Package 'PanelSelect'

October 25, 2025

Type Package

Title Panel Sample Selection Models

Version 1.0.0

Maintainer Jing Peng <jing.peng@uconn.edu>

Description Extends the Heckman selection framework to panel data with individual random effects. The first stage models participation via a panel Probit specification, while the second stage can take a panel linear, Probit, Poisson, or Poisson log-normal form. Model details are provided in Bailey and Peng (2025) <doi:10.2139/ssrn.5475626> and Peng and Van den Bulte (2024) <doi:10.1287/mnsc.2019.01897>.

License GPL (>= 3)

Encoding UTF-8

VignetteBuilder knitr

Imports Rcpp, PanelCount, pbivnorm, maxLik, statmod, MASS, data.table, pbv

LinkingTo Rcpp, RcppArmadillo

Suggests knitr, rmarkdown

RoxygenNote 7.3.3

NeedsCompilation yes

Author Jing Peng [aut, cre]

Repository CRAN

Date/Publication 2025-10-25 12:00:02 UTC

Contents

`PanelSelect` *Sample Selection Models for Panel Data*

Description

This package supports a series of panel sample selection models, where the first stage is a panel Probit model with individual random effects and the second stage can be a panel linear, Probit, Poisson, or Poisson log-normal model with individual random effects. Models for count outcome are imported from the PanelCount package.

Functions

probitRE_linearRE: panel sample selection model with continuous outcome

probitRE_probitRE: panel sample selection model with binary outcome

probitRE_PoissonRE: panel sample selection model with count outcome

probitRE_PLNRE: panel sample selection model with count outcome

`probitRE_linearRE` *Panel Sample Selection Model for Continuous Outcome*

Description

A panel sample selection model for continuous outcome, with selection at both the individual and individual-time levels. The outcome is observed in the second stage only if the first stage outcome is one.

Let $\boldsymbol{w}_{it}$ and $\boldsymbol{x}_{it}$ represent the *row* vectors of covariates in the selection and outcome equations, respectively, with $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ denoting the corresponding *column* vectors of parameters.

First stage (probitRE):

$$d_{it} = 1(\mathbf{w}_{it}\boldsymbol{\alpha} + \delta u_i + \varepsilon_{it} > 0)$$

Second stage (linearRE):

$$y_{it} = \mathbf{x}_{it}\boldsymbol{\beta} + \lambda v_i + \sigma\epsilon_{it}$$

Correlation structure: u_i and v_i are bivariate normally distributed with a correlation of ρ. ε_{it} and ϵ_{it} are bivariate normally distributed with a correlation of τ.

w and x can be the same set of variables. Identification can be weak if w are not good predictors of d.

Usage

```
probitRE_linearRE(
  form_probit,
  form_linear,
  id.name,
  data = NULL,
  par = NULL,
  method = "BFGS",
  rho_off = FALSE,
  tau_off = FALSE,
  H = 10,
  init = c("zero", "unif", "norm", "default")[4],
  rho.init = 0,
  tau.init = 0,
  use.optim = FALSE,
  verbose = 0
)
```

Arguments

form_probit	Formula for the panel probit model with random effects at the individual level
form_linear	Formula for the panel linear model with random effects at the individual level
id.name	the name of the id column in data
data	Input data, must be a data.frame object
par	Starting values for estimates
method	Optimization algorithm. Default is BFGS
rho_off	A Boolean value indicating whether to turn off the correlation between the random effects of the probit and linear models. Default is FALSE.
tau_off	A Boolean value indicating whether to turn off the correlation between the error terms of the probit and linear models. Default is FALSE.
H	Number of quadrature points
init	Initialization method
rho.init	Initial value for the correlation between the random effects of the probit and linear models. Default is 0.
tau.init	Initial value for the correlation between the error terms of the probit and linear models. Default is 0.
use.optim	A Boolean value indicating whether to use optim instead of maxLik. Default is FALSE.
verbose	A integer indicating how much output to display during the estimation process. • <0 - No ouput • 0 - Basic output (model estimates) • 1 - Limited output, providing likelihood of iterations • 2 - Moderate output, basic ouput + parameter and likelihood on each call • 3 - Extensive output, moderate output + gradient values on each call

Value

A list containing the results of the estimated model, some of which are inherited from the return of maxLik

