

Package ‘pmlsp’

January 31, 2026

Title Partial Maximum Likelihood Estimation of Spatial Probit Models

Version 1.0.1

Description Estimate spatial autoregressive nonlinear probit models with and without autoregressive disturbances using partial maximum likelihood estimation. Estimation and inference regarding marginal effects is also possible. For more details see Bille and Leorato (2020) <[doi:10.1080/07474938.2019.1682314](https://doi.org/10.1080/07474938.2019.1682314)>.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Imports abind, Matrix, matrixcalc, maxLik, methods, minqa, mvtnorm,
numDeriv, qrng, spatialreg, spdep, stats, utils

Depends R (>= 3.5)

LazyData true

URL <https://github.com/d-spinelli/pmlsp>

BugReports <https://github.com/d-spinelli/pmlsp/issues>

NeedsCompilation no

Author Daniele Spinelli [aut, cre],
Anna Gloria Bille' [aut],
Samantha Leorato [aut]

Maintainer Daniele Spinelli <daniele.spinelli@unimib.it>

Repository CRAN

Date/Publication 2026-01-31 19:00:02 UTC

Contents

ape	2
coef.pmlsprobit	3

eurostat	4
pmlsbp	4
predict.pmlsprobit	8
print.summary.pmlsprobit	9
summary.pmlsprobit	10
vcov.pmlsprobit	11

Index	12
--------------	-----------

ape	<i>Average Marginal Effects for Spatial Probit Models</i>
------------	---

Description

Computes average direct, indirect, and total marginal effects from an object of class "pmlsprobit" using numerical derivatives.

Usage

```
ape(mod, variables = NULL)
```

Arguments

- | | |
|------------------|---|
| mod | An object of class "pmlsprobit". |
| variables | Character vector specifying the names of covariates for which marginal effects are computed. This option should be used only when the model contains non-linear terms in the specification $X\beta$. |

Value

A named list with three elements corresponding to the average direct, indirect, and total marginal effects for each covariate.

See Also

Partial maximum likelihood estimation is implemented in [pmlsbp](#).

Examples

```
data(oldcol, package = "spdep")
dat <- COL.OLD
dat$y <- as.numeric(dat$CRIME > 35)

listw <- spdep::nb2listw(COL.nb, style = "W")

set.seed(857489)
mod <- pmlsbp(
  y ~ HOVAL + INC, data = dat, W = listw,
  model = "SAR", grouping = 7,
```

```
mvtnorm_control = list(  
  M = 25e2, sim_type = "mc",  
  tol = .Machine$double.eps, fast = FALSE  
,  
  na.action = na.omit, spectral = TRUE,  
  iterlim = 1e5  
)  
  
ape(mod)
```

coef.pmlsprobit

Coefficient Estimates for Partial Maximum Likelihood Spatial Probit Models

Description

Extracts the estimated regression coefficients from an object of class "pmlsprobit".

Usage

```
## S3 method for class 'pmlsprobit'  
coef(object, ...)
```

Arguments

object An object of class "pmlsprobit".
... Additional arguments (currently unused).

Value

A named numeric vector containing the coefficient estimates.

See Also

Partial maximum likelihood estimation is implemented in [pmlsbp](#).

eurostat	<i>Eurostat employment data</i>
----------	---------------------------------

Description

The eurostat data frame has 312 row and 10 variables. Unit of analysis: 312 regions in Europe. Additionally, the eurostat.nb includes the neighbour list based on a consider a k-nearest neighbour approach with k = 11

Usage

```
data(eurostat)
```

Format

A data frame with 312 rows and 10 variables:

nuts_id	Region identifier
country_id	Country identfier
employment	Employement rate in percentage
gdp	GDP per capita
isced_02	percentage of residents aged 25-64 with secondary school achievement or lower
isced_34	percentage of residents aged 25-64 with high school achievement or lower
isced_58	percentage of residents aged 25-64 with bachelor degree or higher
long	Longitude
lat	Latitude

Source

<https://ec.europa.eu/eurostat/data/database>

Description

Fits spatial autoregressive probit models using partial maximum likelihood as proposed in Bille and Leorato (2020) [doi:10.1080/07474938.2019.1682314](https://doi.org/10.1080/07474938.2019.1682314). Both Spatial Autoregressive (SAR) probit models and Spatial Autoregressive Probit Models with Autoregressive Disturbances (SARAR) are supported.

