrix along the diagonal be 1 and the elements outside the gth band be zero,

$$\mathbf{R}_{t,s} = \begin{cases} 1 & \text{if } t = s \\ \rho_{ts} & \text{if } 0 < |t-s| \le g,\, \rho_{ts} = \rho_{st} \\ 0 & \text{otherwise} \end{cases}$$

`corr(unstructured)` imposes only the constraint that the diagonal elements of the working correlation matrix be 1.

$$\mathbf{R}_{t,s} = \begin{cases} 1 & \text{if } t = s \\ \rho_{ts} & \text{otherwise, } \rho_{ts} = \rho_{st} \end{cases}$$

The `corr(fixed` *matname*`)` specification is taken from the user-supplied matrix, such that

$$\mathbf{R} = \textit{matname}$$

Here the correlations are not estimated from the data. The user-supplied matrix must be a valid correlation matrix with 1s on the diagonal.

Full formulas for all the correlation structures are provided in the *Methods and formulas* below.

❑ Technical note

Some `family()`, `link()`, and `corr()` combinations result in models already fit by Stata:

`family()`	`link()`	`corr()`	Other Stata estimation command
`gaussian`	`identity`	`independent`	`regress`
`gaussian`	`identity`	`exchangeable`	`xtreg, re`
`gaussian`	`identity`	`exchangeable`	`xtreg, pa`
`binomial`	`cloglog`	`independent`	`cloglog` (see note 1)
`binomial`	`cloglog`	`exchangeable`	`xtcloglog, pa`
`binomial`	`logit`	`independent`	`logit` or `logistic`
`binomial`	`logit`	`exchangeable`	`xtlogit, pa`
`binomial`	`probit`	`independent`	`probit` (see note 2)
`binomial`	`probit`	`exchangeable`	`xtprobit, pa`
`nbinomial`	`log`	`independent`	`nbreg` (see note 3)
`poisson`	`log`	`independent`	`poisson`
`poisson`	`log`	`exchangeable`	`xtpoisson, pa`
`gamma`	`log`	`independent`	`streg, dist(exp) nohr` (see note 4)
family	*link*	`independent`	`glm, irls` (see note 5)

Notes:

1. For cloglog estimation, `xtgee` with `corr(independent)` and `cloglog` (see [R] **cloglog**) will produce the same coefficients, but the standard errors will be only asymptotically equivalent because cloglog is not the canonical link for the binomial family.

2. For probit estimation, `xtgee` with `corr(independent)` and `probit` will produce the same coefficients, but the standard errors will be only asymptotically equivalent because probit is not the canonical link for the binomial family. If the binomial denominator is not 1, the equivalent maximum-likelihood command is `glm` with options `family(binomial #)` or `family(binomial` *varname*`)` and `link(probit)`; see [R] **probit** and [R] **glm**.

3. Fitting a negative binomial model by using `xtgee` (or using `glm`) will yield results conditional on the specified value of α. The `nbreg` command, however, estimates that parameter and provides unconditional estimates; see [R] **nbreg**.

4. `xtgee` with `corr(independent)` can be used to fit exponential regressions, but this requires specifying `scale(1)`. As with probit, the `xtgee`-reported standard errors will be only asymptotically equivalent to those produced by `streg, dist(exp) nohr` (see [ST] **streg**) because log is not the canonical link for the gamma family. `xtgee` cannot be used to fit exponential regressions on censored data.

   Using the `independent` correlation structure, the `xtgee` command will fit the same model fit with the `glm, irls` command if the family–link combination is the same.

5. If the `xtgee` command is equivalent to another command, using `corr(independent)` and the `vce(robust)` option with `xtgee` corresponds to using the `vce(cluster` *clustvar*`)` option in the equivalent command, where *clustvar* corresponds to the panel variable.

❑

`xtgee` is a generalization of the `glm, irls` command and gives the same output when the same family and link are specified together with an independent correlation structure. What makes `xtgee` useful is

- the number of statistical models that it generalizes for use with panel data, many of which are not otherwise available in Stata;
- the richer correlation structure `xtgee` allows, even when models are available through other xt commands; and

- the availability of robust standard errors (see [U] **20.22 Obtaining robust variance estimates**), even when the model and correlation structure are available through other xt commands.

In the following examples, we illustrate the relationships of xtgee with other Stata estimation commands. Remember that, although xtgee generalizes many other commands, the computational algorithm is different; therefore, the answers you obtain will not be identical. The dataset we are using is a subset of the nlswork data (see [XT] **xt**); we are looking at observations before 1980.

