

Package ‘ProbitSpatial’

March 8, 2016

Type Package

Title Probit with Spatial Dependence, SAR and SEM Models

Version 1.0

Date 2016-02-22 13:30:07

Author Davide Martinetti and Ghislain Geniaux

Maintainer Davide Martinetti <dmartinetti@paca.inra.fr>

Description Binomial Spatial Probit models for big data.

License GPL (>= 2)

Imports numDeriv, RANN

Depends Matrix, methods, Rcpp, speedglm

LinkingTo RcppEigen, Rcpp

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-03-08 14:21:42

R topics documented:

ProbitSpatial-package	2
coef.SpatialProbit	3
conditional_SAR_UC	4
conditional_SAR_UP	4
conditional_SEM_UC	5
conditional_SEM_UP	5
effects	6
fitted.SpatialProbit	7
generate_W	7
Katrina	8
Katrina.raw	11
names.SpatialProbit	13
predict.SpatialProbit	14
residuals.SpatialProbit	14

sim_binomial_probit	15
SpatialProbit-class	16
SpatialProbitFit	17
summary.SpatialProbit	20

Index	21
--------------	-----------

ProbitSpatial-package *Fit Spatial Probit Models.*

Description

SpatialProbit package allows to fit spatial autoregressive (SAR) and spatial error (SEM) probit models. It also provides functions to simulated spatial binary data, an empirical data set and different methods for the diagnostic of the estimated model.

Details

The main function of this package is `SpatialProbitFit`. It allows to fit both SAR and SEM models for big datasets in a reasonable time. The function is based on the maximisation of the approximate likelihood function. The approximation is inspired by the Mendell and Elston algorithm for computing multivariate normal probabilities and take advantage of the sparsity of the spatial weight matrix. Two methods are available for the estimation of the model parameter: the first one is known as conditional method (see Case (1992)) and performs relatively well in terms of accuracy of the estimated parameters and is very rapid. The second method, that minimises the full-log-likelihood, is slower but it should be more accurate. Monte Carlo experiments on simulated data reported in Martinetti and Geniaux (2015) showed that the full-log-likelihood approach is not always overperforming the conditional method in terms of accuracy. At the present stage, our suggestion is to use the conditional method for a first estimation and only attempt the full-likelihood approach in a second moment, when the dataset size is not bigger than a few thousands.

Another feature of the `SpatialProbitFit` function is the possibility to fit the model using the precision matrix instead of the variance-covariance matrix, since it is usually sparser and hence allows faster computations (see LeSage and Pace (2009)).

The output of `SpatialProbitFit` function is an object of class `SpatialProbit`, for which the methods `residuals`, `fitted`, `effects`, `predict` and `coef` are available.

The package also contains the function `sim_binomial_probit` that allows to simulate data samples of both SAR and SEM models. It can be used to replicate the Monte Carlo experiments reported in Martinetti and Geniaux (2015) as well as the experiment of Calabrese and Elkink (2014). An empirical data set [Katrina](#) on the reopening decisions of firms in the aftermath of the Katrina Hurricane in New Orleans is also available (LeSage et al.(2011)).

Other packages in CRAN repository on the same subject are [McSpatial](#) (McMillen (2013)) and `spatialprobit` (Wilhelm and Godinho de Matos (2013)).

The core functions of the present package have been coded using the `Rcpp` and `RcppEigen` libraries (Bates and Eddelbuettel (2013)), that allow direct interchange of rich R objects between R and C++.

Author(s)

Davide Martinetti <davide.martinetti@paca.inra.fr> and Ghislain Geniaux <ghislain.geniaux@avignon.inra.fr>