- estimates: Model estimates with 95% confidence intervals
- estimate or par: Point estimates
- predict. A list containing the predicted probabilities of responding (respond_prob) and the predicted counterfactual outcome values (outcome), their gradients (gr_respond and gr_outcome), and estimated counterfactual population mean (pop_mean).
- variance_type: covariance matrix used to calculate standard errors. Either BHHH or Hessian.
- var: covariance matrix
- se: standard errors
- var_bhhh: BHHH covariance matrix, inverse of the outer product of gradient at the maximum
- se_bhhh: BHHH standard errors
- gradient: Gradient function at maximum
- hessian: Hessian matrix at maximum
- gtHg: $g'H^{-}1g$, where H^-1 is simply the covariance matrix. A value close to zero (e.g., <1e-3 or 1e-6) indicates good convergence.
- LL or maximum: Likelihood
- AIC: AIC
- BIC: BIC
- n_obs: Number of observations
- n_par: Number of parameters
- time: Time takes to estimate the model
- iterations: number of iterations taken to converge
- message: Message regarding convergence status.

Note that the list inherits all the components in the output of maxLik. See the documentation of maxLik for more details.

References

Bailey, M., & Peng, J. (2025). A Random Effects Model of Non-Ignorable Nonresponse in Panel Survey Data. Available at SSRN https://www.ssrn.com/abstract=5475626

See Also

Other PanelSelect: probitRE_PLNRE(), probitRE_PoissonRE(), probitRE_probitRE()

Examples

```
library(PanelSelect)
library(MASS)
N = 200
period = 5
obs = N*period
rho = 0.5
tau = 0
set.seed(100)

re = mvrnorm(N, mu=c(0,0), Sigma=matrix(c(1,rho,rho,1), nrow=2))
u = rep(re[,1], each=period)
v = rep(re[,2], each=period)
e = mvrnorm(obs, mu=c(0,0), Sigma=matrix(c(1,tau,tau,1), nrow=2))
e1 = e[,1]
e2 = e[,2]

t = rep(1:period, N)
id = rep(1:N, each=period)
w = rnorm(obs)
z = rnorm(obs)
x = rnorm(obs)
d = as.numeric(x + w + u + e1 > 0)
y = x + w + v + e2
y[d==0] = NA
dt = data.frame(id, t, y, x, w, z, d)

# As N increases, the parameter estimates will be more accurate
m = probitRE_linearRE(d~x+w, y~x+w, 'id', dt, H=10, verbose=-1)
print(m$estimates, digits=4)
```

---

probitRE_PLNRE *Poisson Lognormal Model with Random Effects and Sample Selection*

Description

Estimates the following two-stage model:

Selection equation (ProbitRE - Probit model with individual level random effects):

$$z_{it} = 1(\boldsymbol{\alpha}\mathbf{w_{it}}' + \delta u_i + \xi_{it} > 0)$$

Outcome Equation (PLN_RE - Poisson Lognormal model with individual-time level random effects):

$$E[y_{it}|x_{it}, v_i, \epsilon_{it}] = exp(\boldsymbol{\beta}\mathbf{x_{it}}' + \sigma v_i + \gamma\epsilon_{it})$$

Correlation (self-selection at both individual and individual-time level):

- u_i and v_i are bivariate normally distributed with a correlation of ρ.
- ξ_{it} and ϵ_{it} are bivariate normally distributed with a correlation of τ.

Notations:

- w_{it}: variables influencing the selection decision z_{it}, which could be a mixture of time-variant variables, time-invariant variables, and time dummies
- x_{it}: variables influencing the outcome y_{it}, which could be a mixture of time-variant variables, time-invariant variables, and time dummies
- u_i: individual level random effect in the selection equation
- v_i: individual level random effect in the outcome equation
- ξ_{it}: error term in the selection equation
- ϵ_{it}: individual-time level random effect in the outcome equation

Usage

```
probitRE_PLNRE(
  sel_form,
  out_form,
  data,
  id.name,
  testData = NULL,
  par = NULL,
  disable_rho = FALSE,
  disable_tau = FALSE,
  delta = NULL,
  sigma = NULL,
  gamma = NULL,
  rho = NULL,
  tau = NULL,
  method = "BFGS",
  se_type = c("BHHH", "Hessian")[1],
  H = c(10, 10),
  psnH = 20,
  prbH = 20,
  plnreH = 20,
  reltol = sqrt(.Machine$double.eps),
  factr = 1e+07,
  verbose = 1,
  offset_w_name = NULL,
  offset_x_name = NULL
)
```