▷ Example 1

We can use xtgee to perform ordinary least squares by regress:

```
. use https://www.stata-press.com/data/r19/nlswork2
(National Longitudinal Survey of Young Women, 14-24 years old in 1968)
. regress ln_w grade age c.age#c.age
      Source |       SS           df       MS      Number of obs   =    16,085
-------------+----------------------------------   F(3, 16081)     =   1413.68
       Model |  597.54468         3  199.18156   Prob > F        =    0.0000
    Residual |  2265.74584    16,081  .14089583   R-squared       =    0.2087
-------------+----------------------------------   Adj R-squared   =    0.2085
       Total |  2863.29052    16,084  .178021047   Root MSE        =    .37536

------------------------------------------------------------------------------
     ln_wage | Coefficient  Std. err.      t    P>|t|     [95% conf. interval]
-------------+----------------------------------------------------------------
       grade |   .0724483   .0014229    50.91   0.000     .0696592    .0752374
         age |   .1064874   .0083644    12.73   0.000     .0900922    .1228825
             |
 c.age#c.age |  -.0016931   .0001655   -10.23   0.000    -.0020174   -.0013688
             |
       _cons |  -.8681487   .1024896    -8.47   0.000     -1.06904   -.6672577
------------------------------------------------------------------------------

. xtgee ln_w grade age c.age#c.age, corr(indep) nmp
Iteration 1:  Tolerance = 8.684e-13
GEE population-averaged model                  Number of obs      =     16,085
Group variable: idcode                         Number of groups   =      3,913
Family: Gaussian                               Obs per group:
Link:   Identity                                             min =          1
Correlation: independent                                     avg =        4.1
                                                             max =          9
                                               Wald chi2(3)       =    4241.04
Scale parameter = .1408958                     Prob > chi2        =     0.0000
Pearson chi2(16081)   =  2265.75               Deviance           =    2265.75
Dispersion (Pearson) = .1408958                Dispersion         =   .1408958

------------------------------------------------------------------------------
     ln_wage | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       grade |   .0724483   .0014229    50.91   0.000     .0696594    .0752372
         age |   .1064874   .0083644    12.73   0.000     .0900935    .1228812
             |
 c.age#c.age |  -.0016931   .0001655   -10.23   0.000    -.0020174   -.0013688
             |
       _cons |  -.8681487   .1024896    -8.47   0.000    -1.069025   -.6672728
------------------------------------------------------------------------------
```

When nmp is specified, the coefficients and the standard errors produced by the estimators are the same. Moreover, the scale parameter estimate from the xtgee command equals the MSE calculation from regress; both are estimates of the variance of the residuals.

◁

▷ Example 2

The identity link and Gaussian family produce regression-type models. With the independent correlation structure, we reproduce ordinary least squares. With the exchangeable correlation structure, we produce an equal-correlation linear regression estimator.

