Package ‘BDgraph’

March 10, 2015

Type Package
Title Graph Estimation Based on Birth-Death MCMC Approach
Version 2.17
Date 2013-02-04
Author Abdolreza Mohammadi and Ernst Wit
Depends R (>= 3.0.0), Matrix, igraph
Maintainer Abdolreza Mohammadi <a.mohammadi@rug.nl>
Description A general framework to perform Bayesian structure learning in undirected graphical models. The main target is high-dimensional data analysis wherein either continuous or discrete variables.
Acknowledgements The help of the CRAN team to integrate C++ code with R is gratefully acknowledged.
License GPL (>= 3)
LazyLoad yes
URL http://www.math.rug.nl/stat/Main/Research
Repository CRAN
Date/Publication 2015-03-10 06:57:54
NeedsCompilation yes

R topics documented:

BDgraph-package .................................................. 2
bdgraph .............................................................. 3
bdgraph.npn .......................................................... 5
bdgraph.sim .......................................................... 6
CellSignal ........................................................... 8
compare ............................................................... 9
geneExpression ..................................................... 11
phat ................................................................. 11
plot.bdgraph ........................................................ 12
plot.simulate ....................................................... 14
**BDgraph-package**

plotcoda ................................................................. 15
plotroc ................................................................. 16
print.bdgraph .......................................................... 17
print.simulate ......................................................... 19
prob ................................................................. 20
rgwish ................................................................. 21
rwish ................................................................. 22
select ................................................................. 24
summary.bdgraph ....................................................... 25
surveyData ............................................................ 26
traceplot ............................................................. 27

**Index**

**BDgraph-package**

*Graph selection based on birth-death MCMC*

**Description**

The R package **BDgraph** provides statistical tools for Bayesian structure learning in undirected graphical models based on birth-death MCMC methodology. It implements the recent improvements in the Bayesian literature, including Mohammadi and Wit (2014) and Mohammadi et al. (2015).

**Details**

The package includes 10 main functions:

**Functions:**

- bdgraph: birth-death MCMC sampling algorithm for graphical models
- bdgraph.sim: Synthetic graph data generator
- bdgraph.npt: Nonparametric transfer
- compare: Comparing the result
- phat: Posterior link probabilities
- plotcoda: Convergence plot
- plotroc: ROC plot
- rgwish: Sampling from G-Wishart distribution
- select: Selecting the best graph
- traceplot: Trace plot of graph size

**Author(s)**

Abdolreza Mohammadi <a.mohammadi@rug.nl> and Ernst Wit
### References


### bdgraph

**Birth-death MCMC sampling algorithm for graphical models**

### Description

The main function of the **BDgraph** package. The function consists of two sampling algorithm for model determination in undirected graphical models based on birth-death MCMC methodology.

### Usage

```r
bdgraph( data, n = NULL, method = "ggm", iter = 5000, burnin = iter / 2,
        b = 3, D = NULL, Gstart = "empty" )
```

### Arguments

- **data**: It could be a $(n \times p)$ matrix or a data.frame of data or a covariance matrix as $S = X'X$ which $X$ is the data matrix. It also could be an object of class "simulate", from function bdgraph.sim.
- **n**: The number of observations. It is needed if the "data" is a covariance matrix.
- **method**: A character with two options "ggm" (default) and "gcgm". Option "ggm" is for Gaussian graphical models based on gaussianity assumiton. Option "gcgm" is for Gaussian copula graphical models for the data that not follow gaussianity assumiton (e.g. continuous non-Gaussian, discrete, or mixed dataset).
- **iter**: The number of iteration for the sampling algorithm.
- **burnin**: The number of burn-in iteration for the sampling algorithm.
- **b**: The degree of freedom for G-Wishart distribution, $W_G(b, D)$, which is a prior distribution of precision matrix. The default is 3.
- **D**: The positive definite matrix for G-Wishart distribution, $W_G(b, D)$. The default is an identity matrix.
- **Gstart**: It corresponds to a starting point for graph. It can be "full" (default), "empty", or an object with S3 class "bdgraph". Option "full" means the initial graph is a full graph and "empty" means a empty graph. Gstart also could be an object with S3 class "bdgraph"; With this option we could run the sampling algorithm from last objects of previous run (see examples).
Value