Usage

```
pmlsbp(
  formula, data, model = "SAR", grouping = 2,
  W = NULL,
  zero.policy = spatialreg::get.ZeroPolicyOption(),
  M = NULL, formula_xlag = NULL, W2 = NULL,
  method_inv = "solve",
  start = NULL, subset = NULL, na.action = na.fail,
  qu = Inf, iterlim = 1000,
  mvtnorm_control = list(M = 1e4, sim_type = "qmc",
    tol = .Machine$double.eps, fast = FALSE),
  finalHessian = ifelse(method == "bhhh", method, TRUE),
  method = "bhhh", print.level = 2,
  vce.type = "asy",
  Conley = list(coords = NULL, LM = 2),
  nBoot = 1e3, spectral = FALSE,
  verbose = TRUE ,
  tol.solve = .Machine$double.eps,
  version = 0, ...
)

## S3 method for class 'pmlsprobit'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

<code>formula</code>	An object of class "formula" specifying the model.
<code>data</code>	An optional data frame containing the variables in the model.
<code>model</code>	A character string specifying the model type; either "SAR" or "SARAR".
<code>grouping</code>	An integer defining the number of observations to include in the tuples used for estimation.
<code>W</code>	A spatial weights object of class "listw", for example created by nb2listw .
<code>zero.policy</code>	Logical; if TRUE, assigns zero to lagged values for observations without neighbors.
<code>M</code>	A spatial weights object of class "listw" used for the disturbance process; relevant only if <code>model = "SARAR"</code> .
<code>formula_xlag</code>	An optional "formula" specifying covariates to be spatially lagged.
<code>W2</code>	A spatial weights object used to construct spatially lagged covariates; relevant only if <code>formula_xlag</code> is not NULL.
<code>method_inv</code>	Character string specifying the method used to invert $I - \rho W$; one of "solve", "chol", or "fast".
<code>start</code>	Numeric vector of starting values.
<code>subset</code>	Optional vector specifying a subset of observations.
<code>na.action</code>	A function specifying how missing values are handled.

<code>qu</code>	Integer used only if <code>method_inv = "fast"</code> .
<code>iterlim</code>	Integer specifying the maximum number of iterations.
<code>mvtnorm_control</code>	A list of control parameters for multivariate normal probability calculations; see lpmvnorm .
<code>finalHessian</code>	Logical or character specifying how the final Hessian matrix is computed; see maxLik .
<code>method</code>	Maximization method passed to maxLik .
<code>print.level</code>	Integer controlling the amount of diagnostic output.
<code>vce.type</code>	Character specifying the variance–covariance estimator; one of "asy", "bootstrap", or "mConley".
<code>Conley</code>	A list specifying options for the modified Conley estimator.
<code>nBoot</code>	Integer specifying the number of bootstrap replications.
<code>spectral</code>	Logical; if TRUE, uses spectral normalization of W .
<code>verbose</code>	Logical; if TRUE, diagnostic messages are printed during model estimation. The default is TRUE. Setting <code>verbose = FALSE</code> suppresses intermediate output.
<code>tol.solve</code>	Numeric tolerance used when inverting matrices.
<code>version</code>	Integer controlling the computation of bivariate normal integrals when grouping = 2.
<code>...</code>	Additional arguments passed to maxLik .
<code>x</code>	a "pmlsprobit" object
<code>digits</code>	integer, used for number formatting the decimal digits

Details

The SARAR model is defined as

$$y^* = \rho W y + X\beta + W_2 Z\gamma + u$$

with

$$u = \lambda M u + \epsilon,$$

where $\epsilon \sim N(0, \sigma^2)$ and the observed binary outcome is $y = 1(y^* > 0)$. The SAR model is obtained by setting $\lambda = 0$.

Value

An object of class "pmlsprobit", which is a list containing:

- `beta` Named vector of regression coefficients.
- `call` The matched function call.
- `code` Convergence code from [maxLik](#).
- `estimate` Named vector of parameter estimates.
- `f` List of matrices used for prediction.

maximum Value of the partial log-likelihood.
 message Convergence message from maxLik.
 model Model frame used in estimation.
 model.type Either "SAR" or "SARAR".
 rho Estimated spatial autoregressive parameter.
 slx Logical indicating the presence of spatially lagged covariates.
 start Starting values used in estimation.
 terms Terms object.
 vcov Variance–covariance matrix of the estimates.
 W Spatial weights matrix W .
 lambda Estimated disturbance parameter (SARAR only).
 M Spatial weights matrix M (SARAR only).
 W2 Spatial weights matrix W_2 (if slx = TRUE).
 No return value

Functions

- `print(pmlsprobit)`: prints the call and the coefficient estimates of a "pmlsprobit" object

References

Bille', A. G., & Leorato, S. (2020). Partial ML estimation for spatial autoregressive nonlinear probit models with autoregressive disturbances. *Econometric Reviews*, 39(5), 437–475. doi:[10.1080/07474938.2019.1682314](https://doi.org/10.1080/07474938.2019.1682314)

See Also

[pmlsbp](#), [maxLik](#)

Examples

```

data(oldcol, package = "spdep")
dat <- COL.OLD
dat$y <- as.numeric(dat$CRIME > 35)

listw <- spdep::nb2listw(COL.nb, style = "W")

set.seed(857489)
mod <- pmlsbp(
  y ~ HOVAL + INC, data = dat, W = listw,
  model = "SAR", grouping = 7,
  na.action = na.omit, spectral = TRUE,
  iterlim = 1e5
)

```

predict.pmlsprobit *Predicted Values and Marginal Effects for Spatial Probit Models*

Description

Computes predicted probabilities or average marginal effects from an object of class "pmlsprobit".

Usage