References

- Bates and Eddelbuettel (2013)** D. Bates and D. Eddelbuettel. Fast and elegant numerical linear algebra using the RcppEigen package. *Journal of Statistical Software* 52, 1–24, 2013.
- Case (1992)** A. C. Case. Neighborhood Influence and Technological Change. *Regional Science and Urban Economics* 22, 491–508, 1992.
- Calabrese and Elkink (2014)** R. Calabrese and J.A. Elkink. Estimators of binary spatial autoregressive models: a Monte Carlo study. *Journal of Regional Science* 54, 664–687, 2014.
- LeSage and Pace (2009)** J. LeSage and R.K. Pace. *Introduction to Spatial Econometrics*, CRC Press, chapter 10.1.6, 2009.
- LeSage et al. (2011)** P. LeSage, R. K. Pace, N. Lam, R. Campanella and X. Liu. New Orleans business recovery in the aftermath of Hurricane Katrina. *Journal of the Royal Statistical Society A*, 174, 1007–1027, 2011.
- Martinetti and Geniaux (2014)** D. Martinetti and G. Geniaux. Approximate likelihood estimation of spatial probit models. *Regional Science and Urban Economics*, submitted, 2015. <https://urbansimul.paca.inra.fr/urbansimul/pdf/recherche/>
- McMillen (2013)** D. McMillen. McSpatial: Nonparametric spatial data analysis. R package version 2.0, <http://CRAN.R-project.org/package=McSpatial>, 2013.
- Mendell and Elston (1974)** N. Mendell and R. Elston. Multifactorial qualitative traits: genetic analysis and prediction of recurrence risks. *Biometrics* 30, 41–57, 1974.
- Wilhelm and Godinho de Matos (2013)** S. Wilhelm and M. Godinho de Matos. Estimating Spatial Probit Models in R. *The R Journal* 5, 130–143, 2013.

coef.SpatialProbit *Estimated coefficients of a spatial probit model.*

Description

Returns the coefficients estimated by a SpatialProbit model.

Usage

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

Arguments

object	an object of class SpatialProbit.
...	ignored

Value

It returns the value of the estimated parameters.

conditional_SAR_UC *Conditional SAR UC.*

Description

Performs conditional estimation of SAR model with variance-covariance matrix

Usage

```
conditional_SAR_UC(myenv)
```

Arguments

myenv an environment.

Details

We discourage the direct use of this function.

Value

the log-likelihood and the estimated parameters.

conditional_SAR_UP *Conditional SAR UP.*

Description

Performs conditional estimation of SAR model with precision matrix

Usage

```
conditional_SAR_UP(myenv)
```

Arguments

myenv an environment.

Details

We discourage the direct use of this function.

Value

the log-likelihood and the estimated parameters.

conditional_SEM_UC *Conditional SEM UC.*

Description

Performs conditional estimation of SEM model with variance-covariance matrix

Usage

```
conditional_SEM_UC(myenv)
```

Arguments

myenv an environment.

Details

We discourage the direct use of this function.

Value

the log-likelihood and the estimated parameters.

conditional_SEM_UP *Conditional SEM UP.*

Description

Performs conditional estimation of SEM model with precision matrix

Usage

```
conditional_SEM_UP(myenv)
```

Arguments

myenv an environment.

Details

We discourage the direct use of this function.

Value

the log-likelihood and the estimated parameters.

effects	<i>Effects of a spatial probit model.</i>
---------	---

Description

Returns the marginal effects of a SpatialProbit model.

Usage

```
effects(object)
```

Arguments

object an object of class SpatialProbit.

Details

The effects function has different outputs according to the DGP of the SpatialProbit model:

"SAR" The marginal effects of a spatial autoregressive model are more complicated than usual measurements of impacts for non spatial models. Here we follow LeSage and Pace and propose the following summaries for impact measures:

Average direct effects: the average over all the observations of the effects of the change of an explanatory variable of a single observation on the choice probability of that same observation.

Average indirect effects: the average over all the observations of the effect of a change on a explanatory variable on the choice probability of the neighbouring observations.

Average total effects: the sum of direct and indirect impacts.

"SEM" marginal effects should be interpreted as if it were a standard probit model.

Value

It returns the marginal effects of the estimated SpatialProbit model.

References

J. LeSage and R.K. Pace. *Introduction to Spatial Econometrics*, CRC Press, chapter 10.1.6, 2009.

fitted.SpatialProbit *Extract spatial probit model fitted values.*

Description

Extract the fitted values of a SpatialProbit model.

