0.1 `probit.gee`: Generalized Estimating Equation for Probit Regression

The GEE probit estimates the same model as the standard probit regression (appropriate when you have a dichotomous dependent variable and a set of explanatory variables). Unlike in probit regression, GEE probit allows for dependence within clusters, such as in longitudinal data, although its use is not limited to just panel data. The user must first specify a "working" correlation matrix for the clusters, which models the dependence of each observation with other observations in the same cluster. The "working" correlation matrix is a $T \times T$ matrix of correlations, where T is the size of the largest cluster and the elements of the matrix are correlations between within-cluster observations. The appeal of GEE models is that it gives consistent estimates of the parameters and consistent estimates of the standard errors can be obtained using a robust "sandwich" estimator even if the "working" correlation matrix is incorrectly specified. If the "working" correlation matrix is correctly specified, GEE models will give more efficient estimates of the parameters. GEE models measure population-averaged effects as opposed to cluster-specific effects (See Zorn (2001)).

Syntax

```
> z.out <- zelig(Y ~ X1 + X2, model = "probit.gee",
                 id = "X3", data = mydata)
> x.out <- setx(z.out)
> s.out <- sim(z.out, x = x.out)
```

where `id` is a variable which identifies the clusters. The data should be sorted by `id` and should be ordered within each cluster when appropriate.

Additional Inputs

- `robust`: defaults to `TRUE`. If `TRUE`, consistent standard errors are estimated using a "sandwich" estimator.

Use the following arguments to specify the structure of the "working" correlations within clusters:

- `corstr`: defaults to `"independence"`. It can take on the following arguments:
  - Independence (`corstr = "independence"`): $\text{cor}(y_{it}, y_{it'}) = 0$, $\forall t, t'$ with $t \neq t'$. It assumes that there is no correlation within the clusters and the model becomes equivalent to standard probit regression. The "working" correlation matrix is the identity matrix.
  - Fixed (`corstr = "fixed"`): If selected, the user must define the "working" correlation matrix with the `R` argument rather than estimating it from the model.

– Stationary m dependent (`corstr = "stat_M_dep"`):

$$\text{cor}(y_{it}, y_{it'}) = \begin{cases} \alpha_{|t-t'|} & \text{if} \quad |t-t'| \leq m \\ 0 & \text{if} \quad |t-t'| > m \end{cases}$$

If (`corstr = "stat_M_dep"`), you must also specify `Mv` $= m$, where m is the number of periods t of dependence. Choose this option when the correlations are assumed to be the same for observations of the same $|t-t'|$ periods apart for $|t-t'| \leq m$.

Sample "working" correlation for Stationary 2 dependence (`Mv`=2)

$$\begin{pmatrix} 1 & \alpha_1 & \alpha_2 & 0 & 0 \\ \alpha_1 & 1 & \alpha_1 & \alpha_2 & 0 \\ \alpha_2 & \alpha_1 & 1 & \alpha_1 & \alpha_2 \\ 0 & \alpha_2 & \alpha_1 & 1 & \alpha_1 \\ 0 & 0 & \alpha_2 & \alpha_1 & 1 \end{pmatrix}$$

– Non-stationary m dependent (`corstr = "non_stat_M_dep"`):

$$\text{cor}(y_{it}, y_{it'}) = \begin{cases} \alpha_{tt'} & \text{if} \quad |t-t'| \leq m \\ 0 & \text{if} \quad |t-t'| > m \end{cases}$$

If (`corstr = "non_stat_M_dep"`), you must also specify `Mv` $= m$, where m is the number of periods t of dependence. This option relaxes the assumption that the correlations are the same for all observations of the same $|t-t'|$ periods apart.

Sample "working" correlation for Non-stationary 2 dependence (`Mv`=2)

$$\begin{pmatrix} 1 & \alpha_{12} & \alpha_{13} & 0 & 0 \\ \alpha_{12} & 1 & \alpha_{23} & \alpha_{24} & 0 \\ \alpha_{13} & \alpha_{23} & 1 & \alpha_{34} & \alpha_{35} \\ 0 & \alpha_{24} & \alpha_{34} & 1 & \alpha_{45} \\ 0 & 0 & \alpha_{35} & \alpha_{45} & 1 \end{pmatrix}$$

– Exchangeable (`corstr = "exchangeable"`): $\text{cor}(y_{it}, y_{it'}) = \alpha$, $\forall t, t'$ with $t \neq t'$. Choose this option if the correlations are assumed to be the same for all observations within the cluster.