Arguments

sel_form	Formula for selection equation, a Probit model with random effects
out_form	Formula for outcome equation, a Poisson Lognormal model with random effects
data	Input data, a data.frame object
id.name	The name of the column representing id. Data will be sorted by id to improve estimation speed.

testData	Test data for prediction, a data.frame object
par	Starting values for estimates. Default to estimates of standalone selection and outcome models.
disable_rho	Whether to disable correlation at the individual level random effect. Defaults to FALSE.
disable_tau	Whether to disable correlation at the individual-time level random effect / error term. Defaults to FALSE.
delta	Starting value for delta. Will be ignored if par is provided.
sigma	Starting value for sigma. Will be ignored if par is provided.
gamma	Starting value for gamma. Will be ignored if par is provided.
rho	Starting value for rho. Defaults to 0 and will be ignored if par is provided.
tau	Starting value for tau. Defaults to 0 and will be ignored if par is provided.
method	Optimization method used by optim. Defaults to 'BFGS'.
se_type	Report Hessian or BHHH standard errors. Defaults to BHHH. Hessian matrix is extremely time-consuming to calculate numerically for large datasets.
H	A integer vector of length 2, specifying the number of points for inner and outer Quadratures
psnH	Number of Quadrature points for Poisson RE model
prbH	Number of Quadrature points for Probit RE model
plnreH	Number of Quadrature points for PLN_RE model
reltol	Relative convergence tolerance. The algorithm stops if it is unable to reduce the value by a factor of reltol * (abs(val) + reltol) at a step. Defaults to sqrt(.Machine$double.eps), typically about 1e-8.
factr	L-BFGS-B method uses factr instead of reltol to control for precision. Default is 1e7, that is a tolerance of about 1e-8.
verbose	A integer indicating how much output to display during the estimation process. • <0 - No ouput • 0 - Basic output (model estimates) • 1 - Moderate output, basic ouput + parameter and likelihood in each iteration • 2 - Extensive output, moderate output + gradient values on each call
offset_w_name	An offset variable whose coefficient is assumed to be 1 in the selection equation
offset_x_name	An offset variable whose coefficient is assumed to be 1 in the outcome equation

Value

A list containing the results of the estimated model, some of which are inherited from the return of optim

- estimates: Model estimates with 95% confidence intervals
- par: Point estimates
- var_bhhh: BHHH covariance matrix, inverse of the outer product of gradient at the maximum

- se_bhhh: BHHH standard errors
- g: Gradient function at maximum
- gtHg: $g'H^{-}1g$, where H^-1 is approximated by var_bhhh. A value close to zero (e.g., <1e-3 or 1e-6) indicates good convergence.
- LL: Likelihood
- AIC: AIC
- BIC: BIC
- n_obs: Number of observations
- time: Time takes to estimate the model
- partial: Average partial effect at the population level
- paritalAvgObs: Partial effect for an individual with average characteristics
- predict: A list with predicted participation probability (prob), predicted potential outcome (outcome), and predicted actual outcome (actual_outcome).
- counts: From optim. A two-element integer vector giving the number of calls to fn and gr respectively. This excludes those calls needed to compute the Hessian, if requested, and any calls to fn to compute a finite-difference approximation to the gradient.
- message: From optim. A character string giving any additional information returned by the optimizer, or NULL.
- convergence: From optim. An integer code. 0 indicates successful completion. Note that the list inherits all the complements in the output of optim. See the documentation of optim for more details.

Note

This function is imported from the *PanelCount* package (see ProbitRE_PLNRE for details).