Usage

```
## S3 method for class 'SpatialProbit'
fitted(object, type = "link", cut = 0.5, ...)
```

Arguments

object	an object of class SpatialProbit.
type	the type of prediction: "link" the value of the latent variable. Default. "response" probability. "binary" binary 0/1 output.
cut	the threshold probability for the "binary" type. Default is 0.5.
...	ignored

Value

Returns the vector of fitted values of the SpatialProbit model

generate_W *Generate a random spatial weight matrix.*

Description

Generate a spatial weight matrix of given size and number of nearest neighbors from randomly-located observations on the unit square.

Usage

```
generate_W(n, nneigh, seed=123)
```

Arguments

n	the size of the matrix.
nneigh	the number of nearest neighbors.
seed	an integer to set the seed for the random generated locations.

Details

The output matrix has zero diagonal and it is row-standardised. The n observations are allocated randomly in the unit square. For each observation, the n_{neigh} closest observations w.r.t. the Euclidean distance are assigned with a weight equal to $1/n_{\text{neigh}}$.

Value

a matrix of class `dgCMatrix` (sparse matrix).

See Also

[sim_binomial_probit](#).

Examples

```
W <- generate_W(100,4,seed=12)
image(W)
```

Katrina

New Orleans business recovery in the aftermath of Hurricane Katrina.

Description

This dataset has been used in the LeSage et al. (2011) paper entitled "New Orleans business recovery in the aftermath of Hurricane Katrina" to study the decisions of shop owners to reopen business after Hurricane Katrina. The dataset contains 673 observations on 3 streets in New Orleans and can be used to estimate the spatial probit models and to replicate the findings in the paper.

Usage

```
data(Katrina)
```

Format

`Katrina.raw` is a data frame with 673 observations on the following 15 variables:

`code` a numeric vector
`long` longitude coordinate of store
`lat` latitude coordinate of store
`street1` a numeric vector
`medinc` median income
`perinc` a numeric vector
`elevation` a numeric vector
`flood` flood depth (measured in feet)

owntype type of store ownership: "sole proprietorship" vs. "local chain" vs. "national chain"
 sesstatus socio-economic status of clientele (1-5): 1-2 = low # status customers, 3 = middle, 4-5
 = high status customers
 sizeemp "small size" vs. "medium size" vs. "large size" firms
 openstatus1 a numeric vector
 openstatus2 a numeric vector
 days days to reopen business
 street 1=Magazine Street, 2=Carrollton Avenue, 3=St. Claude Avenue

Katrina is a data frame with 673 observations on the following 13 variables.

long longitude coordinate of store
 lat latitude coordinate of store
 flood_depth flood depth (measured in feet)
 log_medinc log median income
 small_size binary variable for "small size" firms
 large_size binary variable for "large size" firms
 low_status_customers binary variable for low socio-economic status of clientele
 high_status_customers binary variable for high socio-economic status of clientele
 owntype_sole_proprietor a binary variable indicating "sole proprietor" ownership type
 owntype_national_chain a binary variable indicating "national_chain" ownership type
 y1 reopening status in the very short period 0-3 months; 1=reopened, 0=not reopened
 y2 reopening status in the period 0-6 months; 1=reopened, 0=not reopened
 y3 reopening status in the period 0-12 months; 1=reopened, 0=not reopened

Details

The Katrina.raw dataset contains the data found on the website before some of the variables are recoded. For example, the socio-economic status of clientele is coded as 1-5 in the raw data, but only 3 levels will be used in estimation: 1-2 = low status customers, 3 = middle, 4-5 = high status customers. Hence, with "middle" as the reference category, Katrina contains 2 dummy variables for low status customers and high status customers.

The dataset Katrina is the result of these recoding operations and can be directly used for model estimation.

Note

When defining the reopening status variables y1 (0-3 months), y2 (0-6 months), and y3 (0-12 months) from the days variable, the Matlab code ignores the seven cases where days=90. To be consistent with the number of cases in the paper, we define y1,y2,y3 in the same way: y1=sum(days < 90), y2=sum(days < 180 & days != 90), y3=sum(days < 365 & days != 90). So this is not a bug, its a feature.