```
## S3 method for class 'pmlsprobit'
predict(
  object,
  type = "prob",
  newdata = NULL,
  variables = NULL,
  delta.method = FALSE,
  ...
)
```

Arguments

<code>object</code>	An object of class "pmlsprobit".
<code>type</code>	Character string specifying the type of prediction. Use "prob" for predicted probabilities $P(Y_i = 1)$, or "me" for direct, indirect, and total average marginal effects.
<code>newdata</code>	Optional data frame containing variables used for prediction. If omitted, the fitted values are used.
<code>variables</code>	Character vector of covariate names for which marginal effects are computed using numerical derivatives. Only used if <code>type = "me"</code> .
<code>delta.method</code>	Logical; if TRUE, standard errors of the predictions are computed. Only used if <code>type = "me"</code> .
...	Additional arguments (currently unused).

Value

If `type = "prob"`, a numeric vector of predicted probabilities. If `type = "me"`, a list of matrices containing individual direct, indirect, and total marginal effects for each covariate.

See Also

Partial maximum likelihood estimation is implemented in [pmlsbp](#).

Examples

```

data(olddol, package = "spdep")
dat <- COL.OLD
dat$y <- as.numeric(dat$CRIME > 35)

listw <- spdep::nb2listw(COL.nb, style = "W")

set.seed(857489)
mod <- pmlsbp(
  y ~ HOVAL + INC, data = dat, W = listw,
  model = "SAR", grouping = 7,
  mvtnorm_control = list(
    M = 25e2, sim_type = "mc",
    tol = .Machine$double.eps, fast = FALSE
  ),
  na.action = na.omit, spectral = TRUE,
  iterlim = 1e5
)

prob <- predict(mod)
head(prob)

```

`print.summary.pmlsprobit`

Print method for summary.pmlsprobit objects

Description

S3 method to print objects of class `summary.pmlsprobit`.

Usage

```

## S3 method for class 'summary.pmlsprobit'
print(x, digits = max(3L, getOption("digits") - 3L), ...)

```

Arguments

- `x` An object of class `summary.pmlsprobit`.
- `digits` Number of significant digits to print for coefficients.
- `...` Additional arguments (currently ignored).

Value

No return value

summary.pmlsprobit *Summarizing Partial Maximum Likelihood Spatial Probit Models*

Description

Provides a summary method for objects of class "pmlsprobit".

Usage

```
## S3 method for class 'pmlsprobit'
summary(object, ...)
```

Arguments

object	An object of class "pmlsprobit".
...	Additional arguments (currently unused).

Value

An object of class "summary.pmlsprobit" containing:

- estimate Matrix of estimates, standard errors, z-values, and p-values.
- rho Estimated spatial autoregressive parameter.
- lambda Estimated spatial error parameter.
- model.type Type of spatial probit model fitted.
- loglik Log-likelihood at the optimum.
- iteration Number of iterations performed.
- returnCode Optimizer convergence code.
- returnMessage Message associated with the convergence code.
- vcov Variance-covariance matrix of the parameter estimates.

See Also

Partial maximum likelihood estimation is implemented in [pmlsbp](#).

Examples

```
data(oldcol, package = "spdep")
dat <- COL.OLD
dat$y <- as.numeric(dat$CRIME > 35)

listw <- spdep::nb2listw(COL.nb, style = "W")

set.seed(857489)
mod <- pmlsbp(
  y ~ HOVAL + INC, data = dat, W = listw,
```

```
model = "SAR", grouping = 7,  
na.action = na.omit, spectral = TRUE,  
iterlim = 1e5  
)  
summary(mod)
```

vcov.pmlsprobit

Variance–Covariance Matrix for Partial Maximum Likelihood Spatial Probit Models

Description

Extracts the variance–covariance matrix from an object of class "pmlsprobit".

Usage

```
## S3 method for class 'pmlsprobit'  
vcov(object, ...)
```

Arguments

object	An object of class "pmlsprobit".
...	Additional arguments (currently unused).

Value

A numeric matrix containing the variance–covariance matrix of the parameter estimates.

See Also

Partial maximum likelihood estimation is implemented in [pmlsbp](#).

Index

* **datasets**
 eurostat, [4](#)

ape, [2](#)

coef.pmlsprobit, [3](#)

eurostat, [4](#)

lpmvnorm, [6](#)

maxLik, [6](#), [7](#)

nb2listw, [5](#)

pmlsbp, [2](#), [3](#), [4](#), [7](#), [8](#), [10](#), [11](#)
predict.pmlsprobit, [8](#)
print.pmlsprobit (pmlsbp), [4](#)
print.summary.pmlsprobit, [9](#)

summary.pmlsprobit, [10](#)

vcov.pmlsprobit, [11](#)