xtgee, fam(gauss) link(ident) corr(exch) is asymptotically equivalent to the weighted-GLS estimator provided by xtreg, re and to the full maximum-likelihood estimator provided by xtreg, mle. In balanced data, xtgee, fam(gauss) link(ident) corr(exch) and xtreg, mle produce the same results. With unbalanced data, the results are close but differ because the two estimators handle unbalanced data differently. For both balanced and unbalanced data, the results produced by xtgee, fam(gauss) link(ident) corr(exch) and xtreg, mle differ from those produced by xtreg, re. Below we demonstrate the use of the three estimators with unbalanced data. We begin with xtgee; show the maximum likelihood estimator xtreg, mle; show the GLS estimator xtreg, re; and finally show xtgee with the vce(robust) option.

```
. xtgee ln_w grade age c.age#c.age, nolog

GEE population-averaged model                    Number of obs    =  16,085
Group variable: idcode                           Number of groups =   3,913
Family: Gaussian                                 Obs per group:
Link:   Identity                                              min =       1
Correlation: exchangeable                                     avg =     4.1
                                                              max =       9
                                                 Wald chi2(3)     = 2918.26
Scale parameter = .1416586                       Prob > chi2      =  0.0000

──────────────────────────────────────────────────────────────────────────────
     ln_wage │ Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
─────────────┼────────────────────────────────────────────────────────────────
       grade │   .0717731     .00211    34.02   0.000     .0676377    .0759086
         age │   .1077645    .006885    15.65   0.000     .0942701    .1212589
             │
 c.age#c.age │  -.0016381   .0001362   -12.03   0.000     -.001905   -.0013712
             │
       _cons │  -.9480449   .0869277   -10.91   0.000     -1.11842   -.7776698
──────────────────────────────────────────────────────────────────────────────
```

```
. xtreg ln_w grade age c.age#c.age, mle

Fitting constant-only model:
Iteration 0:  Log likelihood = -5868.3483
Iteration 1:  Log likelihood = -5858.8833
Iteration 2:  Log likelihood = -5858.8244

Fitting full model:
Iteration 0:  Log likelihood = -4591.9241
Iteration 1:  Log likelihood = -4562.4406
Iteration 2:  Log likelihood = -4562.3526
Iteration 3:  Log likelihood = -4562.3525

Random-effects ML regression                    Number of obs     =  16,085
Group variable: idcode                          Number of groups =   3,913

Random effects u_i ~ Gaussian                   Obs per group:
                                                             min =       1
                                                             avg =     4.1
                                                             max =       9

                                                LR chi2(3)        = 2592.94
Log likelihood = -4562.3525                     Prob > chi2       =  0.0000

------------------------------------------------------------------------------
     ln_wage | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       grade |   .0717747    .002142    33.51   0.000     .0675765     .075973
         age |   .1077899   .0068266    15.79   0.000     .0944101    .1211697
             |
 c.age#c.age |  -.0016364    .000135   -12.12   0.000    -.0019011   -.0013718
             |
       _cons |  -.9500833    .086384   -11.00   0.000    -1.119393   -.7807737
-------------+----------------------------------------------------------------
    /sigma_u |   .2689639   .0040854                      .2610748    .2770915
    /sigma_e |   .2669944   .0017113                      .2636613    .2703696
         rho |   .5036748   .0086449                      .4867329      .52061
------------------------------------------------------------------------------
LR test of sigma_u=0: chibar2(01) = 4996.22           Prob >= chibar2 = 0.000
```

```
. xtreg ln_w grade age c.age#c.age, re

Random-effects GLS regression                   Number of obs     =     16,085
Group variable: idcode                          Number of groups  =      3,913

R-squared:                                      Obs per group:
     Within  = 0.0983                                         min =          1
     Between = 0.2946                                         avg =        4.1
     Overall = 0.2076                                         max =          9

                                                Wald chi2(3)      =    2875.02
corr(u_i, X) = 0 (assumed)                      Prob > chi2       =     0.0000

------------------------------------------------------------------------------
     ln_wage | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       grade |   .0717757   .0021666    33.13   0.000     .0675294    .0760221
         age |   .1078042   .0068125    15.82   0.000     .0944519    .1211566
             |
 c.age#c.age |  -.0016355   .0001347   -12.14   0.000    -.0018996   -.0013714
             |
       _cons |  -.9512118   .0863139   -11.02   0.000    -1.120384   -.7820397
-------------+----------------------------------------------------------------
     sigma_u |  .27383747
     sigma_e |  .26624266
         rho |  .51405959   (fraction of variance due to u_i)
------------------------------------------------------------------------------

. xtgee ln_w grade age c.age#c.age, vce(robust) nolog

GEE population-averaged model                       Number of obs    =  16,085
Group variable: idcode                              Number of groups =   3,913
Family: Gaussian                                    Obs per group:
Link:   Identity                                                 min =       1
Correlation: exchangeable                                        avg =     4.1
                                                                 max =       9
                                                    Wald chi2(3)     = 2031.28
Scale parameter = .1416586                          Prob > chi2      =  0.0000

                                 (Std. err. adjusted for clustering on idcode)
------------------------------------------------------------------------------
             |               Robust
     ln_wage | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       grade |   .0717731   .0023341    30.75   0.000     .0671983    .0763479
         age |   .1077645   .0098097    10.99   0.000     .0885379    .1269911
             |
 c.age#c.age |  -.0016381   .0001964    -8.34   0.000     -.002023   -.0012532
             |
       _cons |  -.9480449   .1195009    -7.93   0.000    -1.182262   -.7138274
------------------------------------------------------------------------------
```

In [R] **regress**, `regress, vce(cluster` *clustvar*`)` may produce inefficient coefficient estimates with valid standard errors for random-effects models. These standard errors are robust to model misspecification. The `vce(robust)` option of `xtgee`, on the other hand, requires that the model correctly specify the mean and the link function when the noncanonical link is used.

◁

Stored results

`xtgee` 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(tol)`	target tolerance
`e(dif)`	achieved tolerance
`e(rank)`	rank of `e(V)`
`e(rc)`	return code

Macros

`e(cmd)`	`xtgee`
`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)`	distribution family
`e(link)`	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(estat_cmd)`	program used to implement `estat`
`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

Methods and formulas are presented under the following headings:

Introduction

`xtgee` fits generalized linear models for panel data with the GEE approach described in Liang and Zeger (1986). A related method, referred to as GEE2, is described in Zhao and Prentice (1990) and Prentice and Zhao (1991). The GEE2 method attempts to gain efficiency in the estimation of $\boldsymbol{\beta}$ by specifying a parametric model for $\boldsymbol{\alpha}$ and then assumes that the models for both the mean and dependency parameters are correct. Thus there is a tradeoff in robustness for efficiency. The preliminary work of Liang, Zeger, and Qaqish (1992), however, indicates that there is little efficiency gained with this alternative approach.

In the GLM approach (see McCullagh and Nelder [1989]), we assume that

$$\begin{aligned} h(\boldsymbol{\mu}_{i,j}) &= x_{i,j}^{\mathrm{T}}\boldsymbol{\beta} \\ \mathrm{Var}(y_{i,j}) &= g(\mu_{i,j})\phi \\ \boldsymbol{\mu}_i = E(\mathbf{y}_i) &= \{h^{-1}(x_{i,1}^{\mathrm{T}}\boldsymbol{\beta}),\dots,h^{-1}(x_{i,n_i}^{\mathrm{T}}\boldsymbol{\beta})\}^{\mathrm{T}} \\ \mathbf{A}_i &= \mathrm{diag}\{g(\mu_{i,1}),\dots,g(\mu_{i,n_i})\} \\ \mathrm{Cov}(\mathbf{y}_i) &= \phi\mathbf{A}_i \quad \text{for independent observations.} \end{aligned}$$