An object with S3 class "bdgraph" is returned:

- **sampleGraphs**: A vector which includes the adjacency matrices for all iteration after burn-in.
- **graphWeights**: A vector which includes the waiting times for all iteration after burn-in.
- **allGraphs**: A vector which includes the adjacency matrices for all iteration (includes burn-in iteration). It is needed for monitoring the convergency of the BD-MCMC algorithm.
- **allWeights**: A vector which includes the waiting times for all iteration (includes burn-in iteration). It is needed for monitoring the convergency of the BD-MCMC algorithm.
- **Khat**: Estimation for precision matrix which is a mean of all samples from precision matrices.

Author(s)

Abdolreza Mohammadi and Ernst Wit

References


See Also

`bdgraphNsim`, `summaryNbdgraph`, and `compare`

Examples

```r
# Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraphNsim( n = 20, p = 6, size = 7, vis = TRUE )

output <- bdgraph( data = data.sim, iter = 1000 )
summary(output)

# To compare our result with true graph
compare( data.sim, output, colnames = c("True graph", "BDgraph") )

output2 <- bdgraph( data = data.sim, iter = 5000, Gstart = output )

compare( data.sim, output, output2, colnames = c("True graph", "First run", "Second run") )
```
# generating mixed data from a 'scale-free' graph

data.sim <- bdgraph.sim( n = 100, p = 6, type = "mixed", graph = "scale-free", vis = TRUE )

output <- bdgraph( data = data.sim, method = "gcgm", iter = 10000 )

summary( output )

compare( data.sim, output )

## End(Not run)

### bdgraph.npn

#### Nonparametric transfer

**Description**

This function transfers non-Gaussian data to Gaussian.

**Usage**

bdgraph.npn( data, npn = "shrinkage", npn.thresh = NULL )

**Arguments**

- **data**: It could be a \((n \times p)\) matrix or a data.frame corresponding to the data.
- **npn**: A character with three options "shrinkage" (default), "truncation", and "skeptic". Option "shrinkage" is for the shrunken transformation, option "truncation" is for the truncated transformation and option "skeptic" is for the non-paranormal skeptic transformation. For more details see references.
- **npn.thresh**: The truncation threshold; it is only for the truncated transformation (when npn = "truncation"). The default value is \(1/(4n^{1/4} \sqrt{\pi \log(n)})\).

**Value**

- **data**: A \((n \times p)\) matrix of transferred data, if npn = "shrinkage" or "truncation", and a non-paranormal correlation \((p \times p)\) matrix, if npn = "skeptic".

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**


See Also

bdgraph.sim and bdgraph

Examples

## Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 6, p = 4, size = 4 )
data <- ( data.sim $ data - 3 ) ^ 4
data

# transfer the data by truncation
bdgraph.pnn( data, npn = "truncation" )

# transfer the data by shrunken
bdgraph.pnn( data, npn = "shrunken" )

# transfer the data by skeptic
bdgraph.pnn( data, npn = "skeptic" )

## End(Not run)

bdgraph.sim

Synthetic graph data generator

Description

This function implements a synthetic graph data generation for multivariate distributions with different types of underlying graph structures, including "random", "cluster", "scale-free", "hub", "fixed", and "circle". Based on the underlying graph structure, it generates four different types of datasets, including multivariate Gaussian, non-Gaussian, discrete, or mixed data.

Usage

bdgraph.sim( n = 2, p = 10, graph = "random", size = NULL, prob = 0.2,
        class = NULL, type = "Gaussian", cut = 4, b = 3, D = diag(p),
        K = NULL, sigma = NULL, mean = 0, vis = FALSE )

Arguments

n
  The number of samples required. The default value is 2.

p
  The number of variables (nodes). The default value is 10.

graph
  The graph structure with option "random" (as a default), "cluster", "scale-free", "hub", "fixed", and "circle". It also could be adjacency matrix corresponding to a graph structure (an upper triangular matrix in which \( g_{ij} = 1 \) if there is a link between notes \( i \) and \( j \), otherwise \( g_{ij} = 0 \)).

size
  The number of links in the true graph (graph size).
For "random" graph, it is the probability that a pair of nodes has a link. The default value is 0.2.

