# Package ‘GlarmaVarSel’

October 12, 2022

**Type** Package

**Title** Variable Selection in Sparse GLARMA Models

**Version** 1.0

**Date** 2021-09-15

**Author** M. Gomtsyan, C. Levy-Leduc, S. Ouadah, L. Sansonnet

**Maintainer** Marina Gomtsyan <marina.gomtsyan@agroparistech.fr>

**Description** Performs variable selection in high-dimensional sparse GLARMA models. For further details we refer the reader to the paper Gomtsyan et al. (2020), <arXiv:2007.08623v1>.

**License** GPL-2

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

**VignetteBuilder** knitr

**Suggests** knitr, markdown, formatR, doMC

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2021-09-16 07:40:02 UTC

## R topics documented:

## GlarmaVarSel-package *Variable Selection in Sparse GLARMA Models*

### Description

GlarmaVarSel 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

GlarmaVarSel 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, Celine Levy-Leduc, Sarah Ouadah, Laure Sansonnet

Maintainer: Marina Gomtsyan <marina.gomtsyan@agroparistech.fr>

### References

M. Gomtsyan et al. "Variable selection in sparse GLARMA models", arXiv:2007.08623v1

### 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)
result = variable_selection(Y, X, gamma0, k_max=2, n_iter=100, method="min",
nb_rep_ss=1000, threshold=0.7, parallel=FALSE, nb.cores=1)
beta_est = result$beta_est
Estim_active = result$estim_active
gamma_est = result$gamma_est
```

## grad_hess_beta — *Gradient and Hessian 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, beta0, gamma0)
```

### Arguments

| | |
|---|---|
| Y | Observation matrix |
| X | Design matrix |
| beta0 | Initial beta vector |
| gamma0 | Initial gamma vector |

### 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, Celine Levy-Leduc, Sarah Ouadah, Laure Sansonnet

Maintainer: Marina Gomtsyan <marina.gomtsyan@agroparistech.fr>

### References

M. Gomtsyan et al. "Variable selection in sparse GLARMA models", arXiv:2007.08623v1

### 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_pois<-glm(Y~t(X)[,2:(p+1)],family = poisson)
beta0<-as.numeric(glm_pois$coefficients)
result = grad_hess_beta(Y, X, beta0, gamma0)
grad = result$grad_L_beta
Hessian = result$hess_L_beta
```

grad_hess_gamma *Gradient and Hessian of the log-likelihood with respect to gamma*

## Description

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

## Usage

```
grad_hess_gamma(Y, X, beta0, gamma0)
```

## Arguments

| | |
|---|---|
| `Y` | Observation matrix |
| `X` | Design matrix |
| `beta0` | Initial beta vector |
| `gamma0` | Initial gamma vector |

## 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, Celine Levy-Leduc, Sarah Ouadah, Laure Sansonnet

Maintainer: Marina Gomtsyan <marina.gomtsyan@agroparistech.fr>

## References

M. Gomtsyan et al. "Variable selection in sparse GLARMA models", arXiv:2007.08623v1

## 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_pois<-glm(Y~t(X)[,2:(p+1)],family = poisson)
beta0<-as.numeric(glm_pois$coefficients)
result = grad_hess_gamma(Y, X, beta0, gamma0)
grad = result$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, beta0, gamma0, n_iter)
```

### Arguments

| | |
|---|---|
| Y | Observation matrix |
| X | Design matrix |
| beta0 | Initial beta vector |
| gamma0 | Initial gamma vector |
| n_iter | Number of iterations of the algorithm. Default=100 |

### Value

| | |
|---|---|
| gamma | Estimated gamma vector |

### Author(s)

Marina Gomtsyan, Celine Levy-Leduc, Sarah Ouadah, Laure Sansonnet

Maintainer: Marina Gomtsyan <marina.gomtsyan@agroparistech.fr>

### References

M. Gomtsyan et al. "Variable selection in sparse GLARMA models", arXiv:2007.08623v1

### 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_pois<-glm(Y~t(X)[,2:(p+1)],family = poisson)
beta0<-as.numeric(glm_pois$coefficients)
gamma_est = NR_gamma(Y, X, beta0, gamma0, n_iter=100)
```

## variable_selection *Variable selection*

### Description

This function performs variable selection, estimates a new vector beta and a new vector gamma

### Usage

```
variable_selection(Y, X, gamma0, k_max = 2, n_iter = 100, method = "min",
  nb_rep_ss = 1000, threshold = 0.8, parallel = FALSE, nb.cores = 1)
```

### Arguments

| | |
|---|---|
| `Y` | Observation matrix |
| `X` | Design matrix |
| `gamma0` | Initial gamma vector |
| `k_max` | Number of iteration to repeat the whole algorithm |
| `n_iter` | Number of iteration for Newton-Raphson algorithm |
| `method` | Stability selection method: "fast", "min" or "cv". In "min" the smallest lambda is chosen, in "cv" cross-validation lambda is chosen for stability selection. "fast" is a fater stability selection approach. The default is "min" |
| `nb_rep_ss` | Number of replications in stability selection step. The default is 1000 |
| `threshold` | Threshold for stability selection. The default is 0.9 |
| `parallel` | Whether to parallelize stability selection step or not. The default is FALSE |
| `nb.cores` | Number of cores for parallelization. The default is 1 |

### Value

| | |
|---|---|
| `estim_active` | Estimated active coefficients |
| `beta_est` | Vector of estimated beta values |
| `gamma_est` | Vector of estimated gamma values |

### Author(s)

Marina Gomtsyan, Celine Levy-Leduc, Sarah Ouadah, Laure Sansonnet

Maintainer: Marina Gomtsyan <marina.gomtsyan@agroparistech.fr>

### References

M. Gomtsyan et al. "Variable selection in sparse GLARMA models", arXiv:2007.08623v1

### 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)
result = variable_selection(Y, X, gamma0, k_max=2, n_iter=100, method="min",
nb_rep_ss=1000, threshold=0.7, parallel=FALSE, nb.cores=1)
beta_est = result$beta_est
Estim_active = result$estim_active
gamma_est = result$gamma_est
```

---

## Y *Observation matrix Y*

---

### Description

An example of observation matrix

### Usage

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

### Format

The format is: num [1:50] 11 8 3 3 3 4 4 4 3 1 ...

### References

M. Gomtsyan et al. "Variable selection in sparse GLARMA models", arXiv:2007.08623v1

### Examples

```
data(Y)
```

# Index