Source

The raw data was obtained from the Royal Statistical Society dataset website www.blackwellpublishing.com/rss/Volumes/Av174p4.htm and brought to RData format by Wilhelm and Godinho de Matos (2013).

Examples

```
## Not run:
data(Katrina)
attach(Katrina)
table(y1) # 300 of the 673 firms reopened during 0-3 months horizon, p.1016
table(y2) # 425 of the 673 firms reopened during 0-6 months horizon, p.1016
table(y3) # 478 of the 673 firms reopened during 0-12 months horizon, p.1016
detach(Katrina)

# replicate LeSage et al. (2011), Table 3, p.1017
require(spdep)

# (a) 0-3 months time horizon
# LeSage et al. (2011) use k=11 nearest neighbors in this case
nb <- knn2nb(knearneigh(cbind(Katrina$lat, Katrina$long), k=11))
listw <- nb2listw(nb, style="W")
W1 <- as(as_dgRMatrix_listw(listw), "CsparseMatrix")

fit1_cond <- SpatialProbitFit(y1 ~ flood_depth + log_medinc + small_size +
  large_size + low_status_customers + high_status_customers +
  owntype_sole_proprietor + owntype_national_chain,
  W=W1, data=Katrina, DGP='SAR', method="conditional", varcov="varcov")
summary(fit1_cond)

fit1_FL <- SpatialProbitFit(y1 ~ flood_depth + log_medinc + small_size +
  large_size + low_status_customers + high_status_customers +
  owntype_sole_proprietor + owntype_national_chain,
  W=W1, data=Katrina, DGP='SAR', method="full-lik", varcov="varcov")
summary(fit1_FL)

fit1_cond_10nn <- SpatialProbitFit(y1 ~ flood_depth+ log_medinc+ small_size+
  large_size + low_status_customers + high_status_customers +
  owntype_sole_proprietor + owntype_national_chain,
  W=W1, data=Katrina, DGP='SAR', method="conditional", varcov="varcov",
  control=list(iW_CL=10))
summary(fit1_cond_10nn)

# (b) 0-6 months time horizon
# LeSage et al. (2011) use k=15 nearest neighbors
nb <- knn2nb(knearneigh(cbind(Katrina$lat, Katrina$long), k=15))
listw <- nb2listw(nb, style="W")
W2 <- as(as_dgRMatrix_listw(listw), "CsparseMatrix")

fit2_cond <- SpatialProbitFit(y2 ~ flood_depth + log_medinc + small_size +
  large_size + low_status_customers + high_status_customers +
```

```

owntype_sole_proprietor + owntype_national_chain,
W=W2, data=Katrina, DGP='SAR', method="full-lik", varcov="varcov")
summary(fit2_cond)

fit2_FL <- SpatialProbitFit(y2 ~ flood_depth + log_medinc + small_size +
large_size + low_status_customers + high_status_customers +
owntype_sole_proprietor + owntype_national_chain,
W=W2, data=Katrina, DGP='SAR', method="full-lik", varcov="varcov")
summary(fit2_FL)

# (c) 0-12 months time horizon
# LeSage et al. (2011) use k=15 nearest neighbors as in 0-6 months
W3 <- W2
fit3_cond <- SpatialProbitFit(y3 ~ flood_depth + log_medinc + small_size +
large_size + low_status_customers + high_status_customers +
owntype_sole_proprietor + owntype_national_chain,
W=W3, data=Katrina, DGP='SAR', method="conditional", varcov="varcov")
summary(fit3_cond)

fit3_FL <- SpatialProbitFit(y3 ~ flood_depth + log_medinc + small_size +
large_size + low_status_customers + high_status_customers +
owntype_sole_proprietor + owntype_national_chain,
W=W3, data=Katrina, DGP='SAR', method="full-lik", varcov="varcov")
summary(fit3_FL)

# replicate LeSage et al. (2011), Table 4, p.1018
# SAR probit model effects estimates for the 0-3-month time horizon
effects(fit1_cond)

# replicate LeSage et al. (2011), Table 5, p.1019
# SAR probit model effects estimates for the 0-6-month time horizon
effects(fit2_cond)

# replicate LeSage et al. (2011), Table 6, p.1020
# SAR probit model effects estimates for the 0-12-month time horizon
effects(fit3_cond)

## End(Not run)

```

Katrina.raw

New Orleans business recovery in the aftermath of Hurricane Katrina.

