

# Package ‘BDgraph’

December 21, 2015

**Type** Package

**Title** Bayesian Graph Selection Based on Birth-Death MCMC Approach

**Version** 2.26

**Date** 2015-12-21

**Author** Abdolreza Mohammadi and Ernst Wit

**Depends** R (>= 3.0.0), Matrix, igraph

**Maintainer** Abdolreza Mohammadi <a.mohammadi@rug.nl>

**Description** Provides statistical tools for Bayesian structure learning in undirected graphical models with 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** 2015-12-21 15:29:01

**NeedsCompilation** yes

## R topics documented:

BDgraph-package . . . . .	2
bdgraph . . . . .	3
bdgraph.npn . . . . .	5
bdgraph.sim . . . . .	6
compare . . . . .	8
geneExpression . . . . .	10
gnorm . . . . .	11
pgraph . . . . .	12
plinks . . . . .	13
plot.bdgraph . . . . .	14
plot.sim . . . . .	16

plotcoda . . . . .	17
plotroc . . . . .	18
print.bdgraph . . . . .	19
print.sim . . . . .	20
rgwish . . . . .	21
rwish . . . . .	23
select . . . . .	24
summary.bdgraph . . . . .	25
surveyData . . . . .	26
traceplot . . . . .	27

<b>Index</b>	<b>29</b>
--------------	-----------

---

BDgraph-package	<i>Bayesian Structure Learning in Graphical Models</i>
-----------------	--

---

## Description

The R package **BDgraph** provides statistical tools for Bayesian structure learning in undirected graphical models. The package is implemented the recent improvements in the Bayesian literature, including Mohammadi and Wit (2015) and Mohammadi et al. (2015). The computationally intensive tasks of the package is implemented in C++ and interfaced with R, to speed up the computations. Besides, the package contains several functions for simulation and visualization, as well as three multivariate datasets taken from the literature.

## Details

The package includes 10 main functions:

bdgraph	Search algorithm in graphical models
bdgraph.sim	Synthetic graph data generator
bdgraph.npn	Nonparametric transfer
compare	Comparing the results
plinks	Estimated posterior link probabilities
plotcoda	Convergence plot
plotroc	ROC plot
rgwish	Sampling from G-Wishart distribution
select	Graph selection
traceplot	Trace plot of graph size

## Author(s)

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

## References

- Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138
- Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*
- Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*
- Lenkoski, A. (2013). A direct sampler for G-Wishart variates, *Stat*, 2:119-128

---

 bdgraph

*Search algorithm in graphical models*


---

## Description

As the main function of the **BDgraph** package, this function consists of several sampling algorithms for Bayesian model determination in undirected graphical models.

## Usage

```
bdgraph( data, n = NULL, method = "ggm", algorithm = "bdmcmc", iter = 5000,
  burnin = iter / 2, g.start = "empty", prior.df = 3,
  multi.update = NULL, save.all = FALSE )
```

## Arguments

- |           |   |
|-----------|---|
| data      | There are two options: (1) an $(n \times p)$ matrix or a data.frame corresponding to the data, (2) an $(p \times p)$ covariance matrix as $S = X'X$ which $X$ is the data matrix ( $n$ is the sample size and $p$ is the number of variables). It also could be an object of class "sim", from function <code>bdgraph.sim</code> . The input matrix is automatically identified by checking the symmetry. |
| n         | The number of observations. It is needed if the "data" is a covariance matrix.  |
| method    | A character with two options "ggm" (default) and "gcg". Option "ggm" is for Gaussian graphical models based on Gaussianity assumption. Option "gcg" is for Gaussian copula graphical models for the data that not follow Gaussianity assumption (e.g. continuous non-Gaussian, discrete, or mixed dataset).   |
| algorithm | A character with two options "bdmcmc" (default) and "rjmc". Option "bdmcmc" is based on birth-death MCMC algorithm. Option "rjmc" is based on reversible jump MCMC algorithm.   |
| iter      | The number of iteration for the sampling algorithm.   |
| burnin    | The number of burn-in iteration for the sampling algorithm.   |
| g.start   | Corresponds to a starting point of the graph. It could be "empty" (default) and "full". Option "empty" means the initial graph is an empty graph and "full" means a full graph. It also could be an object with S3 class "bdgraph"; with this option we could run the sampling algorithm from the last objects of previous run (see examples).  |

prior.df	The degree of freedom for G-Wishart distribution, $W_G(b, D)$ , which is a prior distribution of the precision matrix. The default is 3.
multi.update	It is only for the BDMCMC algorithm ( <code>algorithm = "bdmcmc"</code> ). It is for simultaneously updating multiple links at the same time to update graph in the BDMCMC algorithm.
save.all	Logical: if FALSE (default), the adjacency matrices are NOT saved. If TRUE, the adjacency matrices after burn-in are saved.