References

1. Peng, J., & Van den Bulte, C. (2023). Participation vs. Effectiveness in Sponsored Tweet Campaigns: A Quality-Quantity Conundrum. Management Science (forthcoming). Available at SSRN: https://www.ssrn.com/abstract=2702053
2. Peng, J., & Van den Bulte, C. (2015). How to Better Target and Incent Paid Endorsers in Social Advertising Campaigns: A Field Experiment. 2015 International Conference on Information Systems. https://aisel.aisnet.org/icis2015/proceedings/SocialMedia/24/

See Also

Other PanelSelect: probitRE_PoissonRE(), probitRE_linearRE(), probitRE_probitRE()

Examples

```
library(MASS)
library(PanelSelect)
set.seed(1)
N = 500
```

```
periods = 5
rho = 0.5
tau = 0

id = rep(1:N, each=periods)
time = rep(1:periods, N)
x = rnorm(N*periods)
w = rnorm(N*periods)

# correlated random effects at the individual level
r = mvrnorm(N, mu=c(0,0), Sigma=matrix(c(1,rho,rho,1), nrow=2))
r1 = rep(r[,1], each=periods)
r2 = rep(r[,2], each=periods)

# correlated error terms at the individual-time level
e = mvrnorm(N*periods, mu=c(0,0), Sigma=matrix(c(1,tau,tau,1), nrow=2))
e1 = e[,1]
e2 = e[,2]

# selection
z = as.numeric(1+x+w+r1+e1>0)
# outcome
y = rpois(N*periods, exp(-1+x+r2+e2))
y[z==0] = NA
dt = data.frame(id,time,x,w,z,y)

# As N increases, the parameter estimates will be more accurate
m = probitRE_PLNRE(z~x+w, y~x, data=sim, id.name='id', verbose=-1)
print(m$estimates, digits=4)
```

---

probitRE_PoissonRE *Poisson RE model with Sample Selection*

Description

Estimates the following two-stage model

Selection equation (ProbitRE - Probit model with individual level random effects):

$$z_{it} = 1(\boldsymbol{\alpha}\mathbf{w_{it}}' + \delta u_i + \xi_{it} > 0)$$

Outcome Equation (PoissonRE - Poisson with individual level random effects):

$$E[y_{it}|x_{it}, v_i] = exp(\boldsymbol{\beta}\mathbf{x_{it}}' + \sigma v_i)$$

Correlation (self-selection at individual level):

- u_i and v_i are bivariate normally distributed with a correlation of ρ.

Notations:

- w_{it}: variables influencing the selection decision z_{it}, which could be a mixture of time-variant variables, time-invariant variables, and time dummies
- x_{it}: variables influencing the outcome y_{it}, which could be a mixture of time-variant variables, time-invariant variables, and time dummies
- u_i: individual level random effect in the selection equation
- v_i: individual level random effect in the outcome equation
- ξ_{it}: error term in the selection equation

Usage

```
probitRE_PoissonRE(
  sel_form,
  out_form,
  data,
  id.name,
  testData = NULL,
  par = NULL,
  delta = NULL,
  sigma = NULL,
  rho = NULL,
  method = "BFGS",
  se_type = c("BHHH", "Hessian")[1],
  H = c(10, 10),
  psnH = 20,
  prbH = 20,
  reltol = sqrt(.Machine$double.eps),
  verbose = 1,
  offset_w_name = NULL,
  offset_x_name = NULL
)
```

Arguments

sel_form	Formula for selection equation, a Probit model with random effects
out_form	Formula for outcome equation, a Poisson model with random effects
data	Input data, a data.frame object
id.name	The name of the column representing id. Data will be sorted by id to improve estimation speed.
testData	Test data for prediction, a data.frame object
par	Starting values for estimates. Default to estimates of standalone selection and outcome models.
delta	Starting value for delta. Will be ignored if par is provided.
sigma	Starting value for sigma. Will be ignored if par is provided.
rho	Starting value for rho. Defaults to 0 and will be ignored if par is provided.
method	Optimization method used by optim. Defaults to 'BFGS'.

se_type	Report Hessian or BHHH standard errors. Defaults to BHHH.
H	A integer vector of length 2, specifying the number of points for inner and outer Quadratures
psnH	Number of Quadrature points for Poisson RE model
prbH	Number of Quddrature points for Probit RE model
reltol	Relative convergence tolerance. The algorithm stops if it is unable to reduce the value by a factor of reltol * (abs(val) + reltol) at a step. Defaults to sqrt(.Machine$double.eps), typically about 1e-8.
verbose	A integer indicating how much output to display during the estimation process. • <0 - No ouput • 0 - Basic output (model estimates) • 1 - Moderate output, basic ouput + parameter and likelihood in each iteration • 2 - Extensive output, moderate output + gradient values on each call
offset_w_name	An offset variable whose coefficient is assumed to be 1 in the selection equation
offset_x_name	An offset variable whose coefficient is assumed to be 1 in the outcome equation