In the absence of a convenient likelihood function with which to work, we can rely on a multivariate analog of the quasiscore function introduced by Wedderburn (1974):

$$\mathbf{S}_{\boldsymbol{\beta}}(\boldsymbol{\beta},\boldsymbol{\alpha}) = \sum_{i=1}^{m}\left(\frac{\partial\boldsymbol{\mu}_i}{\partial\boldsymbol{\beta}}\right)^{\mathrm{T}}\mathrm{Var}(\mathbf{y}_i)^{-1}(\mathbf{y}_i-\boldsymbol{\mu}_i) = 0$$

We can solve for correlation parameters $\boldsymbol{\alpha}$ by simultaneously solving

$$\mathbf{S}_{\boldsymbol{\alpha}}(\boldsymbol{\beta},\boldsymbol{\alpha}) = \sum_{i=1}^{m}\left(\frac{\partial\boldsymbol{\eta}_i}{\partial\boldsymbol{\alpha}}\right)^{\mathrm{T}}\mathbf{H}_i^{-1}(\mathbf{W}_i-\boldsymbol{\eta}_i) = 0$$

In the GEE approach to GLM, we let $\mathbf{R}_i(\boldsymbol{\alpha})$ be a "working" correlation matrix depending on the parameters in $\boldsymbol{\alpha}$ (see the *Correlation structures* section for the number of parameters), and we estimate $\boldsymbol{\beta}$ by solving the GEE,

$$\mathbf{U}(\boldsymbol{\beta}) = \sum_{i=1}^{m} \left(\frac{\partial \boldsymbol{\mu}_i}{\partial \boldsymbol{\beta}}\right)^{\mathrm{T}} \mathbf{V}_i^{-1}(\boldsymbol{\alpha})(\mathbf{y}_i - \boldsymbol{\mu}_i) = 0$$

$$\text{where } \mathbf{V}_i(\boldsymbol{\alpha}) = \mathbf{A}_i^{1/2}\mathbf{R}_i(\boldsymbol{\alpha})\mathbf{A}_i^{1/2}$$

To solve this equation, we need only a crude approximation of the variance matrix, which we can obtain from a Taylor series expansion, where

$$\begin{aligned} \mathrm{Cov}(\mathbf{y}_i) &= \mathbf{L}_i\mathbf{Z}_i\mathbf{D}_i\mathbf{Z}_i^{\mathrm{T}}\mathbf{L}_i + \phi\mathbf{A}_i = \widetilde{\mathbf{V}}_i \\ \mathbf{L}_i &= \mathrm{diag}\{\partial h^{-1}(u)/\partial u, u = x_{i,j}^{\mathrm{T}}\boldsymbol{\beta}, j = 1, \ldots, n_i\} \end{aligned}$$

which allows that

$$\widehat{\mathbf{D}}_i \approx (\mathbf{Z}_i^{\mathrm{T}}\mathbf{Z}_i)^{-1}\mathbf{Z}_i\widehat{\mathbf{L}}_i^{-1}\left\{(\mathbf{y}_i - \widehat{\boldsymbol{\mu}}_i)(\mathbf{y}_i - \widehat{\boldsymbol{\mu}}_i)^{\mathrm{T}} - \widehat{\phi}\widehat{\mathbf{A}}_i\right\}\widehat{\mathbf{L}}_i^{-1}\mathbf{Z}_i^{\mathrm{T}}(\mathbf{Z}_i'\mathbf{Z}_i)^{-1}$$

$$\hat{\phi} = \sum_{i=1}^{m}\sum_{j=1}^{n_i} \frac{(y_{i,j} - \hat{\mu}_{i,j})^2 - (\widehat{\mathbf{L}}_{i,j})^2\mathbf{Z}_{i,j}^{\mathrm{T}}\widehat{\mathbf{D}}_i\mathbf{Z}_{i,j}}{g(\hat{\mu}_{i,j})}$$

Calculating GEE for GLM

Using the notation from Liang and Zeger (1986), let $\mathbf{y}_i = (y_{i,1}, \ldots, y_{i,n_i})^{\mathrm{T}}$ be the $n_i \times 1$ vector of outcome values, and let $\mathbf{X}_i = (x_{i,1}, \ldots, x_{i,n_i})^{\mathrm{T}}$ be the $n_i \times p$ matrix of covariate values for the ith subject $i = 1, \ldots, m$. We assume that the marginal density for $y_{i,j}$ may be written in exponential family notation as

$$f(y_{i,j}) = \exp\left[\left\{y_{i,j}\theta_{i,j} - a(\theta_{i,j}) + b(y_{i,j})\right\}\phi\right]$$

where $\theta_{i,j} = h(\eta_{i,j}), \eta_{i,j} = x_{i,j}\boldsymbol{\beta}$. Under this formulation, the first two moments are given by

$$E(y_{i,j}) = a'(\theta_{i,j}), \qquad \mathrm{Var}(y_{i,j}) = a''(\theta_{i,j})/\phi$$