### Value

An object with S3 class "bdgraph" is returned:

p_links	An upper triangular matrix which corresponds the estimated posterior probabilities of all possible links.
K_hat	The posterior estimation of the precision matrix.

For the case "save.all = TRUE" is returned:

sample_graphs	A vector of strings which includes the adjacency matrices of visited graphs after burn-in.
graph_weights	A vector which includes the waiting times of visited graphs after burn-in.
all_graphs	A vector which includes the identity of the adjacency matrices for all iterations after burn-in. It is needed for monitoring the convergence of the BD-MCMC algorithm.
all_weights	A vector which includes the waiting times for all iterations after burn-in. It is needed for monitoring the convergence of the BD-MCMC algorithm.

### Author(s)

Abdolreza Mohammadi and Ernst Wit

### References

- Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138
- Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*
- Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

### See Also

[bdgraph.sim](#), [summary.bdgraph](#), and [compare](#)

**Examples**

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

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

summary( bdgraph.obj )

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

# Running algorithm with starting points from previous run
bdgraph.obj2 <- bdgraph( data = data.sim, iter = 5000, g.start = bdgraph.obj )

compare( data.sim, bdgraph.obj, bdgraph.obj2,
         colnames = c( "True graph", "Frist run", "Second run" ) )

# Generating mixed data from a 'scale-free' graph
data.sim <- bdgraph.sim( n = 50, p = 6, type = "mixed", graph = "scale-free", vis = TRUE )

bdgraph.obj <- bdgraph( data = data.sim, method = "gcm", iter = 10000 )

summary( bdgraph.obj )

compare( data.sim, bdgraph.obj )

## End(Not run)
```

---

bdgraph.npn

*Nonparametric transfer*


---

**Description**

Transfers non-Gaussian data to Gaussian.

**Usage**

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

**Arguments**

data	An $(n \times p)$ matrix or a data frame corresponding to the data ( $n$ is the sample size and $p$ is the number of variables).
npn	A character with three options "shrinkage" (default), "truncation", and "skeptical". Option "shrinkage" is for the shrunken transformation, option "truncation" is for the truncated transformation and option "skeptical" is for the non-paranormal skeptical 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 An  $(n \times p)$  matrix of transferred data, if npn = "shrinkage" or "truncation", and a non-paranormal correlation  $(p \times p)$  matrix, if npn = "skeptical".

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

Liu, H., F. Han, M. Yuan, J. Lafferty, and L. Wasserman (2012). High Dimensional Semiparametric Gaussian Copula Graphical Models, *Annals of Statistics* 40(4):2293-2326

Zhao, T. and H. Liu (2012). The **huge** Package for High-dimensional Undirected Graph Estimation in R, *Journal of Machine Learning Research*, 13:1059-1062

**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.npn( data, npn = "truncation" )

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

# Transfer the data by skeptic
bdgraph.npn( data, npn = "skeptical" )

## End(Not run)
```

---

bdgraph.sim

*Synthetic graph data generator*

---

**Description**

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, type = "Gaussian", graph = "random", prob = 0.2,
             size = NULL, mean = 0, class = NULL, cut = 4, b = 3,
             D = diag(p), K = NULL, sigma = NULL, 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.
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.
graph	The graph structure with option "random" (default), "cluster", "scale-free", "hub", "fixed", and "circle". It also could be an adjacency matrix corresponding to a graph structure (an upper triangular matrix in which $g_{ij} = 1$ if there is a link between nodes $i$ and $j$ , otherwise $g_{ij} = 0$ ).
prob	If graph="random", it is the probability that a pair of nodes has a link. The default value is 0.2.
size	The number of links in the true graph (graph size).
mean	A vector specifies the mean of the variables. The default value is a zero vector.
class	If graph="cluster", it is the number of classes.
cut	If type="discrete", it is the number of categories for simulating discrete data. 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	If graph="fixed", it is a positive-definite symmetric matrix specifies as a true precision matrix.
sigma	If graph="fixed", it is a positive-definite symmetric matrix specifies as a true covariance matrix.
vis	Visualize the true graph structure. The default value is FALSE.