Description

This dataset has been used in the LeSage et al. (2011) paper entitled "New Orleans business recovery in the aftermath of Hurricane Katrina" to study the decisions of shop owners to reopen business after Hurricane Katrina. The dataset contains 673 observations on 3 streets in New Orleans and can be used to estimate the spatial probit models and to replicate the findings in the paper.

Usage

```
data(Katrina)
```

Format

Katrina.raw is a data frame with 673 observations on the following 15 variables:

```
code a numeric vector
long longitude coordinate of store
lat latitude coordinate of store
street1 a numeric vector
medinc median income
perinc a numeric vector
elevation a numeric vector
flood flood depth (measured in feet)
owntype type of store ownership: "sole proprietorship" vs. "local chain" vs. "national chain"
sesstatus socio-economic status of clientele (1-5): 1-2 = low # status customers, 3 = middle, 4-5
      = high status customers
sizeemp "small size" vs. "medium size" vs. "large size" firms
openstatus1 a numeric vector
openstatus2 a numeric vector
days days to reopen business
street 1=Magazine Street, 2=Carrollton Avenue, 3=St. Claude Avenue
```

Katrina is a data frame with 673 observations on the following 13 variables.

```
long longitude coordinate of store
lat latitude coordinate of store
flood_depth flood depth (measured in feet)
log_medinc log median income
small_size binary variable for "small size" firms
large_size binary variable for "large size" firms
low_status_customers binary variable for low socio-economic status of clientele
high_status_customers binary variable for high socio-economic status of clientele
owntype_sole_proprietor a binary variable indicating "sole proprietor" ownership type
owntype_national_chain a binary variable indicating "national_chain" ownership type
y1 reopening status in the very short period 0-3 months; 1=reopened, 0=not reopened
y2 reopening status in the period 0-6 months; 1=reopened, 0=not reopened
y3 reopening status in the period 0-12 months; 1=reopened, 0=not reopened
```

Details

The Katrina.raw dataset contains the data found on the website before some of the variables are recoded. For example, the socio-economic status of clientele is coded as 1-5 in the raw data, but only 3 levels will be used in estimation: 1-2 = low status customers, 3 = middle, 4-5 = high status customers. Hence, with "middle" as the reference category, Katrina contains 2 dummy variables for low status customers and high status customers.

The dataset Katrina is the result of these recoding operations and can be directly used for model estimation.

Note

When defining the reopening status variables y1 (0-3 months), y2 (0-6 months), and y3 (0-12 months) from the days variable, the Matlab code ignores the seven cases where days=90. To be consistent with the number of cases in the paper, we define y1,y2,y3 in the same way: y1=sum(days < 90), y2=sum(days < 180 & days != 90), y3=sum(days < 365 & days != 90). So this is not a bug, its a feature.

Source

The raw data was obtained from the Royal Statistical Society dataset website www.blackwellpublishing.com/rss/Volumes/Av174p4.htm and brought to RData format by Wilhelm and Godinho de Matos (2013).

names.SpatialProbit *Extract names of SpatialProbit class.*

Description

Extract names of SpatialProbit class.

Usage

```
## S3 method for class 'SpatialProbit'  
names(x, ...)
```

Arguments

x an object of class SpatialProbit.
... ignored

Value

Returns the names of the SpatialProbit object.

```
predict.SpatialProbit Spatial probit model predictions.
```

Description

Predicts of a SpatialProbit model on a set X of covariates

Usage

```
## S3 method for class 'SpatialProbit'
predict(object, X, type = "link", cut = 0.5, ...)
```

Arguments

object	an object of class SpatialProbit.
X	a martix of explanatory variables.
type	the type of prediction: "link" the value of the latent variable. Default "response" probability. "binary" binary 0/1 output.
cut	the threshold probability for the "binary" type. Default is 0.5.
...	ignored

Value

Returns a vector of predicted values for the set X of covariates

```
residuals.SpatialProbit  
Extract spatial probit model residuals.
```

Description

Compute the residuals of an estimated SpatialProbit model.