Value

A list containing the results of the estimated model, some of which are inherited from the return of optim

- estimates: Model estimates with 95% confidence intervals
- par: Point estimates
- var_bhhh: BHHH covariance matrix, inverse of the outer product of gradient at the maximum
- se_bhhh: BHHH standard errors
- g: Gradient function at maximum
- gtHg: $g'H^-1g$, where H^-1 is approximated by var_bhhh. A value close to zero (e.g., <1e-3 or 1e-6) indicates good convergence.
- LL: Likelihood
- AIC: AIC
- BIC: BIC
- n_obs: Number of observations
- time: Time takes to estimate the model
- partial: Average partial effect at the population level
- paritalAvgObs: Partial effect for an individual with average characteristics
- predict: A list with predicted participation probability (prob), predicted potential outcome (outcome), and predicted actual outcome (actual_outcome).
- counts: From optim. A two-element integer vector giving the number of calls to fn and gr respectively. This excludes those calls needed to compute the Hessian, if requested, and any calls to fn to compute a finite-difference approximation to the gradient.

- message: From optim. A character string giving any additional information returned by the optimizer, or NULL.
- convergence: From optim. An integer code. 0 indicates successful completion. Note that the list inherits all the complements in the output of optim. See the documentation of optim for more details.

Note

This function is imported from the *PanelCount* package (see ProbitRE_PoissonRE for details).

References

1. Peng, J., & Van den Bulte, C. (2023). Participation vs. Effectiveness in Sponsored Tweet Campaigns: A Quality-Quantity Conundrum. Management Science (forthcoming). Available at SSRN: https://www.ssrn.com/abstract=2702053
2. Peng, J., & Van den Bulte, C. (2015). How to Better Target and Incent Paid Endorsers in Social Advertising Campaigns: A Field Experiment. 2015 International Conference on Information Systems. https://aisel.aisnet.org/icis2015/proceedings/SocialMedia/24/

See Also

Other PanelSelect: probitRE_PLNRE(), probitRE_linearRE(), probitRE_probitRE()

Examples

```
library(MASS)
library(PanelSelect)
set.seed(1)
N = 500
periods = 5
rho = 0.5

id = rep(1:N, each=periods)
time = rep(1:periods, N)
x = rnorm(N*periods)
w = rnorm(N*periods)

# correlated random effects at the individual level
r = mvrnorm(N, mu=c(0,0), Sigma=matrix(c(1,rho,rho,1), nrow=2))
r1 = rep(r[,1], each=periods)
r2 = rep(r[,2], each=periods)

e = rnorm(N*periods)

# selection
z = as.numeric(1+x+w+r1+e>0)
# outcome
y = rpois(N*periods, exp(-1+x+r2))
y[z==0] = NA
dt = data.frame(id,time,x,w,z,y)
```

```
# As N increases, the parameter estimates will be more accurate
m = probitRE_PoissonRE(z~x+w, y~x, data=dt, id.name='id', verbose=-1)
print(m$estimates, digits=4)
```

---

probitRE_probitRE *Panel Sample Selection Model for Binary Outcome*

Description

A panel sample selection model for binary outcome, with selection at both the individual and individual-time levels. The outcome is observed in the second stage only if the first stage outcome is one.

Let $\mathbf{w}_{it}$ and $\mathbf{x}_{it}$ represent the *row* vectors of covariates in the selection and outcome equations, respectively, with $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ denoting the corresponding *column* vectors of parameters.

First stage (probitRE):

$$d_{it} = 1(\mathbf{w}_{it}\boldsymbol{\alpha} + \delta u_i + \varepsilon_{it} > 0)$$

Second stage (probitRE):

$$y_{it} = 1(\mathbf{x}_{it}\boldsymbol{\beta} + \lambda v_i + \epsilon_{it} > 0)$$

Correlation structure: u_i and v_i are bivariate normally distributed with a correlation of ρ. ε_{it} and ϵ_{it} are bivariate normally distributed with a correlation of τ.

w and x can be the same set of variables. Identification can be weak if w are not good predictors of d.