class
Number of classes for option "cluster".

type
Type of data with four options "Gaussian" (as a default), "non-Gaussian", "discrete", and "mixed". For option "Gaussian", data are generated from multivariate normal distribution. For option "non-Gaussian", data are transferred multivariate normal distribution to continuous multivariate non-Gaussian distribution. For option "discrete", data are transferred from multivariate normal distribution to discrete multivariate distribution. For option "mixed", data are transferred from multivariate normal distribution to mixture of 'count', 'ordinal', 'non-Gaussian', 'binary' and 'Gaussian', respectively.

cut
Number of categories for simulating discrete data (type = "discrete"). The default value is 4.

b
The degree of freedom for G-Wishart distribution, \( W_G(b, D) \). The default is 3.

\( D \)
The positive definite \((p \times p)\) "scale" matrix for G-Wishart distribution, \( W_G(b, D) \). The default is an identity matrix.

\( K \)
A positive-definite symmetric matrix specifying the precision matrix. It is for option graph = "fixed".

sigma
A positive-definite symmetric matrix specifying the covariance matrix. It is for option graph = "fixed".

mean
A vector specifying the mean of the variables. The default value is a zero vector.

vis
Visualize the true graph pattern. The default value is FALSE.

Value
An object with S3 class "simulate" is returned:

data
Generated data as a \( n \times p \) matrix.

sigma
The covariance matrix of the generated data.

K
The precision matrix of the generated data.

G
The adjacency matrix corresponding to the true graph structure.

Author(s)
Abdolreza Mohammadi and Ernst Wit

References


See Also

bdgraph

Examples

```r
# Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 50, p = 10, prob = 0.3, vis = TRUE )
print(data.sim)

# generating multivariate normal data from a 'hub' graph
data.sim <- bdgraph.sim( n = 3, p = 6, graph = "hub", vis = FALSE )
data.sim $ data

# generating mixed data from a 'hub' graph
data.sim <- bdgraph.sim( n = 10, p = 8, graph = "hub", type = "mixed" )
data.sim $ data

# End(Not run)
```

---

**CellSignal**

*A flow cytometry dataset*

#### Description

This dataset contains flow cytometry of 11 proteins measured on 11672 red blood cells.

#### Usage

```r
data(CellSignal)
```

#### Format

The format is a list with 11672 rows (number of cells) and 11 column (number of proteins).

#### Source

Data are publicly available at http://www.sciencemag.org/content/308/5721/523/suppl/DC1

#### Examples

```r
data( CellSignal )
head( CellSignal )
boxplot( CellSignal )
```
## Description

With this function, we can check the performance of our method and compare it with other alternative approaches.

## Usage

```r
compare( G, est, est2 = NULL, est3 = NULL, colnames = NULL, vis = FALSE )
```

## Arguments

- **G**: The adjacency matrix corresponding to the true graph in which \( g_{ij} = 1 \) if there is a link between notes \( i \) and \( j \), otherwise \( g_{ij} = 0 \). It also can be an object with S3 class "simulate" from function `bdgraph.sim`.

- **est**: An adjacency matrix corresponding to an estimated graph. It also can be an object with S3 class "bdgraph" from function `bdgraph` or "select" (from `huge` package).

- **est2**: An adjacency matrix corresponding to an estimated graph. It also can be an object with S3 class "bdgraph" from function `bdgraph` or "select" (from `huge` package).

- **est3**: An adjacency matrix corresponding to an estimated graph. It also can be an object with S3 class "bdgraph" from function `bdgraph` or "select" (from `huge` package).

- **colnames**: A character vector giving the column names for the result table.

- **vis**: Logical: if TRUE you will see a plot result. The default is FALSE.

## Value

- **True positive**: The number of correctly estimated links.

- **True negative**: The number of true non-existing links which is correctly estimated.

- **False positive**: The number of links which they are not in the true graph, but are incorrectly estimated.

- **False negative**: The number of links which they are in the true graph, but are not estimated.

- **Accuracy**: The number of true results (both true positives and true negatives) divided by the total number of true and false results.

- **Balanced F-score**: A weighted average of the "positive predictive" and "true positive rate". F-score value reaches its best value at 1 and worst score at 0.