Usage

```
## S3 method for class 'SpatialProbit'
residuals(object, ...)
```

Arguments

object	an object of class SpatialProbit.
...	ignored

Value

Return a vector containing the generalised residuals of the SpatialProbit model.

sim_binomial_probit *Simulate the dependent variable of a SAR/SEM/SARAR model.*

Description

The function `sim_binomial_probit` is used to generate the dependent variable of a spatial binomial probit model, where all the data and parameters of the model can be modified by the user.

Usage

```
sim_binomial_probit(W,X,beta,rho,model="SAR",M=NULL,lambda=NULL,
sigma2=1,ord_iW=6,seed=123)
```

Arguments

W	the spatial weight matrix (works for "SAR" and "SEM" models).
X	the matrix of covariates.
beta	the value of the covariates parameters.
rho	the value of the spatial dependence parameter (works for "SAR" and "SEM" models).
model	the type of model, between "SAR", "SEM", "SARAR" (Default is "SAR").
M	the second spatial weight matrix (only if model is "SARAR").
lambda	the value of the spatial dependence parameter (only if model is "SARAR").
sigma2	the variance of the error term (Default is 1).
ord_iW	the order of approximation of the matrix $(I_n - \rho W)^{-1}$.
seed	to set the random generator seed of the error term.

Details

The `sim_binomial_probit` generates a vector of dependent variables for a spatial probit model. It allows to simulate the following DGPs (Data Generating Process): SAR

$$z = (I_n - \rho W)^{-1}(X\beta + \epsilon)$$

SEM

$$z = (X\beta + (I_n - \rho W)^{-1}\epsilon)$$

SARAR

$$z = (I_n - \rho W)^{-1}(X\beta + (I_n - \lambda M)^{-1}\epsilon)$$

where ϵ are independent and normally distributed with mean zero and variance `sigma2` (default is 1).

The matrix X of covariates, the corresponding parameters β , the spatial weight matrix W and the corresponding spatial dependence parameter ρ need to be passed by the user. The matrix $(I_n - \rho W)^{-1}$ is computed using the `ApproxIW` function, that can either invert $(I_n - \rho W)$ exactly, if `order_iW=0` (not suitable for n bigger than 1000), or using the Taylor approximation

$$(I_n - \rho W)^{-1} = I_n + \rho W + \rho^2 W^2 + \dots$$

of order `order_iW` (default is approximation of order 6).

Value

a vector of zeros and ones

See Also

[generate_W](#), [SpatialProbitFit](#).

Examples

```
n <- 1000
nneigh <- 3
rho <- 0.5
beta <- c(4,-2,1)
W <- generate_W(n,nneigh)
X <- cbind(1,rnorm(n,2,2),rnorm(n,0,1))
y <- sim_binomial_probit(W,X,beta,rho,model="SAR") #SAR model
y <- sim_binomial_probit(W,X,beta,rho,model="SEM") #SEM model
M <- generate_W(n,nneigh,seed=1)
lambda <- -0.5
y <- sim_binomial_probit(W,X,beta,rho,model="SARAR",M=M,lambda=lambda) #SARAR
```

SpatialProbit-class *Class of Spatial Probit Model.*

Description

Class of Spatial Probit Model.

Slots

`beta` numeric, the estimated parameters for the covariates.
`rho` numeric, the estimated spatial dependence parameter.
`coeff` numeric, all estimated parameters.
`loglik` numeric, the likelihood associated to the estimated model.
`formula` formula.
`nobs` numeric, number of observations.

nvar numeric, number of covariates.
 y numeric, vector of observed dependent variable.
 X matrix, matrix of covariates.
 time numeric, estimation time.
 DGP character, DGP of the model (SAR or SEM).
 method character, estimation method ("conditional" or "full-lik").
 varcov character, indicates the matrix used in the algorithm ("varcov" or "precision").
 W SparseMatrix, the spatial weight matrix.
 iW_CL numeric, the order of approximation used in the conditional method.
 iW_FL numeric, the order of approximation used inside the likelihood function for the full-lik method.
 iW_FG numeric, the order of approximation used inside the gradient functions for the full-lik method.
 reltol numeric, the relative convergence tolerance.
 prune numeric, the pruning for the gradient functions.
 env an environment containing information for use in later function calls to save time.
 message a integer giving any additional information or NULL.