In what follows, we let $n_i = n$ without loss of generality. We define the quantities, assuming that we have an $n \times n$ working correlation matrix $\mathbf{R}(\boldsymbol{\alpha})$,

$$\begin{aligned} \boldsymbol{\Delta}_i &= \mathrm{diag}(d\theta_{i,j}/d\eta_{i,j}) && n \times n \text{ matrix} \\ \mathbf{A}_i &= \mathrm{diag}\{a''(\theta_{i,j})\} && n \times n \text{ matrix} \\ \mathbf{S}_i &= \mathbf{y}_i - a'(\boldsymbol{\theta}_i) && n \times 1 \text{ matrix} \\ \mathbf{D}_i &= \mathbf{A}_i\boldsymbol{\Delta}_i\mathbf{X}_i && n \times p \text{ matrix} \\ \mathbf{V}_i &= \mathbf{A}_i^{1/2}\mathbf{R}(\boldsymbol{\alpha})\mathbf{A}_i^{1/2} && n \times n \text{ matrix} \end{aligned}$$

such that the GEE becomes

$$\sum_{i=1}^{m} \mathbf{D}_i^{\mathrm{T}}\mathbf{V}_i^{-1}\mathbf{S}_i = 0$$

We then have that

$$\widehat{\boldsymbol{\beta}}_{j+1} = \widehat{\boldsymbol{\beta}}_j - \left\{\sum_{i=1}^{m} \mathbf{D}_i^{\mathrm{T}}(\widehat{\boldsymbol{\beta}}_j)\widetilde{\mathbf{V}}_i^{-1}(\widehat{\boldsymbol{\beta}}_j)\mathbf{D}_i(\widehat{\boldsymbol{\beta}}_j)\right\}^{-1}\left\{\sum_{i=1}^{m} \mathbf{D}_i^{\mathrm{T}}(\widehat{\boldsymbol{\beta}}_j)\widetilde{\mathbf{V}}_i^{-1}(\widehat{\boldsymbol{\beta}}_j)\mathbf{S}_i(\widehat{\boldsymbol{\beta}}_j)\right\}$$

where the term

$$\left\{\sum_{i=1}^{m} \mathbf{D}_i^{\mathrm{T}}(\widehat{\boldsymbol{\beta}}_j)\widetilde{\mathbf{V}}_i^{-1}(\widehat{\boldsymbol{\beta}}_j)\mathbf{D}_i(\widehat{\boldsymbol{\beta}}_j)\right\}^{-1}$$

is what we call the conventional variance estimate. It is used to calculate the standard errors if the `vce(robust)` option is not specified. This command supports the clustered version of the Huber/White/sandwich estimator of the variance with panels treated as clusters when `vce(robust)` is specified. See [P] **_robust**, particularly *Maximum likelihood estimators* and *Methods and formulas*. Liang and Zeger (1986) also discuss the calculation of the robust variance estimator.

Define the following:

$$\begin{aligned}
\mathbf{D} &= (\mathbf{D}_1^{\mathrm{T}}, \dots, \mathbf{D}_m^{\mathrm{T}})\\
\mathbf{S} &= (\mathbf{S}_1^{\mathrm{T}}, \dots, \mathbf{S}_m^{\mathrm{T}})^{\mathrm{T}}\\
\widetilde{\mathbf{V}} &= nm \times nm \text{ block diagonal matrix with } \widetilde{\mathbf{V}}_i\\
\mathbf{Z} &= \mathbf{D}\boldsymbol{\beta} - \mathbf{S}
\end{aligned}$$

At a given iteration, the correlation parameters $\boldsymbol{\alpha}$ and scale parameter ϕ can be estimated from the current Pearson residuals, defined by

$$\widehat{r}_{i,j} = \{y_{i,j} - a'(\widehat{\theta}_{i,j})\}/\{a''(\widehat{\theta}_{i,j})\}^{1/2}$$

where $\widehat{\theta}_{i,j}$ depends on the current value for $\widehat{\boldsymbol{\beta}}$. We can then estimate ϕ by

$$\widehat{\phi}^{-1} = \sum_{i=1}^{m}\sum_{j=1}^{n_i} \widehat{r}_{i,j}^{\,2}/(N-p)$$

As this general derivation is complicated, let's follow the derivation of the Gaussian family with the identity link (regression) to illustrate the generalization. After making appropriate substitutions, we will see a familiar updating equation. First, we rewrite the updating equation for $\boldsymbol{\beta}$ as

$$\widehat{\boldsymbol{\beta}}_{j+1} = \widehat{\boldsymbol{\beta}}_j - \mathbf{Z}_1^{-1}\mathbf{Z}_2$$

and then derive $\mathbf{Z}_1$ and $\mathbf{Z}_2$.

