Package ‘NBtsVarSel’

July 17, 2023

Type Package

Title Variable Selection in a Specific Regression Time Series of Counts

Version 1.0

Date 2023-07-17

Description Performs variable selection in sparse negative binomial GLARMA (Generalised Linear Autoregressive Moving Average) models. For further details we refer the reader to the paper Gomtsyan (2023), <arXiv:2307.00929>.

License GPL-2

Depends R (>= 3.5.0), Matrix, glmnet, stats, MASS, mpath, ggplot2

VignetteBuilder knitr

Suggests knitr, markdown, formatR

NeedsCompilation no

Author Marina Gomtsyan [aut, cre]

Maintainer Marina Gomtsyan <mgomtsian@gmail.com>

Repository CRAN

Date/Publication 2023-07-17 17:40:02 UTC

R topics documented:

NBtsVarSel-package ........................................... 2
grad_hess_beta ........................................... 3
grad_hess_gamma .......................................... 4
NR_gamma .................................................. 5
variable_selection .......................................... 6
Y ............................................................. 7

Index 8
NBtsVarSel-package

Variable Selection in a Specific Regression Time Series of Counts

Description

NBtsVarSel consists of four functions: "variable_selection.R", "grad_hess_beta.R", "grad_hess_gamma.R" and "NR_gamma.R". For further information on how to use these functions, we refer the reader to the vignette of the package.

Details

This package consists of four functions: "variable_selection.R", "grad_hess_beta.R", "grad_hess_gamma.R" and "NR_gamma.R". For further information on how to use these functions, we refer the reader to the vignette of the package.

Author(s)

Marina Gomtsyan

Maintainer: Marina Gomtsyan <mgomtsian@gmail.com>

References

M. Gomtsyan "Variable selection in a specific regression time series of counts.", arXiv:2307.00929

Examples

```r
n = 50
p = 30
X = matrix(NA,(p+1),n)
f = 1/0.7
for(t in 1:n){X[,t] = c(1,cos(2*pi*(1:(p/2))*t*f/n),sin(2*pi*(1:(p/2))*t*f/n))}
gamma0 = c(0)
data(Y)
result = variable_selection(Y, X, gamma.init=gamma0, alpha.init=NULL, k.max=1, method="cv", tr=0.3, n.iter=100, n.rep=1000)
beta_est = result$beta_est
Estim_active = result$estim_active
gamma_est = result$gamma_est
alpha_est = result$alpha_est
```
Grad and Hess of the log-likelihood with respect to beta

Description

This function calculates the gradient and Hessian of the log-likelihood with respect to beta.

Usage

grad_hess_beta(Y, X, beta, gamma, alpha)

Arguments

Y: Observation matrix
X: Design matrix
beta: Initial beta vector
gamma: Initial gamma vector
alpha: Initial overdispersion parameter

Value

grad_L_beta: Vector of the gradient of L with respect to beta
hess_L_beta: Matrix of the Hessian of L with respect to beta

Author(s)

Marina Gomtsyan
Maintainer: Marina Gomtsyan <mgomtsian@gmail.com>

References

M. Gomtsyan "Variable selection in a specific regression time series of counts.", arXiv:2307.00929

Examples

n = 50
p = 30
X = matrix(NA,(p+1),n)
f = 1/0.7
for(t in 1:n){X[,t] = c(1,cos(2*pi*(1:(p/2))*t*f/n),sin(2*pi*(1:(p/2))*t*f/n))}
gamma0 = c(0)
data(Y)
glm_nb = glm.nb(Y~t(X)[,2:(p+1)])
beta0 = as.numeric(glm_nb$coefficients)
alpha0 = glm_nb$theta
result = grad_hess_beta(Y, X, beta0, gamma0, alpha0)
grad = result$grad_L_beta
Hessian = result$hess_L_beta
Grad_hess_gamma

**Description**

This function calculates the gradient and Hessian of the log-likelihood with respect to gamma

**Usage**

`grad_hess_gamma(Y, X, beta, gamma, alpha)`

**Arguments**

- **Y**: Observation matrix
- **X**: Design matrix
- **beta**: Initial beta vector
- **gamma**: Initial gamma vector
- **alpha**: Initial overdispersion parameter

**Value**

- **grad_L_gamma**: Vector of the gradient of L with respect to gamma
- **hess_L_gamma**: Matrix of the Hessian of L with respect to gamma