**Value**

An object with S3 class "sim" is returned:

data	Generated data as an $(n \times p)$ 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**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

**See Also**

[bdgraph](#)

**Examples**

```
## 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 )
round( data.sim $ data, 2 )

# Generating mixed data from a 'hub' graph
data.sim <- bdgraph.sim( n = 10, p = 8, graph = "hub", type = "mixed" )
round( data.sim $ data, 2 )

## End(Not run)
```

---

compare

*Comparing the results*

---

**Description**

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

**Usage**

```
compare( sim.obj, bdgraph.obj, bdgraph.obj2 = NULL, bdgraph.obj3 = NULL,
         colnames = NULL, vis = FALSE )
```



**Arguments**

sim.obj	An object with S3 class "sim" from function <code>bdgraph.sim</code> . It also can be the adjacency matrix corresponding to the true graph structure.
bdgraph.obj	An object with S3 class "bdgraph" from function <code>bdgraph</code> . It also can be an adjacency matrix corresponding to an estimated graph.
bdgraph.obj2	An object with S3 class "bdgraph" from function <code>bdgraph</code> . It also can be an adjacency matrix corresponding to an estimated graph. It is for comparing two different approaches.
bdgraph.obj3	An object with S3 class "bdgraph" from function <code>bdgraph</code> . It also can be an adjacency matrix corresponding to an estimated graph. It is for comparing three different approaches.
colnames	A character vector giving the column names for the result table.
vis	Visualize the true graph and estimated graph structures. 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**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

### See Also

[bdgraph](#) and [select](#)

### Examples

```
## 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 based on GGMs
sample.ggm <- bdgraph( data = data.sim, method = "ggm", iter = 10000 )
# Comparing the results
compare( data.sim, sample.ggm, colnames = c( "True graph", "GGM" ), vis = TRUE )

# Running sampling algorithm based on GCGMs
sample.gcgm <- bdgraph( data = data.sim, method = "gcgm", iter = 10000 )
# Comparing GGM and GCGM methods
compare( data.sim, sample.ggm, sample.gcgm, colnames = c("True graph", "GGM", "GCGM"), vis = TRUE )

## End(Not run)
```

---

geneExpression

*Human gene expression dataset*

---

### Description

This dataset 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 )
```

gnorm

*Log of the normalizing constant of G-Wishart distribution***Description**

Calculates log of the normalizing constant of G-Wishart distribution based on the Monte Carlo method, developed by Atay-Kayis and Massam (2005).

**Usage**

```
gnorm( adj.g, b = 3, D = diag( ncol(adj.g) ), iter = 100 )
```

**Arguments**

<code>adj.g</code>	The adjacency matrix corresponding to the graph structure. It is an upper triangular matrix in which $a_{ij} = 1$ if there is a link between notes $i$ and $j$ , otherwise $a_{ij} = 0$ .
<code>b</code>	The degree of freedom for G-Wishart distribution, $W_G(b, D)$ . The default value is 3.
<code>D</code>	The positive definite ( $p \times p$ ) "scale" matrix for G-Wishart distribution, $W_G(b, D)$ . The default is an identity matrix.
<code>iter</code>	The number of iteration for the Monte Carlo approximation. The default value is 100.

**Details**

Log of the normalizing constant approximation using Monte Carlo method for a G-Wishart distribution,  $K \sim W_G(b, D)$ , with density:

$$Pr(K) = \frac{1}{I(b, D)} |K|^{(b-2)/2} \exp \left\{ -\frac{1}{2} \text{trace}(K \times D) \right\}.$$

**Value**

Log of the normalizing constant of G-Wishart distribution.

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

## References

- Atay-Kayis, A. and H. Massam (2005). A monte carlo method for computing the marginal likelihood in nondecomposable Gaussian graphical models, *Biometrika*, 92(2):317-335
- Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

## Examples

```
adj.g <- matrix( c(0,0,1,
                  0,0,1,
                  0,0,0), 3, 3, byrow = TRUE )
# adj.g: adjacency matrix of graph with 3 nodes and 2 links

gnorm( adj.g, b = 3, D = diag(3) )
```

---

pgraph

*Posterior probabilities of the graphs*

---

## Description

Provides the estimated posterior probabilities for the most likely graphs or a specific graph.

## Usage