Sample "working" correlation for Exchangeable

$$\begin{pmatrix} 1 & \alpha & \alpha & \alpha & \alpha \\ \alpha & 1 & \alpha & \alpha & \alpha \\ \alpha & \alpha & 1 & \alpha & \alpha \\ \alpha & \alpha & \alpha & 1 & \alpha \\ \alpha & \alpha & \alpha & \alpha & 1 \end{pmatrix}$$

– Stationary mth order autoregressive (`corstr = "AR-M"`): If (`corstr = "AR-M"`), you must also specify `Mv` $= m$, where m is the number of periods t of dependence. For example, the first order autoregressive model (AR-1) implies $\text{cor}(y_{it}, y_{it'}) = \alpha^{|t-t'|}, \forall t, t'$ with $t \neq t'$. In AR-1, observation 1 and observation 2 have a correlation of α. Observation 2 and observation 3 also have a correlation of α. Observation 1 and observation 3 have a correlation of α^2, which is a function of how 1 and 2 are correlated (α) multiplied by how 2 and 3 are correlated (α). Observation 1 and 4 have a correlation that is a function of the correlation between 1 and 2, 2 and 3, and 3 and 4, and so forth.

Sample "working" correlation for Stationary AR-1 (`Mv`=1)

$$\begin{pmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 \\ \alpha & 1 & \alpha & \alpha^2 & \alpha^3 \\ \alpha^2 & \alpha & 1 & \alpha & \alpha^2 \\ \alpha^3 & \alpha^2 & \alpha & 1 & \alpha \\ \alpha^4 & \alpha^3 & \alpha^2 & \alpha & 1 \end{pmatrix}$$

– Unstructured (`corstr = "unstructured"`): $\text{cor}(y_{it}, y_{it'}) = \alpha_{tt'}, \forall t, t'$ with $t \neq t'$. No constraints are placed on the correlations, which are then estimated from the data.

- `Mv`: defaults to 1. It specifies the number of periods of correlation and only needs to be specified when `corstr` is `"stat_M_dep"`, `"non_stat_M_dep"`, or `"AR-M"`.
- `R`: defaults to `NULL`. It specifies a user-defined correlation matrix rather than estimating it from the data. The argument is used only when `corstr` is `"fixed"`. The input is a $T \times T$ matrix of correlations, where T is the size of the largest cluster.

Examples

1. Example with Stationary 3 Dependence

   Attaching the sample turnout dataset:

   ```
   > data(turnout)
   ```

   Variable identifying clusters

   ```
   > turnout$cluster <- rep(c(1:200), 10)
   ```

   Sorting by cluster

   ```
   > sorted.turnout <- turnout[order(turnout$cluster), ]
   ```

   Estimating parameter values:

```
> z.out1 <- zelig(vote ~ race + educate, model = "probit.gee",
+     id = "cluster", data = sorted.turnout, robust = TRUE, corstr = "stat_M_dep",
+     Mv = 3)
```

Setting values for the explanatory variables to their default values:

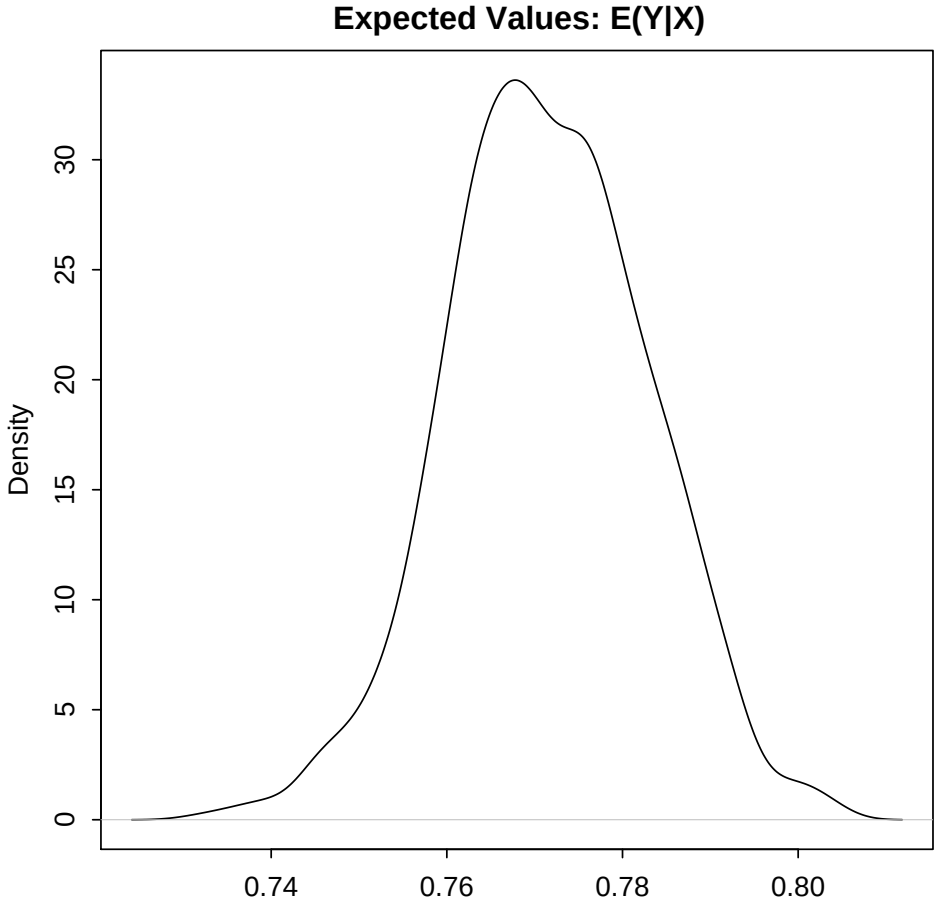
```
> x.out1 <- setx(z.out1)
```

Simulating quantities of interest:

```
> s.out1 <- sim(z.out1, x = x.out1)
```

```
> summary(s.out1)
```