- **Positive predictive**: The number of correctly estimated links divided by the total number of links in the estimated graph.
True positive rate
The number of correctly estimated links divided by the total number of links in the true graph.

False positive rate
The false positive value divided by the total number of links in the true graph.

Author(s)
Abdolreza Mohammadi, Antonio Abbruzzo, Ivan Vujacic, and Ernst Wit

References


See Also
bdgraph and select

Examples
```R
# Not run:
data.sim <- bdgraph.sim( n = 50, p = 6, size = 7, vis = TRUE )

# running sampling algorithm
output <- bdgraph( data = data.sim, iter = 10000 )
# comparing the result
compare( data.sim, output, colnames = c("true", "BDgraph"), vis = TRUE )

# estimate the true graph by 'huge' package
require( huge )
huge.g <- huge( data.sim $ data, method = "mb" )
huge.g <- huge.select( huge.g )

# comparing the result
compare( data.sim, output, huge.g, colnames = c("true", "BDgraph", "huge"), vis = TRUE )

## End(Not run)
```
geneExpression

Human gene expression dataset

description

This data set contains human gene expression of 100 transcripts (with unique Illumina TargetID) measured on 60 unrelated individuals.

Usage

data(geneExpression)

Format

The format is a matrix with 60 rows (number of individuals) and 100 column (number of transcripts).

Source

The genotypes of those 60 unrelated individuals are available from the Sanger Institute website at ftp://ftp.sanger.ac.uk/pub/genevar

Examples

data( geneExpression )
dim( geneExpression )
head( geneExpression )
boxplot( geneExpression )

phat

Posterior link probabilities

Description

Provides the posterior link probabilities for all possible links in the graph.

Usage

phat( output, round = 3 )

Arguments

output An object of S3 class "bdgraph", from function bdgraph.
round A value for rounding all probabilities to the specified number of decimal places (default is 3).
Value

phat  Upper triangular matrix which corresponds the posterior probabilities for all possible links.

Author(s)

Abdolreza Mohammadi and Ernst Wit

References


See Also

bdgraph and bdgraphNsim

Examples

```r
## Not run:  # generating multivariate normal data from a 'circle' graph
data.sim <- bdgraph.sim( n = 70, p = 6, graph = "circle", vis = TRUE )

output <- bdgraph( data = data.sim, iter = 10000 )

phat( output, round = 2 )

## End(Not run)
```

---

**plot.bdgraph**  
*Plot function for S3 class "bdgraph"

Description

Visualize pattern of the graphs with the highest posterior probabilities.

Usage

```r
## S3 method for class 'bdgraph'
plot( x, g = 1, layout = layout.circle, ... )
```
Arguments

- **x**: An object of S3 class "bdgraph", from function `bdgraph`.
- **g**: The number of graphs with highest probabilities (default is 1).
- **layout**: The vertex placement algorithm which is according to igraph package. The default is "layout.circle".
- **...**: System reserved (no specific usage).

Author(s)

Abdolreza Mohammadi and Ernst Wit

References


See Also

`bdgraph`

Examples

```r
## Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 50, p = 6, vis = TRUE )
output <- bdgraph( data = data.sim )
plot(output)
plot( output, g = 4 )

## End(Not run)
```
plot.simulate  

Plot function for S3 class "simulate"

Description

Visualize structure of the true graph.

Usage

```r
## S3 method for class 'simulate'
plot( x, main = NULL, layout = layout.circle, ... )
```

Arguments

- `x`: An object of S3 class "simulate", from function `bdgraph.sim`.
- `main`: Graphical parameter (see plot).
- `layout`: The vertex placement algorithm which is according to `igraph` package. The default is "layout.circle".
- `...`: System reserved (no specific usage).