```
pgraph( bdgraph.obj, number.g = 4, adj.g = NULL )
```

## Arguments

bdgraph.obj	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> .
number.g	The number of graphs with the highest posterior probabilities to be shown (default is 4). This option is ignored if 'adj.g' is specified.
adj.g	An adjacency matrix corresponding to a graph structure. It is an upper triangular matrix in which $a_{ij} = 1$ if there is a link between nodes $i$ and $j$ , otherwise $a_{ij} = 0$ . It also can be an object of S3 class "sim", from function <a href="#">bdgraph.sim</a> .

## Value

selected_g	The graphs with the highest posterior probabilities.
prob_g	A vector of the posterior probabilities of the graphs corresponding to 'selected_g'.

## Author(s)

Abdolreza Mohammadi and Ernst Wit

## References

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

## See Also

[bdgraph](#)

## Examples

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

bdgraph.obj <- bdgraph( data = data.sim, save.all = TRUE )

# Estimated posterior probability of the true graph
pgraph( bdgraph.obj, adj_g = data.sim )

# Estimated posterior probability of first and second graphs with highest probabilities
pgraph( bdgraph.obj, number.g = 2 )

## End(Not run)
```

---

plinks

*Estimated posterior link probabilities*

---

## Description

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

## Usage

```
plinks( bdgraph.obj, round = 3 )
```

## Arguments

`bdgraph.obj` 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**

p\_links            An upper triangular matrix which corresponds the estimated posterior probabilities for all possible links.

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

**See Also**

[bdgraph](#) and [bdgraph.sim](#)

**Examples**

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

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

plinks( bdgraph.obj, round = 2 )

## End(Not run)
```

---

plot.bdgraph

*Plot function for S3 class "bdgraph"*

---

**Description**

Visualizes structure of the selected graphs which could be a graph with links for which their estimated posterior probabilities are greater than 0.5 or graph with the highest posterior probability.

**Usage**

```
## S3 method for class 'bdgraph'
plot( x, cut = 0.5, number.g = 1, layout = layout.circle, ... )
```

**Arguments**

x	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> .
cut	Threshold for including the links in the selected graph based on the estimated posterior probabilities of the links; See the examples.
number.g	The number of graphs with the highest probabilities (default is 1).
layout	The vertex placement algorithm which is according to <b>igraph</b> package. The default is "layout.circle".
...	System reserved (no specific usage).

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

**See Also**

[bdgraph](#)

**Examples**

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

bdgraph.obj <- bdgraph( data = data.sim )

plot( bdgraph.obj )

bdgraph.obj <- bdgraph( data = data.sim, save.all = TRUE )

plot( bdgraph.obj, number.g = 4 )

plot( bdgraph.obj, cut = 0.4 )

## End(Not run)
```

---

plot.sim	<i>Plot function for S3 class "sim"</i>
----------	---

---

**Description**

Visualizes structure of the true graph.

**Usage**

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

**Arguments**

x	An object of S3 class "sim", from function <a href="#">bdgraph.sim</a> .
main	Graphical parameter (see plot).
layout	The vertex placement algorithm which is according to <b>igraph</b> package. The default is "layout.circle".
...	System reserved (no specific usage).

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108v2*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849v2*

**See Also**

[bdgraph](#)

**Examples**

```
## 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	<i>Convergence plot</i>
----------	-------------------------

---

**Description**

Visualizes the cumulative occupancy fractions of all possible links in the graph. It can be used for monitoring the convergence of the sampling algorithms, BDMCMC and RJMCMC.

**Usage**

```
plotcoda( bdgraph.obj, thin = NULL, control = TRUE, main = NULL, ... )
```

**Arguments**

bdgraph.obj	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> .
thin	An option for getting fast result for a cumulative plot according to part of the iteration.
control	Logical: if TRUE (default) and the number of nodes is greater than 15, then 100 links randomly is selected for visualization.
main	Graphical parameter (see plot).
...	System reserved (no specific usage).

**Details**

Note that a spending time for this function depends on the number of nodes.

For fast result, you can choose bigger value for the 'thin' option.

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

**See Also**

[bdgraph](#)

**Examples**

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

bdgraph.obj <- bdgraph( data = data.sim, iter = 10000, burnin = 0 , save.all = TRUE )

plotcoda( bdgraph.obj )

## End(Not run)
```

---

plotroc

*ROC plot*


---

**Description**

Draws the ROC curve according to the true graph structure for object of S3 class "bdgraph", from function [bdgraph](#).