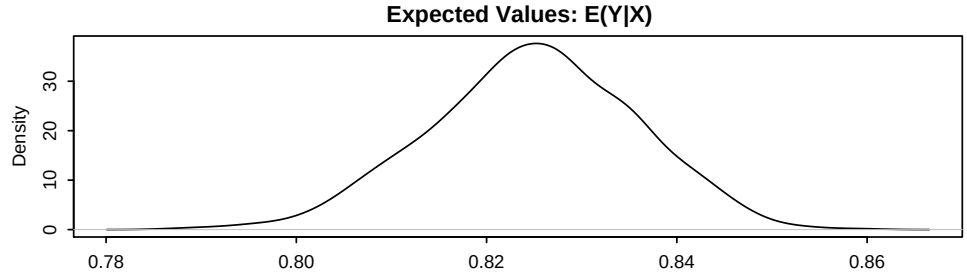
```
> plot(s.out1)
```

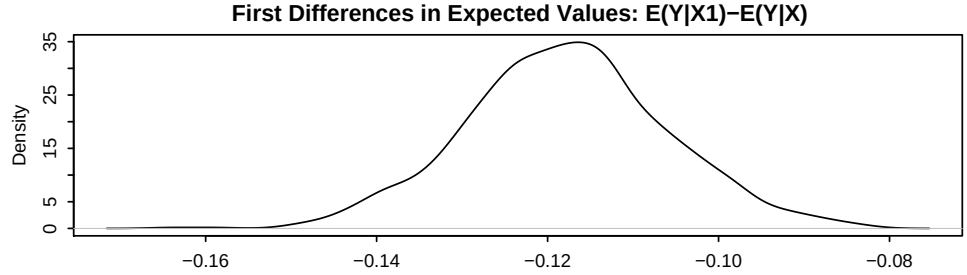


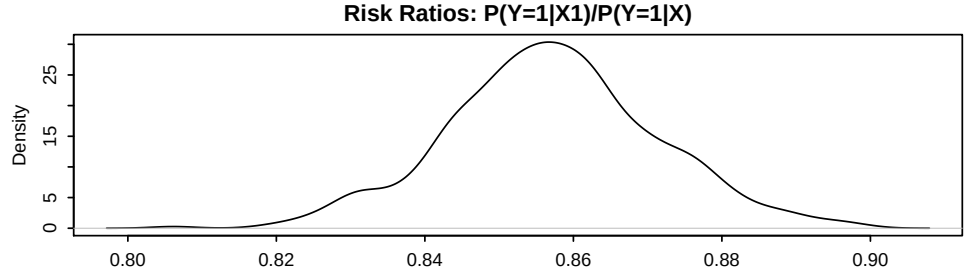
2. Simulating First Differences

   Estimating the risk difference (and risk ratio) between low education (25th percentile) and high education (75th percentile) while all the other variables held at their default values.

```
> x.high <- setx(z.out1, educate = quantile(turnout$educate, prob = 0.75))
> x.low <- setx(z.out1, educate = quantile(turnout$educate, prob = 0.25))

> s.out2 <- sim(z.out1, x = x.high, x1 = x.low)

> summary(s.out2)

> plot(s.out2)
```







3. Example with Fixed Correlation Structure

   User-defined correlation structure

```
> corr.mat <- matrix(rep(0.5, 100), nrow = 10, ncol = 10)
> diag(corr.mat) <- 1
```

Generating empirical estimates:

```
> z.out2 <- zelig(vote ~ race + educate, model = "probit.gee",
+     id = "cluster", data = sorted.turnout, robust = TRUE,
+     corstr = "fixed", R = corr.mat)
```

Viewing the regression output:

```
> summary(z.out2)
```

The Model

Suppose we have a panel dataset, with Y_{it} denoting the binary dependent variable for unit i at time t. Y_i is a vector or cluster of correlated data where y_{it} is correlated with $y_{it'}$ for some or all t, t'. Note that the model assumes correlations within i but independence across i.

- The *stochastic component* is given by the joint and marginal distributions

$$\begin{aligned} Y_i &\sim f(y_i \mid \pi_i) \\ Y_{it} &\sim g(y_{it} \mid \pi_{it}) \end{aligned}$$

  where f and g are unspecified distributions with means π_i and π_{it}. GEE models make no distributional assumptions and only require three specifications: a mean function, a variance function, and a correlation structure.

- The *systematic component* is the *mean function*, given by:

$$\pi_{it} = \Phi(x_{it}\beta)$$

  where $\Phi(\mu)$ is the cumulative distribution function of the Normal distribution with mean 0 and unit variance, x_{it} is the vector of k explanatory variables for unit i at time t and β is the vector of coefficients.

- The *variance function* is given by:

$$V_{it} = \pi_{it}(1 - \pi_{it})$$

- The *correlation structure* is defined by a $T \times T$ "working" correlation matrix, where T is the size of the largest cluster. Users must specify the structure of the "working" correlation matrix *a priori*. The "working" correlation matrix then enters the variance term for each i, given by:

$$V_i = \phi\, A_i^{\frac{1}{2}} R_i(\alpha) A_i^{\frac{1}{2}}$$

  where A_i is a $T \times T$ diagonal matrix with the variance function $V_{it} = \pi_{it}(1 - \pi_{it})$ as the tth diagonal element, $R_i(\alpha)$ is the "working" correlation matrix, and ϕ is a scale parameter. The parameters are then estimated via a quasi-likelihood approach.

- In GEE models, if the mean is correctly specified, but the variance and correlation structure are incorrectly specified, then GEE models provide consistent estimates of the parameters and thus the mean function as well, while consistent estimates of the standard errors can be obtained via a robust "sandwich" estimator. Similarly, if the mean and variance are correctly specified but the correlation structure is incorrectly specified, the parameters can be estimated consistently and the standard errors can be estimated consistently with the sandwich estimator. If all three are specified correctly, then the estimates of the parameters are more efficient.

- The robust "sandwich" estimator gives consistent estimates of the standard errors when the correlations are specified incorrectly only if the number of units i is relatively large and the number of repeated periods t is relatively small. Otherwise, one should use the "naïve" model-based standard errors, which assume that the specified correlations are close approximations to the true underlying correlations. See **?** for more details.

Quantities of Interest

- All quantities of interest are for marginal means rather than joint means.