**Author(s)**

Marina Gomtsyan

Maintainer: Marina Gomtsyan <mgomtsian@gmail.com>

**References**

M. Gomtsyan "Variable selection in a specific regression time series of counts.", arXiv:2307.00929

**Examples**

```r
n = 50
p = 30
X = matrix(NA,(p+1),n)
f = 1/0.7
for(t in 1:n){X[,t] = c(1,cos(2*pi*(1:(p/2))*t*f/n),sin(2*pi*(1:(p/2))*t*f/n))}
gamma0 = c(0)
data(Y)
glm_nb = glm.nb(Y~t(X[,2:(p+1)]))
beta0 = as.numeric(glm_nb$coefficients)
alpha0 = glm_nb$theta
result = grad_hess_gamma(Y, X, beta0, gamma0, alpha0)
grad = result$grad_L_gamma
Hessian = result$hess_L_gamma
```
NR_gamma

Newton-Raphson method for estimation of gamma

Description

This function estimates gamma with Newton-Raphson method.

Usage

NR_gamma(Y, X, beta, gamma, alpha, n.iter)

Arguments

Y Observation matrix
X Design matrix
beta Initial beta vector
gamma Initial gamma vector
alpha Initial overdispertion parameter
n.iter Number of iterations of the algorithm. Default=100

Value

gamma Estimated gamma vector

Author(s)

Marina Gomtsyan
Maintainer: Marina Gomtsyan <mgomtsian@gmail.com>

References

M. Gomtsyan "Variable selection in a specific regression time series of counts.". arXiv:2307.00929

Examples

n = 50
p = 30
X = matrix(NA,(p+1),n)
f = 1/0.7
for(t in 1:n){X[,t] = c(1,cos(2*pi*(1:(p/2))*t*f/n),sin(2*pi*(1:(p/2))*t*f/n))}
gamma0 = c(0)
data(Y)
glm_nb = glm.nb(Y~t(X[,2:(p+1)])
beta0 = as.numeric(glm_nb$coefficients)
alpha0 = glm_nb$theta
gamma_est = NR_gamma(Y, X, beta0, gamma0, alpha0, n.iter=100)
variable_selection       Variable selection

Description
This function performs variable selection, estimates new vectors of beta and gamma and a new alpha.

Usage
variable_selection(Y, X, gamma.init, alpha.init = NULL, k.max = 1, method = "cv", tr = 0.3, n.iter = 100, n.rep = 1000)

Arguments
Y      Observation matrix
X      Design matrix
gamma.init    Initial gamma vector
alpha.init    Optional initial alpha value. The default is NULL
k.max    Number of iteration to repeat the whole algorithm
method    Stability selection method: "min" or "cv". In "min" the smallest lambda is chosen, in "cv" cross-validation lambda is chosen for stability selection. The default is "cv"
tr    Threshold for stability selection. The default is 0.3
n.iter    Number of iteration for Newton-Raphson algorithm. The default is 100
n.rep    Number of replications in stability selection step. The default is 1000

Value
estim_active    Estimated active coefficients
beta_est    Vector of estimated beta values
gamma_est    Vector of estimated gamma values
alpha_est    Estimation of alpha

Author(s)
Marina Gomtsyan
Maintainer: Marina Gomtsyan <mgomtsian@gmail.com>

References
M. Gomtsyan "Variable selection in a specific regression time series of counts.", arXiv:2307.00929
Examples

```r
n = 50
p = 30
X = matrix(NA,(p+1),n)
f = 1/0.7
for(t in 1:n){X[,t] = c(1,cos(2*pi*(1:(p/2))*t*f/n),sin(2*pi*(1:(p/2))*t*f/n))}
gamma0 = c(0)
data(Y)
result = variable_selection(Y, X, gamma.init=gamma0, alpha.init=NULL, k.max=1, method="cv", tr=0.3, n.iter=100, n.rep=1000)
beta_est = result$beta_est
Estim_active = result$estim_active
gamma_est = result$gamma_est
alpha_est = result$alpha_est
```

**Y**

*Observation matrix Y*

Description

An example of observation matrix

Usage

```r
data("Y")
```

Format

The format is: num [1:50] 9 2 11 14 18 17 1 0 1 0 ...

References

M. Gomtsyan "Variable selection in a specific regression time series of counts.", arXiv:2307.00929

Examples

```r
data(Y)
```
Index

* package
  NBtsVarSel-package, 2

grad_hess_beta, 3
grad_hess_gamma, 4

NBtsVarSel (NBtsVarSel-package), 2
NBtsVarSel-package, 2
NR_gamma, 5

variable_selection, 6

Y, 7