**Usage**

```
plotroc( sim.obj, bdgraph.obj, bdgraph.obj2 = NULL, bdgraph.obj3 = NULL,
         cut.num = 20, smooth = FALSE, label = TRUE )
```

**Arguments**

sim.obj	An object of S3 class "sim", from function <a href="#">bdgraph.sim</a> . It also can be the adjacency matrix corresponding to the true graph structure in which $a_{ij} = 1$ if there is a link between notes $i$ and $j$ , otherwise $a_{ij} = 0$ .
bdgraph.obj	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> . It also can be an upper triangular matrix corresponding to the estimated posterior probabilities for all possible links.
bdgraph.obj2	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> . It also can be an upper triangular matrix corresponding to the estimated posterior probabilities for all possible links. It is for comparing two different approaches.
bdgraph.obj3	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> . It also can be an upper triangular matrix corresponding to the estimated posterior probabilities for all possible links. It is for comparing three different approaches.
cut.num	Number of cut points. The default value is 20.
smooth	Logical: for smoothing the ROC curve. The default is FALSE.
label	Logical: for adding legend to the ROC plot. The default is TRUE.

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

## References

- Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138
- Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*
- Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

## See Also

[bdgraph](#) and [compare](#)

## Examples

```
## Not run:
# Generating multivariate normal data from a 'random' graph
data.sim <- bdgraph.sim( n = 30, p = 6, size = 7, vis = TRUE )

# Running sampling algorithm
bdgraph.obj <- bdgraph( data = data.sim, iter = 10000 )
# Comparing the results
plotroc( data.sim, bdgraph.obj )

# To compare the results based on CGGMs approach
bdgraph.obj2 <- bdgraph( data = data.sim, method = "gcgms", iter = 10000 )
# Comparing the results
plotroc( data.sim, bdgraph.obj, bdgraph.obj2, label = FALSE )
legend( "bottomright", c( "GGMs", "GCGMs" ), lty = c( 1, 2 ), col = c( 1, 4 ) )

## End(Not run)
```

---

print.bdgraph                      *Print function for S3 class "bdgraph"*

---

## Description

Prints the information about the selected graph which could be a graph with links for which their estimated posterior probabilities are greater than 0.5 or graph with the highest posterior probability. It provides adjacency matrix, size and posterior probability of the selected graph.

## Usage

```
## S3 method for class 'bdgraph'
print( x, round = 3, ... )
```

**Arguments**

x	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> .
round	A value to round the probabilities to the specified number of decimal places (default is 3).
...	System reserved (no specific usage).

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

**See Also**

[bdgraph](#)

**Examples**

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

bdgraph.obj <- bdgraph( data = data.sim )

print( bdgraph.obj )

## End(Not run)
```

---

print.sim

*Print function for S3 class "sim"*

---

**Description**

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

**Usage**

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

**Arguments**

x                    An object of S3 class "sim", from function [bdgraph.sim](#).  
...                   System reserved (no specific usage).

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138  
Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108v2*  
Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849v2*

**See Also**

[bdgraph.sim](#)

**Examples**

```
## 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)
```

---

rgwish

*Sampling from G-Wishart distribution*

---

**Description**

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

**Usage**

```
rgwish( n = 1, adj.g = NULL, b = 3, D = NULL )
```

**Arguments**

n	The number of samples required. The default value is 1.
adj.g	The adjacency matrix corresponding to the graph structure. It should be an upper triangular matrix in which $a_{ij} = 1$ if there is a link between notes $i$ and $j$ , otherwise $a_{ij} = 0$ .
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**

- Lenkoski, A. (2013). A direct sampler for G-Wishart variates, *Stat*, 2:119-128
- Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138
- Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*
- Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

**Examples**

```
## Not run:
adj.g <- toeplitz( c( 0, 1, rep( 0, 3 ) ) )
adj.g    # adjacency of graph with 5 nodes and 4 links

sample <- rgwish( n = 3, adj.g = adj.g, b = 3, D = diag(5) )
round( sample, 2 )

## End(Not run)
```

---

rwish	<i>Sampling from Wishart distribution</i>
-------	---

---

**Description**

Generates random matrices, distributed according to the Wishart distribution with parameters  $b$  and  $D$ ,  $W(b, D)$ .

**Usage**

```
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  $A[, , i]$  is a positive definite matrix, a realization of the Wishart distribution  $W(b, D)$ .

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

- Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138
- Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*
- Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

**Examples**

```
## Not run:
p      <- 5
sample <- rwish( n = 3, p = p, b = 3, D = diag(p) )
round( sample, 2 )

## End(Not run)
```

---

select

*Graph selection*


---

**Description**

Provides the selected graph which, based on input, could be a graph with links for which their estimated posterior probabilities are greater than 0.5 (as a default) or a graph with the highest posterior probability; see examples.