$$\begin{aligned}
\mathbf{Z}_1 &= \sum_{i=1}^{m} \mathbf{D}_i^{\mathrm{T}}(\widehat{\boldsymbol{\beta}}_j)\widetilde{\mathbf{V}}_i^{-1}(\widehat{\boldsymbol{\beta}}_j)\mathbf{D}_i(\widehat{\boldsymbol{\beta}}_j) = \sum_{i=1}^{m} \mathbf{X}_i^{\mathrm{T}}\boldsymbol{\Delta}_i^{\mathrm{T}}\mathbf{A}_i^{\mathrm{T}}\{\mathbf{A}_i^{1/2}\mathbf{R}(\boldsymbol{\alpha})\mathbf{A}_i^{1/2}\}^{-1}\mathbf{A}_i\boldsymbol{\Delta}_i\mathbf{X}_i\\
&= \sum_{i=1}^{m} \mathbf{X}_i^{\mathrm{T}} \operatorname{diag}\left\{\frac{\partial\theta_{i,j}}{\partial(\mathbf{X}\boldsymbol{\beta})}\right\} \operatorname{diag}\left\{a''(\theta_{i,j})\right\} \left[\operatorname{diag}\left\{a''(\theta_{i,j})\right\}^{1/2}\mathbf{R}(\boldsymbol{\alpha})\operatorname{diag}\left\{a''(\theta_{i,j})\right\}^{1/2}\right]^{-1}\\
&\qquad \operatorname{diag}\left\{a''(\theta_{i,j})\right\}\operatorname{diag}\left\{\frac{\partial\theta_{i,j}}{\partial(\mathbf{X}\boldsymbol{\beta})}\right\}\mathbf{X}_i\\
&= \sum_{i=1}^{m} \mathbf{X}_i^{\mathrm{T}}\mathbf{II}(\mathbf{III})^{-1}\mathbf{II}\mathbf{X}_i = \sum_{i=1}^{m} \mathbf{X}_i^{\mathrm{T}}\mathbf{X}_i = \mathbf{X}^{\mathrm{T}}\mathbf{X}
\end{aligned}$$

$$
\begin{aligned}
\mathbf{Z}_2 &= \sum_{i=1}^{m} \mathbf{D}_i^{\mathrm{T}}(\widehat{\boldsymbol{\beta}}_j)\widetilde{\mathbf{V}}_i^{-1}(\widehat{\boldsymbol{\beta}}_j)\mathbf{S}_i(\widehat{\boldsymbol{\beta}}_j) = \sum_{i=1}^{m} \mathbf{X}_i^{\mathrm{T}}\boldsymbol{\Delta}_i^{\mathrm{T}}\mathbf{A}_i^{\mathrm{T}}\{\mathbf{A}_i^{1/2}\mathbf{R}(\boldsymbol{\alpha})\mathbf{A}_i^{1/2}\}^{-1}\left(\mathbf{y}_i - \mathbf{X}_i\widehat{\boldsymbol{\beta}}_j\right) \\
&= \sum_{i=1}^{m} \mathbf{X}_i^{\mathrm{T}} \operatorname{diag}\left\{\frac{\partial\theta_{i,j}}{\partial(\mathbf{X}\boldsymbol{\beta})}\right\} \operatorname{diag}\left\{a''(\theta_{i,j})\right\} \left[\operatorname{diag}\left\{a''(\theta_{i,j})\right\}^{1/2}\mathbf{R}(\boldsymbol{\alpha})\operatorname{diag}\left\{a''(\theta_{i,j})\right\}^{1/2}\right]^{-1} \\
&\qquad \left(\mathbf{y}_i - \mathbf{X}_i\widehat{\boldsymbol{\beta}}_j\right) \\
&= \sum_{i=1}^{m} \mathbf{X}_i^{\mathrm{T}}\mathbf{II}(\mathbf{III})^{-1}(\mathbf{y}_i - \mathbf{X}_i\widehat{\boldsymbol{\beta}}_j) = \sum_{i=1}^{m} \mathbf{X}_i^{\mathrm{T}}(\mathbf{y}_i - \mathbf{X}_i\widehat{\boldsymbol{\beta}}_j) = \mathbf{X}^{\mathrm{T}}\widehat{s}_j
\end{aligned}
$$

So, we may write the update formula as

$$\widehat{\boldsymbol{\beta}}_{j+1} = \widehat{\boldsymbol{\beta}}_j - (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\widehat{s}_j$$

which is the same formula for GLS in regression.

Correlation structures

The working correlation matrix $\mathbf{R}$ is a function of $\boldsymbol{\alpha}$ and is more accurately written as $\mathbf{R}(\boldsymbol{\alpha})$. Depending on the assumed correlation structure, $\boldsymbol{\alpha}$ might be

Independent	no parameters to estimate
Exchangeable	$\boldsymbol{\alpha}$ is a scalar
Autoregressive	$\boldsymbol{\alpha}$ is a vector
Stationary	$\boldsymbol{\alpha}$ is a vector
Nonstationary	$\boldsymbol{\alpha}$ is a matrix
Unstructured	$\boldsymbol{\alpha}$ is a matrix

Also, throughout the estimation of a general unbalanced panel, it is more proper to discuss $\mathbf{R}_i$, which is the upper left $n_i \times n_i$ submatrix of the ultimately stored matrix in `e(R)`, $\max\{n_i\} \times \max\{n_i\}$.