SpatialProbitFit *Fit a spatial probit model.*

Description

Approximate likelihood estimation of the spatial autoregressive probit model (SAR) or spatial error probit model (SEM).

Usage

```
SpatialProbitFit(formula,data,W,
                 DGP='SAR',method="conditional",varcov="varcov",control=list())
```

Arguments

formula	an object of class formula: a symbolic description of the model to be fitted.
data	the data set containing the variables of the model.
W	the spatial weight matrix of class "dgCMatrix".
DGP	the data generating process of data: SAR or SEM (Default is SAR).
method	the optimisation method: "conditional" or "full-lik" (Default is "conditional", see Details).
varcov	the likelihood function is computed using the variance-covariance matrix ("varcov") or the precision matrix ("precision")? Default is "varcov".
control	a list of control parameters. See Details.

Details

The estimation is based on the approximate value of the true likelihood of spatial autoregressive (SAR) or spatial error (SEM) probit models. The DGP of the spatial autoregressive model (SAR) model is the following

$$y = (I_n - \rho W)^{-1}(X\beta + \epsilon),$$

where the disturbances ϵ are iid standard normally distributed, W is a sparse spatial weight matrix and ρ is the spatial lag parameter. The variance of the error term is equal to $\Sigma = \sigma^2((I_n - \rho W)^{-1}((I_n - \rho W)^{-1})^t)$. The DGP of the spatial error model (SEM) is as follows

$$y = X\beta + (I_n - \rho W)^{-1}\epsilon,$$

where the disturbances ϵ are iid standard normally distributed, W is a sparse spatial weight matrix and ρ is the spatial error parameter. The variance of the error term is equal to $\Sigma = \sigma^2((I_n - \rho W)^{-1}((I_n - \rho W)^{-1})^t)$.

The approximation is inspired by the Mendell-Elston approximation of the multivariate normal probabilities (see References). It makes use of the Cholesky decomposition of the variance-covariance matrix Σ .

The `SpatialProbitFit` command estimates the model by maximising the approximate log-likelihood. We propose two optimisation method:

"conditional": it relies on a standard probit estimation (we use `speedglm`) which applies to the model estimated conditional on ρ .

"full-lik": it minimises the full-log-likelihood using the analytical gradient functions. The optimisation is performed by means of the `optim` function with `method = "BFGS"`.

In both cases a "conditional" estimation is performed. If `method="conditional"`, then `SpatialProbitFit` returns the results of this first estimation. In case `method="full-lik"`, the function tries to improve the log-likelihood by means of a further exploration around the value of the parameters found by the conditional step. The conditional step is usually very accurate and particularly fast. The second step is more time consuming and does not always improve the results of the first step. We dissuade the user from using the full-likelihood method for sample sizes bigger than ten thousands, since the computation of the gradients is quite slow. Simulation studies reported in Martinetti and Geniaux (2015) prove that the conditional estimation is highly reliable, even if compared to the full-likelihood ones.

In order to reduce the computation time of the function `SpatialProbitFit`, we propose a variant of the likelihood-function estimation that uses the inverse of the variance-covariance matrix (a.k.a. precision matrix). This variant applies to both the "conditional" and the "full-lik" methods and can be invoked by setting `varcov="precision"`. Simulation studies reported in Martinetti and Geniaux (2015) suggest that the accuracy of the results with the precision matrix are sometimes worst than the one with the true variance-covariance matrix, but the estimation time is considerably reduced.

The control argument is a list that can supply any of the following components:

`iW_CL` the order of approximation of $(I_n - \rho W)^{-1}$ used in the "conditional" method. Default is 6, while 0 means no approximation (it uses exact inversion of matrixes, not suitable for big sample sizes). See Martinetti and Geniaux (2015) for further references.