Author(s)

Abdolreza Mohammadi and Ernst Wit

References


See Also

- `bdgraph`

Examples

```r
## Not run:
# generating synthetic multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 10, p = 15 )

plot( data.sim )

## End(Not run)```
plotcoda

Convergence plot

Description
Displays a plot which provides the cumulative occupancy fractions of all possible links in the graph. The plot can be used for monitoring the convergence of the BD-MCMC sampling algorithm.

Usage
plotcoda( output, thin = NULL, main = NULL, links = TRUE, ... )

Arguments
- output: An object of S3 class "bdgraph", from function bdgraph.
- thin: An option for getting fast result for a cumulative plot according to part of the iteration.
- main: Graphical parameter (see plot).
- links: Logical: if TRUE and number of variables (p) is more than 15, you will see a plot result for 100 selected links. The default is TRUE.
- ...: System reserved (no specific usage).

Details
Note that a spending time for this function depends on the graph. It should be slow for the high-dimensional graphs. To make it faster, you can choose bigger value for 'thin' option.

Author(s)
Abdolreza Mohammadi and Ernst Wit

References

See Also
bdgraph
Examples

```r
## Not run:
# generating multivariate normal data from a 'circle' graph
data.sim < - bdgraph.sim( n = 50, p = 6, graph = "circle", vis = TRUE )
output < - bdgraph( data = data.sim )
plotcoda( output )
## End(Not run)
```

---

**plotroc**  

*ROC plot*

Description

Draws the ROC curve according to the true graph structure for the object of S3 class "bdgraph", from function `bdgraph`.

Usage

```r
plotroc( G, prob, prob2 = NULL, cut = 20, smooth = FALSE )
```

Arguments

- `G`  
  The adjacency matrix corresponding to the true graph structure in which \( g_{ij} = 1 \) if there is a link between notes \( i \) and \( j \), otherwise \( g_{ij} = 0 \). It also can be an object of S3 class "simulate", from function `bdgraph.sim`.

- `prob`  
  Upper triangular matrix which shows the estimated posterior probabilities for all possible links. It also can be an object of S3 class "bdgraph", from function `bdgraph`.

- `prob2`  
  This option is for comparing two different estimations. Upper triangular matrix which shows the estimated posterior inclusion probabilities for all possible links. It also can be an object of S3 class "bdgraph", from function `bdgraph`.

- `cut`  
  Number of cut points. The default value is 20.

- `smooth`  
  Logical: for smoothing the ROC curve. The default is FALSE.

Author(s)

Abdolreza Mohammadi and Ernst Wit
References


See Also

bdgraph and compare

Examples

```r
# Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 50, p = 6, size = 7, vis = TRUE )

# running sampling algorithm
output <- bdgraph( data = data.sim, iter = 10000 )
# comparing the result
plotroc( data.sim, output )

# To compare the result based on CGGMs approach
output2 <- bdgraph( data = data.sim, method = "gcgm", iter = 10000 )
# comparing the results
plotroc( data.sim, output, output2 )
legend("bottomright", c("Gaussian", "Copula"), lty = c(1,2), col = c(1, 4))

# End(Not run)
```

print.bdgraph

*Print function for S3 class "bdgraph"*

Description

Print the information about the best graph which is the graph with the highest posterior probability. It provides adjacency matrix, size and posterior probability of the best graph.

Usage

```r
# S3 method for class 'bdgraph'
print( x, round = 3, Khat = FALSE, phat = FALSE, ... )
```
Arguments

- **x**: An object of S3 class "bdgraph", from function `bdgraph`.
- **round**: A value to round the probabilities to the specified number of decimal places (default is 3).
- **khat**: Logical: if TRUE you will see the estimation of precision matrix. The default is FALSE.
- **phat**: Logical: if TRUE you will see the posterior link probabilities. The default is FALSE.
- ... System reserved (no specific usage).

Author(s)

Abdolreza Mohammadi and Ernst Wit

References


See Also

`bdgraph`

Examples

```r
## Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 50, p = 6, vis = TRUE )

output <- bdgraph( data = data.sim )

print( output )

print( output, Khat = TRUE, phat = TRUE )
## End(Not run)
```
**print.simulate**

*Print function for S3 class "simulate"*

**Description**

Print the information about the type of data, the sample size, the graph type, the number of nodes, number of linkes and sparsity of the true graph.

**Usage**

```r
## S3 method for class 'simulate'
print( x, ... )
```

**Arguments**

- `x`: An object of S3 class "simulate", from function `bdgraph.sim`.
- `...`: System reserved (no specific usage).

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**


**See Also**

`bdgraph.sim`

**Examples**