The only panels that enter into the estimation for a lag-dependent correlation structure are those with $n_i > g$ (assuming a lag of g). `xtgee` drops panels with too few observations (and mentions when it does so).

Independent

The working correlation matrix $\mathbf{R}$ is an identity matrix.

Exchangeable

$$\boldsymbol{\alpha} = \frac{\sum_{i=1}^{m}\left(\sum_{j=1}^{n_i}\sum_{k=1}^{n_i}\widehat{r}_{i,j}\widehat{r}_{i,k} - \sum_{j=1}^{n_i}\widehat{r}_{i,j}^{\,2}\right)}{\sum_{i=1}^{m}\{n_i(n_i-1)\}} \Bigg/ \frac{\sum_{i=1}^{m}\left(\sum_{j=1}^{n_i}\widehat{r}_{i,j}^{\,2}\right)}{\sum_{i=1}^{m} n_i}$$

and the working correlation matrix is given by

$$\mathbf{R}_{s,t} = \begin{cases} 1 & s = t \\ \alpha & \text{otherwise} \end{cases}$$

Autoregressive and stationary

These two structures require g parameters to be estimated so that $\boldsymbol{\alpha}$ is a vector of length $g+1$ (the first element of $\boldsymbol{\alpha}$ is 1).

$$\boldsymbol{\alpha} = \sum_{i=1}^{m} \left(\frac{\sum_{j=1}^{n_i} \widehat{r}_{i,j}^{\,2}}{n_i} \;,\; \frac{\sum_{j=1}^{n_i-1} \widehat{r}_{i,j}\widehat{r}_{i,j+1}}{n_i} \;,\; \dots \;,\; \frac{\sum_{j=1}^{n_i-g} \widehat{r}_{i,j}\widehat{r}_{i,j+g}}{n_i} \right) \Bigg/ \left(\sum_{i=1}^{m} \frac{\sum_{j=1}^{n_i} \widehat{r}_{i,j}^{\,2}}{n_i} \right)$$

The working correlation matrix for the AR model is calculated as a function of Toeplitz matrices formed from the $\boldsymbol{\alpha}$ vector; see Newton (1988). The working correlation matrix for the stationary model is given by

$$\mathbf{R}_{s,t} = \begin{cases} \boldsymbol{\alpha}_{1,|s-t|} & \text{if } |s-t| \le g \\ 0 & \text{otherwise} \end{cases}$$

Nonstationary and unstructured

These two correlation structures require a matrix of parameters. $\boldsymbol{\alpha}$ is estimated (where we replace $\widehat{r}_{i,j} = 0$ whenever $i > n_i$ or $j > n_i$) as

$$\boldsymbol{\alpha} = \sum_{i=1}^{m} m \begin{pmatrix} N_{1,1}^{-1}\widehat{r}_{i,1}^{\,2} & N_{1,2}^{-1}\widehat{r}_{i,1}\widehat{r}_{i,2} & \cdots & N_{1,n}^{-1}\widehat{r}_{i,1}\widehat{r}_{i,n} \\ N_{2,1}^{-1}\widehat{r}_{i,2}\widehat{r}_{i,1} & N_{2,2}^{-1}\widehat{r}_{i,2}^{\,2} & \cdots & N_{2,n}^{-1}\widehat{r}_{i,2}\widehat{r}_{i,n} \\ \vdots & \vdots & \ddots & \vdots \\ N_{n,1}^{-1}\widehat{r}_{i,n_i}\widehat{r}_{i,1} & N_{n,2}^{-1}\widehat{r}_{i,n_i}\widehat{r}_{i,2} & \cdots & N_{n,n}^{-1}\widehat{r}_{i,n}^{\,2} \end{pmatrix} \Bigg/ \left(\sum_{i=1}^{m} \frac{\sum_{j=1}^{n_i} \widehat{r}_{i,j}^{\,2}}{n_i} \right)$$

where $N_{p,q} = \sum_{i=1}^{m} I(i,p,q)$ and

$$I(i,p,q) = \begin{cases} 1 & \text{if panel } i \text{ has valid observations at times } p \text{ and } q \\ 0 & \text{otherwise} \end{cases}$$

where $N_{i,j} = \min(N_i, N_j)$, $N_i =$ number of panels observed at time i, and $n = \max(n_1, n_2, \dots, n_m)$.

The working correlation matrix for the nonstationary model is given by

$$\mathbf{R}_{s,t} = \begin{cases} 1 & \text{if } s = t \\ \boldsymbol{\alpha}_{s,t} & \text{if } 0 < |s-t| \le g \\ 0 & \text{otherwise} \end{cases}$$

The working correlation matrix for the unstructured model is given by

$$\mathbf{R}_{s,t} = \begin{cases} 1 & \text{if } s = t \\ \boldsymbol{\alpha}_{s,t} & \text{otherwise} \end{cases}$$

such that the unstructured model is equal to the nonstationary model at lag $g = n - 1$, where the panels are balanced with $n_i = n$ for all i.