`iW_FL` the order of approximation of $(I_n - \rho W)^{-1}$ used in the computation of the likelihood function for the "full-lik" method. Default is 0, meaning no approximation.

- iW_FG** the order of approximation of $(I_n - \rho W)^{-1}$ used in the computation of the gradient functions for the "full-lik" method. Default is 0, meaning no approximation.
- reltol** relative convergence tolerance. It represents `tol` in `optimize` function for `method="conditional"` and `reltol` in `optim` function for `method="full-lik"`. Default is 1e-5.
- prune** the pruning value used in the gradients. Default is 0, meaning no pruning. Typical values are around 1e-3 and 1e-6. They help reducing the estimation time of the gradient functions.

Value

Return a structure of class `SpatialProbit`:

- beta** the estimated parameters for the covariates
- rho** the estimated spatial dependence parameter
- coeff** all estimated parameters
- loglik** the log-likelihood associated to the estimated model
- formula** same as `formula`
- nobs** number of observations
- nvar** number of covariates or explanatory variables
- y** the vector of the dependent variable
- X** the matrix of covariates or explanatory variables
- time** estimation time
- DGP** the chosen DGP (SAR or SEM)
- method** the estimation method (see Details)
- varcov** the matrix used in the approximation (see Details)
- W** the spatial weight matrix
- iW_CL** the order of approximation used in the conditional method
- iW_FL** the order of approximation used in the likelihood function for the full-lik method
- iW_FG** the order of approximation used in the gradient functions for the full-lik method
- reltol** the relative convergence tolerance
- prune** the pruning used in the gradient functions
- env** an environment containing information for use in later function calls to save time
- message** a integer giving any additional information or NULL.

Examples

```
n <- 1000
nneigh <- 3
rho <- 0.5
beta <- c(4,-2,1)
W <- generate_W(n,nneigh,seed=123)
X <- cbind(1,rnorm(n,2,2),rnorm(n,0,1))
colnames(X) <- c("intercept","X1","X2")
y <- sim_binomial_probit(W=W,X=X,beta=beta,rho=rho,model="SAR")
```

```
d <- as.data.frame(cbind(y,X))
mod <- SpatialProbitFit(y~X1+X2,d,W,
  DGP='SAR',method="conditional",varcov="varcov")
```

summary.SpatialProbit *Spatial probit model summaries.*

Description

Print the results of a SpatialProbit model.

Usage

```
## S3 method for class 'SpatialProbit'
summary(object, covar = FALSE, ...)
```

Arguments

object	an object of class SpatialProbit.
covar	should the statistics be computed with the matrix of variance of the parameters or not. Default is FALSE, hence Likelihood-ratio statistics are printed.
...	further arguments

Details

The summary function prints

Model Features on the model and dataset.

Time Estimation time.

Statistics Standard errors of the estimated parameters. If covar=TRUE, it uses the matrix of variance of the parameters, else the likelihood ratio test.

Accuracy Confusion Matrix and accuracy of the estimated model.

Value

This function does not return any value.

Index

*Topic **datasets**

Katrina, [8](#)

Katrina.raw, [11](#)

coef.SpatialProbit, [3](#)

conditional_SAR_UC, [4](#)

conditional_SAR_UP, [4](#)

conditional_SEM_UC, [5](#)

conditional_SEM_UP, [5](#)

effects, [6](#)

fitted.SpatialProbit, [7](#)

generate_W, [7](#), [16](#)

Katrina, [2](#), [8](#)

Katrina.raw, [11](#)

McSpatial, [2](#)

names.SpatialProbit, [13](#)

optim, [18](#), [19](#)

optimize, [19](#)

predict.SpatialProbit, [14](#)

ProbitSpatial (ProbitSpatial-package), [2](#)

ProbitSpatial-package, [2](#)

residuals.SpatialProbit, [14](#)

sim_binomial_probit, [8](#), [15](#)

SpatialProbit (SpatialProbit-class), [16](#)

SpatialProbit-class, [16](#)

SpatialProbitFit, [16](#), [17](#)

speedglm, [18](#)

summary.SpatialProbit, [20](#)