```r
## Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 20, p = 10, vis = TRUE )

print( data.sim )

## End(Not run)
```
Posterior probabilities of the graphs

Description

Provides the posterior probabilities for the ‘g’ most likely graphs or a specific graph, ‘G’.

Usage

`prob( output, g = 4, G = NULL )`

Arguments

- `output`: An object of S3 class "bdgraph", from function `bdgraph`.
- `g`: The number of graphs with the highest posterior probabilities to be shown (default is 4). This option is ignored if ‘G’ is specified.
- `G`: Adjacency matrix corresponding to a graph structure. It is an upper triangular matrix in which \( g_{ij} = 1 \) if there is a link between notes \( i \) and \( j \), otherwise \( g_{ij} = 0 \). It also can be an object of S3 class "simulate", from function `bdgraph_sim`.

Value

- `best.G`: The graphs with the highest posterior probabilities.

Author(s)

Abdolreza Mohammadi and Ernst Wit

References


See Also

`bdgraph`
Examples

```r
## Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 50, p = 6, size = 6, vis = TRUE )

output <- bdgraph( data = data.sim )

# Estimated posterior probability of the true graph
prob( output, G = data.sim )

# Estimated posterior probability of the first and second graphs with highest probabilities
prob( output, g = 2 )

## End(Not run)
```

### rgwish

**Sampling from G-Wishart distribution**

**Description**

Generate random matrices, distributed according to the G-Wishart distribution with parameters \( b \) and \( D \), \( W_G(b, D) \).

**Usage**

```r
rgwish( n = 1, G = NULL, b = 3, D = NULL )
```

**Arguments**

- `n` The number of samples required. The default value is 1.
- `G` Adjacency matrix corresponding to the graph structure. It is an upper triangular matrix in which \( g_{ij} = 1 \) if there is a link between notes \( i \) and \( j \), otherwise \( g_{ij} = 0 \).
- `b` The degree of freedom for G-Wishart distribution, \( W_G(b, D) \). The default value is 3.
- `D` The positive definite \((p \times p)\) "scale" matrix for G-Wishart distribution, \( W_G(b, D) \). The default is an identity matrix.

**Details**

Sampling from G-Wishart distribution, \( K \sim W_G(b, D) \), with density:

\[
Pr(K) \propto |K|^{(b-2)/2} \exp \left\{ -\frac{1}{2} \text{trace}(K \times D) \right\},
\]

which \( b > 2 \) is the degree of freedom and \( D \) is a symmetric positive definite matrix.
Value

A numeric array, say \(A\), of dimension \((p \times p \times n)\), where each \(A[:,:,i]\) is a positive definite matrix, a realization of the G-Wishart distribution, \(W_G(b, D)\).

Author(s)

Abdolreza Mohammadi and Ernst Wit

References


Examples

```r
# Not run:
G <- toeplitz(c(0, 1, rep(0, 3)))
G # graph with 5 nodes and 4 links

sample <- rgwish(n = 3, G = G, b = 3, D = diag(5))
sample

# End(Not run)
```

### Usage

```r
rwish( n = 1, p = 2, b = 3, D = diag(p) )
```
Arguments

- **n**: The number of samples required. The default value is 1.
- **p**: The number of variables (nodes). The default value is 2.
- **b**: The degree of freedom for Wishart distribution, \( W(b, D) \). The default value is 3.
- **D**: The positive definite \((p \times p)\) "scale" matrix for Wishart distribution, \( W(b, D) \). The default is an identity matrix.

Details

Sampling from Wishart distribution, \( K \sim W(b, D) \), with density:

\[
Pr(K) \propto |K|^{(b-2)/2} \exp \left\{ -\frac{1}{2} \text{trace}(K \times D) \right\},
\]

which \( b > 2 \) is the degree of freedom and \( D \) is a symmetric positive definite matrix.

Value

A numeric array, say \( A \), of dimension \((p \times p \times n)\), where each \( A[i, j] \) is a positive definite matrix, a realization of the Wishart distribution \( W(b, D) \).

Author(s)

Abdolreza Mohammadi and Ernst Wit

References


Examples