References

Aloisio, K. M., N. Micali, S. A. Swanson, A. Field, and N. J. Horton. 2014. Analysis of partially observed clustered data using generalized estimating equations and multiple imputation. *Stata Journal* 14: 863–883.

Cabanillas, O. B., J. D. Michler, A. Michuda, and E. Tjernström. 2018. Fitting and interpreting correlated random-coefficient models using Stata. *Stata Journal* 18: 159–173.

Caria, M. P., M. R. Galanti, R. Bellocco, and N. J. Horton. 2011. The impact of different sources of body mass index assessment on smoking onset: An application of multiple-source information models. *Stata Journal* 11: 386–402.

Cui, J. 2007. QIC program and model selection in GEE analyses. *Stata Journal* 7: 209–220.

Earnest, A. 2017. *Essentials of a Successful Biostatistical Collaboration*. Boca Raton, FL: CRC Press.

Gallis, J. A., F. Li, and E. L. Turner. 2020. xtgeebcv: A command for bias-corrected sandwich variance estimation for GEE analyses of cluster randomized trials. *Stata Journal* 20: 363–381.

Hardin, J. W., and J. M. Hilbe. 2013. *Generalized Estimating Equations*. 2nd ed. Boca Raton, FL: Chapman and Hall/CRC.

Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. *Applied Logistic Regression*. 3rd ed. Hoboken, NJ: Wiley.

Lenzi, J., and S. Pildava. 2019. Tips for calculating and displaying risk-standardized hospital outcomes in Stata. *Stata Journal* 19: 477–496.

Liang, K.-Y. 1987. Estimating functions and approximate conditional likelihood. *Biometrika* 4: 695–702. https://doi.org/10.2307/2336463.

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.

Liang, K.-Y., S. L. Zeger, and B. Qaqish. 1992. Multivariate regression analyses for categorical data. *Journal of the Royal Statistical Society*, B ser., 54: 3–40. https://doi.org/10.1111/j.2517-6161.1992.tb01862.x.

McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models*. 2nd ed. London: Chapman and Hall/CRC.

Nelder, J. A., and R. W. M. Wedderburn. 1972. Generalized linear models. *Journal of the Royal Statistical Society*, A ser., 135: 370–384. https://doi.org/10.2307/2344614.

Newton, H. J. 1988. *TIMESLAB: A Time Series Analysis Laboratory*. Belmont, CA: Wadsworth.

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.

Prentice, R. L., and L. P. Zhao. 1991. Estimating equations for parameters in means and covariances of multivariate discrete and continuous responses. *Biometrics* 47: 825–839. https://doi.org/10.2307/2532642.

Rabe-Hesketh, S., A. Skrondal, and A. Pickles. 2002. Reliable estimation of generalized linear mixed models using adaptive quadrature. *Stata Journal* 2: 1–21.

Shults, J., and J. M. Hilbe. 2014. *Quasi-Least Squares Regression*. Boca Raton, FL: Chapman and Hall/CRC.

Shults, J., S. J. Ratcliffe, and M. Leonard. 2007. Improved generalized estimating equation analysis via xtqls for quasi-least squares in Stata. *Stata Journal* 7: 147–166.

Twisk, J. W. R. 2013. *Applied Longitudinal Data Analysis for Epidemiology: A Practical Guide*. 2nd ed. Cambridge: Cambridge University Press.

Wedderburn, R. W. M. 1974. Quasi-likelihood functions, generalized linear models, and the Gauss–Newton method. *Biometrika* 61: 439–447. https://doi.org/10.2307/2334725.

Zeger, S. L., and K.-Y. Liang. 1986. Longitudinal data analysis for discrete and continuous outcomes. *Biometrics* 42: 121–130. https://doi.org/10.2307/2531248.

Zeger, S. L., K.-Y. Liang, and P. S. Albert. 1988. Models for longitudinal data: A generalized estimating equation approach. *Biometrics* 44: 1049–1060. https://doi.org/10.2307/2531734.

Zhao, L. P., and R. L. Prentice. 1990. Correlated binary regression using a quadratic exponential model. *Biometrika* 77: 642–648. https://doi.org/10.2307/2337004.

Also see

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

[XT] **xtcloglog** — Random-effects and population-averaged cloglog models

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

[XT] **xtnbreg** — Fixed-effects, random-effects, & population-averaged negative binomial models

[XT] **xtpoisson** — Fixed-effects, random-effects, and population-averaged Poisson models

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

[XT] **xtreg** — Linear models for panel data

[XT] **xtregar** — Fixed- and random-effects linear models with an AR(1) disturbance

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

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

[R] **glm** — Generalized linear models

[R] **logistic** — Logistic regression, reporting odds ratios

[R] **regress** — Linear regression

[U] 20 Estimation and postestimation commands

For suggested citations, see the FAQ on citing Stata documentation.