Usage

```
probitRE_probitRE(
  probit1,
  probit2,
  id.name,
  data = NULL,
  par = NULL,
  method = "BFGS",
  rho_off = FALSE,
  tau_off = FALSE,
  H = 10,
  init = c("zero", "unif", "norm", "default")[4],
  rho.init = 0,
  tau.init = 0,
  use.optim = FALSE,
  verbose = 0
)
```

Arguments

probit1	Formula for the first-stage probit model with random effects at the individual level
probit2	Formula for the second-stage probit model with random effects at the individual level
id.name	the name of the id column in data
data	Input data, must be a data.frame object
par	Starting values for estimates
method	Optimization algorithm. Default is BFGS
rho_off	A Boolean value indicating whether to turn off the correlation between the random effects of the probit and linear models. Default is FALSE.
tau_off	A Boolean value indicating whether to turn off the correlation between the error terms of the probit and linear models. Default is FALSE.
H	Number of quadrature points
init	Initialization method
rho.init	Initial value for the correlation between the random effects of the probit and linear models. Default is 0.
tau.init	Initial value for the correlation between the error terms of the probit and linear models. Default is 0.
use.optim	A Boolean value indicating whether to use optim instead of maxLik. Default is FALSE.
verbose	A integer indicating how much output to display during the estimation process. • <0 - No ouput • 0 - Basic output (model estimates) • 1 - Limited output, providing likelihood of iterations • 2 - Moderate output, basic ouput + parameter and likelihood on each call • 3 - Extensive output, moderate output + gradient values on each call

Value

A list containing the results of the estimated model, some of which are inherited from the return of maxLik

- estimates: Model estimates with 95% confidence intervals
- estimate or par: Point estimates
- predict. A list containing the predicted probabilities of responding (respond_prob) and the predicted counterfactual outcome values (outcome_prob), their gradients (gr_respond and gr_outcome), and estimated counterfactual population mean (pop_mean).
- variance_type: covariance matrix used to calculate standard errors. Either BHHH or Hessian.
- var: covariance matrix
- se: standard errors
- var_bhhh: BHHH covariance matrix, inverse of the outer product of gradient at the maximum

- se_bhhh: BHHH standard errors
- gradient: Gradient function at maximum
- hessian: Hessian matrix at maximum
- gtHg: $g'H^{-}1g$, where H^-1 is simply the covariance matrix. A value close to zero (e.g., <1e-3 or 1e-6) indicates good convergence.
- LL or maximum: Likelihood
- AIC: AIC
- BIC: BIC
- n_obs: Number of observations
- n_par: Number of parameters
- time: Time takes to estimate the model
- iterations: number of iterations taken to converge
- message: Message regarding convergence status.

Note that the list inherits all the components in the output of maxLik. See the documentation of maxLik for more details.

References

Bailey, M., & Peng, J. (2025). A Random Effects Model of Non-Ignorable Nonresponse in Panel Survey Data. Available at SSRN https://www.ssrn.com/abstract=5475626

See Also

Other PanelSelect: probitRE_PLNRE(), probitRE_PoissonRE(), probitRE_linearRE()

Examples

```
library(PanelSelect)
library(MASS)
N = 150
period = 5
obs = N*period
rho = 0.5
tau = 0
set.seed(100)

re = mvrnorm(N, mu=c(0,0), Sigma=matrix(c(1,rho,rho,1), nrow=2))
u = rep(re[,1], each=period)
v = rep(re[,2], each=period)
e = mvrnorm(obs, mu=c(0,0), Sigma=matrix(c(1,tau,tau,1), nrow=2))
e1 = e[,1]
e2 = e[,2]

t = rep(1:period, N)
id = rep(1:N, each=period)
w = rnorm(obs)
z = rnorm(obs)
```

```
x = rnorm(obs)
d = as.numeric(x + w + u + e1 > 0)
y = as.numeric(x + w + v + e2 > 0)
y[d==0] = NA
dt = data.frame(id, t, y, x, w, z, d)

# As N increases, the parameter estimates will be more accurate
m = probitRE_probitRE(d~x+w, y~x+w, 'id', dt, H=10, verbose=-1)
print(m$estimates, digits=4)
```

Index