```r
## Not run:
sample <- rwish( n = 3, p = 5, b = 3, D = diag(p) )
sample
## End(Not run)
```
select

Selecting the best graph

Description

Provides the graphs with the highest posterior probabilities. For more specific selection of graphs consult the 'prob' function.

Usage

select( output, cut = NULL, vis = FALSE )

Arguments

output  
An object of S3 class "bdgraph", from function bdgraph.

cut    
Threshold for including the links in the selected graph based on the posterior probabilities of the links; See example.

vis   
Logical: if TRUE you will see the plot of best graph. The default is FALSE.

Value

G       Adjacency matrix corresponding to the selected graph.

Author(s)

Abdolreza Mohammadi and Ernst Wit

References


See Also

bdgraph
summarbdgraph

Examples

## Not run:

# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 50, p = 6, size = 7, vis = TRUE )

output <- bdgraph( data = data.sim )
select( output )
select( output, cut = 0.5, vis = TRUE )

## End(Not run)

summary.bdgraph  

Summary function for S3 class "bdgraph"

Description

This function provides a summary of the result from BD-MCMC sampling algorithm.

Usage

## S3 method for class 'bdgraph'
summary( object, vis = TRUE, ... )

Arguments

object An object of S3 class "bdgraph", from function bdgraph.
vis Logical: if TRUE (default) you will see the plot result.
... System reserved (no specific usage).

Value

best.graph The adjacency matrix corresponding to the selected graph which has the highest posterior probability.
phat Upper triangular matrix corresponding to the posterior probabilities for all possible links.
Khat The estimated precision matrix.

Author(s)

Abdolreza Mohammadi and Ernst Wit
**References**


**See Also**

bdgraph

**Examples**

```r
## Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 50, p = 6, vis = TRUE )

output <- bdgraph( data = data.sim )

summary(output)

summary( output, vis = FALSE )

## End(Not run)
```

**surveyData**

*Labor force survey data*

**Description**

The survey data concerns 1002 males in the U.S labor force, described in Hoff (2007). The seven observed variables which have been measured on various scales are as follow: the income (income), degree (degree), number of children (children), parents income (pincome), parents degree (pdegree), number of parents children (pchildren), and age (age).

**Usage**

```r
data(surveyData)
```

**Format**

The format is a matrix with 1002 rows (number of individuals) and 7 column (number of variables).
References


Examples

data( surveydata )
dim( surveydata )
head( surveydata )
boxplot( surveydata )

---

traceplot Trace plot of graph size

Description

Trace plot for graph size for the objects of S3 class "bdgraph", from function `bdgraph`. It is a tool for monitoring the convergency of the BD-MCMC sampling algorithm.

Usage

```
traceplot ( output, acf = FALSE, pacf = FALSE, main = NULL, ... )
```

Arguments

- **output** An object of S3 class "bdgraph", from function `bdgraph`.
- **acf** Logical: if TRUE you will see the plot of autocorrelation functions for graph size. The default is FALSE.
- **pacf** Logical: if TRUE you will see the plot of partial autocorrelations for graph size. The default is FALSE.
- **main** Graphical parameter (see plot).
- **...** System reserved (no specific usage).

Author(s)

Abdolreza Mohammadi and Ernst Wit
References


See Also

bdgraph

Examples

```r
## Not run:
# generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 50, p = 6, size = 7, vis = TRUE )

output <- bdgraph( data = data.sim, iter = 10000 )

traceplot(output)

traceplot( output, acf = TRUE, pacf = TRUE )

## End(Not run)
```
Index

*Topic **datasets**
  CellSignal, 8
  geneExpression, 11
  surveyData, 26
*Topic **package**
  BDgraph-package, 2
BDgraph (BDgraph-package), 2
bdgraph, 3, 6, 8–18, 20, 24–28
BDgraph-package, 2
bdgraph.npn, 5
bdgraph.sim, 3, 4, 6, 9, 12, 14, 16, 19, 20

CellSignal, 8
compare, 4, 9, 17
geneExpression, 11
phat, 11
plot.bdgraph, 12
plot.simulate, 14
plotcoda, 15
plotroc, 16
print.bdgraph, 17
print.simulate, 19
prob, 20
rgwish, 21
rwish, 22
select, 10, 24
summary.bdgraph, 4, 25
surveyData, 26
traceplot, 27