

Package ‘FastStepGraph’

May 7, 2026

Type Package

Title A Fast Algorithm for Sparse Precision Matrix Estimation

Version 0.1.1

Maintainer Juan G. Colonna <juancolonna@icomp.ufam.edu.br>

Description It implements an improved and computationally faster version of the original Stepwise Gaussian Graphical Algorithm for estimating the Omega precision matrix from high-dimensional data. Zamar, R., Ruiz, M., Lafit, G. and Nogales, J. (2021) <[doi:10.52933/jdssv.v1i2.11](https://doi.org/10.52933/jdssv.v1i2.11)>.

License MIT + file LICENSE

URL <https://github.com/juancolonna/FastStepGraph>

Depends R (>= 4.3),

Imports doParallel (>= 1.0), foreach (>= 1.5), MASS (>= 7.3)

Suggests knitr, rmarkdown, devtools

VignetteBuilder knitr

Encoding UTF-8

Language en-US

RoxygenNote 7.2.3

NeedsCompilation no

Author Juan G. Colonna [cre, aut] (ORCID: <<https://orcid.org/0000-0002-1740-2618>>),
Marcelo Ruiz [aut]

Repository CRAN

Date/Publication 2023-10-12 16:10:02 UTC

Contents

cv.FastStepGraph	2
FastStepGraph	3
SigmaAR	5

Index	6
--------------	----------

cv.FastStepGraph	<i>Searches for the optimal combination of alpha_f and alpha_b parameters using Cross-Validation</i>
------------------	--

Description

cv.FastStepGraph implements the cross-validation for the Fast Step Graph algorithm.

Usage

```
cv.FastStepGraph(
    x,
    n_folds = 5,
    alpha_f_min = 0.2,
    alpha_f_max = 0.8,
    b_coef = 0.5,
    n_alpha = 32,
    nei_max = 5,
    data_scale = FALSE,
    data_shuffle = TRUE,
    max.iterations = NULL,
    return_model = FALSE,
    parallel = FALSE,
    n_cores = NULL
)
```

Arguments

x	Data matrix (of size n x p).
n_folds	Number of folds for the cross-validation procedure (default value 5).
alpha_f_min	Minimum threshold value for the cross-validation procedure (default value 0.2).
alpha_f_max	Minimum threshold value for the cross-validation procedure (default value 0.8).
b_coef	This parameter applies the empirical rule $\alpha_b = b_coef * \alpha_f$ during the initial search for the optimal α_f parameter while α_b remains fixed, after finding optimal α_f , α_b is varied to find its optimal value. The default value of b_coef is 0.5.
n_alpha	Number of elements in the grid for the cross-validation (default value 32).
nei_max	Maximum number of variables in every neighborhood (default value 5).
data_scale	Boolean parameter (TRUE or FALSE), when to scale data to zero mean and unit variance (default FALSE).
data_shuffle	Boolean parameter (default TRUE), when samples (rows of X) must be randomly shuffled.
max.iterations	Maximum number of iterations (integer), the defaults values is set to $p*(p-1)$.

return_model	Default FALSE. If set to TRUE, at the end of cross-validation, FastStepGraph is called with the optimal parameters alpha_f and alpha_b, returning vareps, beta, Edges and Omega.
parallel	Boolean parameter (TRUE or FALSE), when to run Cross-Validation in parallel using a multicore architecture (default FALSE).
n_cores	An 'int' value specifying the number of cores do you want to use if 'parallel=TRUE'. If n_cores is not specified, the maximum number of cores on your machine minus one will be set automatically.

Value

A list with the values:

alpha_f_opt	the optimal alpha_f value.
alpha_b_opt	the optimal alpha_b value.
CV.loss	minimum loss.

If return_model=TRUE, then also returns:

vareps	Response variables.
beta	Regression coefficients.
Edges	Estimated set of edges.
Omega	Estimated precision matrix.

Author(s)

Prof. Juan G. Colonna, PhD. <juancolonna@icomp.ufam.edu.br>

Prof. Marcelo Ruiz, PhD. <mruiz@exa.unrc.edu.ar>

Examples

```
data <- FastStepGraph::SigmaAR(30, 50, 0.4) # Simulate Gaussian Data
res <- FastStepGraph::cv.FastStepGraph(data$X, data_scale=TRUE)
```

FastStepGraph

Fast Stepwise Gaussian Graphical Model

Description

Improved and faster implementation of the Stepwise Gaussian Graphical Algorithm.

Usage

```
FastStepGraph(
  x,
  alpha_f,
  alpha_b = NULL,
  nei.max = 5,
  data_scale = FALSE,
  max.iterations = NULL
)
```

Arguments

<code>x</code>	Data matrix (of size <code>n_samples</code> x <code>p_variables</code>).
<code>alpha_f</code>	Forward threshold (no default value).
<code>alpha_b</code>	Backward threshold. If <code>alpha_b=NULL</code> , then the rule <code>alpha_b <- 0.5*alpha_f</code> is applied.
<code>nei.max</code>	Maximum number of variables in every neighborhood (default value 5).
<code>data_scale</code>	Boolean parameter (TRUE or FALSE), when to scale data to zero mean and unit variance (default FALSE).
<code>max.iterations</code>	Maximum number of iterations (integer), the defaults values is set to $p*(p-1)$.

Value

A list with the values:

<code>vareps</code>	Response variables.
<code>beta</code>	Regression coefficients.
<code>Edges</code>	Estimated set of edges.
<code>Omega</code>	Estimated precision matrix.

Author(s)

Prof. Juan G. Colonna, PhD. <juancolonna@icomp.ufam.edu.br>

Prof. Marcelo Ruiz, PhD. <mruiz@exa.unrc.edu.ar>

Examples

```
data <- FastStepGraph::SigmaAR(30, 50, 0.4) # Simulate Gaussian Data
G <- FastStepGraph::FastStepGraph(data$X, alpha_f = 0.22, alpha_b = 0.14, data_scale=TRUE)
```

`SigmaAR`*Simulate Covariance Matrix with an Auto-regressive (AR) Model*

Description

Helper function to simulate Simulate Gaussian Data with an Autoregressive (AR) Model

Usage

```
SigmaAR(n_rows, p_columns, phi)
```

Arguments

<code>n_rows</code>	Number of samples (rows of X).
<code>p_columns</code>	Number of variables (columns of X).
<code>phi</code>	Auto-regression coefficient.

Value

A list with the values:

<code>Sigma</code>	A covariance matrix.
<code>Omega</code>	A precision matrix.
<code>X</code>	A normalized data matrix with Gaussian distribution.

Author(s)

Prof. Juan G. Colonna, PhD. <juancolonna@icomp.ufam.edu.br>

Prof. Marcelo Ruiz, PhD. <mruiz@exa.unrc.edu.ar>

Examples

```
data <- FastStepGraph::SigmaAR(30, 50, 0.4) # Simulate Gaussian Data
```

Index

`cv.FastStepGraph`, [2](#)

`FastStepGraph`, [3](#)

`SigmaAR`, [5](#)