**Usage**

```
select( bdgraph.obj, cut = NULL, vis = FALSE )
```

**Arguments**

bdgraph.obj	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> .
cut	Threshold for including the links in the selected graph based on the estimated posterior probabilities of the links; see the examples.
vis	Visualize the selected graph structure. The default value is FALSE.

**Value**

G	An adjacency matrix corresponding to the selected graph.
---	--

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*



**See Also**[bdgraph](#)**Examples**

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

bdgraph.obj <- bdgraph( data = data.sim )

select( bdgraph.obj )

bdgraph.obj <- bdgraph( data = data.sim, save.all = TRUE )

select( bdgraph.obj )

select( bdgraph.obj, cut = 0.5, vis = TRUE )

## End(Not run)
```

summary.bdgraph

*Summary function for S3 class "bdgraph"***Description**

Provides a summary of the results for function [bdgraph](#).

**Usage**

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

**Arguments**

object	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> .
vis	Visualize the results. The default value is TRUE.
...	System reserved (no specific usage).

**Value**

best.graph	The adjacency matrix corresponding to the selected graph which has the highest posterior probability.
p_links	An upper triangular matrix corresponding to the posterior probabilities of all possible links.
K_hat	The estimated precision matrix.

**Author(s)**

Abdolreza Mohammadi and Ernst Wit

**References**

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

**See Also**

[bdgraph](#)

**Examples**

```
## Not run:  
# Generating multivariate normal data from a 'random' graph  
data.sim <- bdgraph.sim( n = 50, p = 6, size = 7, vis = TRUE )  
  
bdgraph.obj <- bdgraph( data = data.sim )  
  
summary( bdgraph.obj )  
  
bdgraph.obj <- bdgraph( data = data.sim, save.all = TRUE )  
  
summary( bdgraph.obj )  
  
summary( bdgraph.obj, vis = FALSE )  
  
## End(Not run)
```

---

surveyData

*Labor force survey data*

---

**Description**

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

**Usage**

```
data( surveyData )
```

**Format**

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

**References**

Hoff, P. (2007). Extending the rank likelihood for semiparametric copula estimation, *The Annals of Applied Statistics*, 1(1), 265-283.

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *Arxiv preprint arXiv:1501.05108*

**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 convergence of the sampling algorithms, BDMCMC and RJMCMC.

**Usage**

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

**Arguments**

bdgraph.obj	An object of S3 class "bdgraph", from function <a href="#">bdgraph</a> .
acf	Visualize the autocorrelation functions for graph size. The default is FALSE.
pacf	Visualize the 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

Mohammadi, A. and E. Wit (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models, *Bayesian Analysis*, 10(1):109-138

Mohammadi, A. and E. Wit (2015). **BDgraph**: An R Package for Bayesian Structure Learning in Graphical Models, *arXiv:1501.05108*

Mohammadi, A., F. Abegaz Yazew, E. van den Heuvel, and E. Wit (2015). Bayesian Gaussian Copula Graphical Modeling for Dupuytren Disease, *arXiv:1501.04849*

## See Also

[bdgraph](#)

## Examples

```
## Not run:  
# Generating multivariate normal data from a 'random' graph  
data.sim <- bdgraph.sim( n = 50, p = 6, size = 7, vis = TRUE )  
  
bdgraph.obj <- bdgraph( data = data.sim, iter = 10000, burnin = 0, save.all = TRUE )  
  
traceplot( bdgraph.obj )  
  
traceplot( bdgraph.obj, acf = TRUE, pacf = TRUE )  
  
## End(Not run)
```

# Index

## \*Topic **datasets**

geneExpression, 10

surveyData, 26

## \*Topic **package**

BDgraph-package, 2

BDgraph (BDgraph-package), 2

bdgraph, 3, 6, 8–10, 12–20, 24–28

BDgraph-package, 2

bdgraph.npn, 5

bdgraph.sim, 3, 4, 6, 6, 9, 12, 14, 16, 18, 21

compare, 4, 8, 19

geneExpression, 10

gnorm, 11

pgraph, 12

plinks, 13

plot.bdgraph, 14

plot.sim, 16

plotcoda, 17

plotroc, 18

print.bdgraph, 19

print.sim, 20

rgwish, 21

rwish, 23

select, 10, 24

summary.bdgraph, 4, 25

surveyData, 26

traceplot, 27