- The method of bootstrapping generally should not be used in GEE models. If you must bootstrap, bootstrapping should be done within clusters, which is not currently supported in Zelig. For conditional prediction models, data should be matched within clusters.

- The expected values (`qi$ev`) for the GEE probit model are simulations of the predicted probability of a success:

  $$E(Y) = \pi_c = \Phi(x_c\beta),$$

  given draws of β from its sampling distribution, where x_c is a vector of values, one for each independent variable, chosen by the user.

- The first difference (`qi$fd`) for the GEE probit model is defined as

  $$\text{FD} = \Pr(Y = 1 \mid x_1) - \Pr(Y = 1 \mid x).$$

- The risk ratio (`qi$rr`) is defined as

  $$\text{RR} = \Pr(Y = 1 \mid x_1) \;/\; \Pr(Y = 1 \mid x).$$

- In conditional prediction models, the average expected treatment effect (`att.ev`) for the treatment group is

  $$\frac{1}{\sum_{i=1}^{n}\sum_{t=1}^{T} tr_{it}} \sum_{i:tr_{it}=1}^{n} \sum_{t:tr_{it}=1}^{T} \left\{Y_{it}(tr_{it}=1) - E[Y_{it}(tr_{it}=0)]\right\},$$

where tr_{it} is a binary explanatory variable defining the treatment ($tr_{it} = 1$) and control ($tr_{it} = 0$) groups. Variation in the simulations are due to uncertainty in simulating $E[Y_{it}(tr_{it} = 0)]$, the counterfactual expected value of Y_{it} for observations in the treatment group, under the assumption that everything stays the same except that the treatment indicator is switched to $tr_{it} = 0$.

Output Values

The output of each Zelig command contains useful information which you may view. For example, if you run `z.out <- zelig(y ~ x, model = "probit.gee", id, data)`, then you may examine the available information in `z.out` by using `names(z.out)`, see the `coefficients` by using `z.out$coefficients`, and a default summary of information through `summary(z.out)`. Other elements available through the `$` operator are listed below.

- From the `zelig()` output object `z.out`, you may extract:
  - `coefficients`: parameter estimates for the explanatory variables.
  - `residuals`: the working residuals in the final iteration of the fit.
  - `fitted.values`: the vector of fitted values for the systemic component, π_{it}.
  - `linear.predictors`: the vector of $x_{it}\beta$
  - `max.id`: the size of the largest cluster.
- From `summary(z.out)`, you may extract:
  - `coefficients`: the parameter estimates with their associated standard errors, p-values, and z-statistics.
  - `working.correlation`: the "working" correlation matrix
- From the `sim()` output object `s.out`, you may extract quantities of interest arranged as matrices indexed by simulation $\times$ `x`-observation (for more than one `x`-observation). Available quantities are:
  - `qi$ev`: the simulated expected probabilities for the specified values of `x`.
  - `qi$fd`: the simulated first difference in the expected probabilities for the values specified in `x` and `x1`.
  - `qi$rr`: the simulated risk ratio for the expected probabilities simulated from `x` and `x1`.
  - `qi$att.ev`: the simulated average expected treatment effect for the treated from conditional prediction models.

How To Cite

To cite the *probit.gee* Zelig model:

> Patrick Lam. 2007. "probit.gee: General Estimating Equation for Probit Regression" in Kosuke Imai, Gary King, and Olivia Lau, "Zelig: Everyone's Statistical Software,"`http://gking.harvard.edu/zelig`

To cite Zelig as a whole, please reference these two sources:

> Kosuke Imai, Gary King, and Olivia Lau. 2007. "Zelig: Everyone's Statistical Software," `http://GKing.harvard.edu/zelig`.

> Imai, Kosuke, Gary King, and Olivia Lau. (2008). "Toward A Common Framework for Statistical Analysis and Development." Journal of Computational and Graphical Statistics, Vol. 17, No. 4 (December), pp. 892-913.

See also

The gee function is part of the gee package by Vincent J. Carey, ported to R by Thomas Lumley and Brian Ripley. Advanced users may wish to refer to `help(gee)` and `help(family)`. Sample data are from `?`.

Bibliography

Zorn, C. (2001), "Generalized Estimating Equation Models for Correlated Data: A Review with Applications," *American Journal of Political Science*, 45, 470–490.