Package 'gmat'

August 30, 2020

Type Package

Title Simulation of Graphical Correlation Matrices

Version 0.2.2

Date 2020-08-29

Encoding UTF-8

Description Simulation of correlation matrices possibly constrained by a given undirected or acyclic directed graph. In particular, the package provides functions that implement the simulation methods described in Córdoba et al. (2018) <doi:10.1007/978-3-030-03493-1_13> and Córdoba et al. (2020) <doi:10.1016/j.ijar.2020.07.007>.

License GPL (>= 2)

RoxygenNote 7.1.1

ByteCompile true

URL https://github.com/gherardovarando/gmat

BugReports https://github.com/gherardovarando/gmat/issues

Suggests testthat, covr

Imports gRbase, igraph, methods, stats

NeedsCompilation yes

Author Irene Córdoba [aut] (<https://orcid.org/0000-0002-3252-4234>),
Gherardo Varando [aut, cre] (<https://orcid.org/0000-0002-6708-1103>),
Concha Bielza [ths] (<https://orcid.org/0000-0001-7109-2668>),
Pedro Larrañaga [ths] (<https://orcid.org/0000-0003-0652-9872>)

Maintainer Gherardo Varando <gherardo.varando@math.ku.dk>

Repository CRAN

Date/Publication 2020-08-30 01:00:18 UTC

R topics documented:

`dag-constrained correlation matrices`

Simulation of correlation matrices

Description

Sample correlation matrices, possibly with a zero pattern in its Cholesky decomposition constrained by an acyclic digraph.

Usage

```
chol_mh(N = 1, p = 3, d = 1, dag = NULL, ...)

chol_iid(N = 1, p = 3, d = 1, dag = NULL)
```

Arguments

`N`	Number of samples.
`p`	Matrix dimension. Ignored if `dag` is provided.
`d`	Number in [0,1], the proportion of non-zero entries in the Cholesky factor of the sampled matrices. Ignored if `dag` is provided.
`dag`	An igraph acyclic digraph specifying the zero pattern in the upper Cholesky factor of the sampled matrices. Nodes must be in ancestral order, with the first one having no parents.
`...`	Additional parameters for `mh_u()`.

Details

Function `chol_mh()` uses the method described in Córdoba et al. (2018) and implemented in `mh_u()`, based on a Metropolis-Hastings algorithm over the upper Cholesky factorization.

The entries in the upper Cholesky factor are sampled i.i.d. by function `chol_iid()`, following Kalisch and Buhlmann (2007).

Value

A three-dimensional array of length p x p x N.

References

Córdoba I., Varando G., Bielza C., Larrañaga P. A fast Metropolis-Hastings method for generating random correlation matrices. *Lecture Notes in Computer Science* (IDEAL 2018), vol 11314, pp. 117-124, 2018.

Kalisch, M., Buhlmann, P. Estimating high-dimensional directed acyclic graphs with the PC-algorithm, *Journal of Machine Learning Research*, 8:613-636, 2007.

Examples

```
## Cholesky sampling via Metropolis-Hastings
# Generate a full matrix (default behaviour)
chol_mh()

# Generate a matrix with a percentage of zeros
chol_mh(d = 0.5)

# Generate a random acyclic digraph structure
dag <- rgraph(p = 3, d = 0.5, dag = TRUE)
igraph::print.igraph(dag)

# Generate a matrix complying with the predefined zero pattern
chol_mh(dag = dag)
## Cholesky sampling via i.i.d. Cholesky factor
# Generate a full matrix (default behaviour)
chol_iid()

# Generate a matrix with a percentage of zeros
chol_iid(d = 0.5)

# Generate a matrix complying with the predefined zero pattern
igraph::print.igraph(dag)
chol_iid(dag = dag)
```

`gmat` *gmat*

Description

Simulation of graphical correlation matrices

metropolis-hastings sampling *Upper Cholesky factor sampling using Metropolis-Hastings*

Description

Metropolis-Hasting algorithms to sample the upper Cholesky factor, using positive hemispheres of different dimensions. A zero pattern may be specified using an acyclic digraph.

Usage

```
mh_u(N = 1, p = 3, dag = NULL, ...)

mh_sphere(N = 1, k, i = 1, h = 100, eps = 0.01)
```

Arguments

N	Number of samples.
p	Dimension of the upper Cholesky factor.
dag	An igraph acyclic digraph specifying the zero pattern in the upper Cholesky factor of the sampled matrices. Nodes must be in ancestral order, with the first one having no parents.
...	Additional parameters for `mh_sphere()`.
k	Dimension of the hemisphere from which the sample is taken.
i	Integer, power of the first coordinate in the density.
h	Heating phase size.
eps	Perturbation variance.

Details

Function `mh_u()` returns a sample of `N` upper Cholesky factors whose rows have been generated using `mh_sphere()`. The dimensions of the hemispheres used to sample vary depending both on the row number of the Cholesky factor, and whether there is a zero pattern specified by `dag`.

The details of the algorithm implemented by `mh_sphere()` can be found in the paper Córdoba et al. (2018), including a discussion on theoretical convergence and numerical experiments for choosing its hyper parameters `h` and `eps`.

Author(s)

Gherardo Varando <`gherardo.varando@math.ku.dk`>

References

Córdoba I., Varando G., Bielza C., Larrañaga P. A fast Metropolis-Hastings method for generating random correlation matrices. *Lecture Notes in Computer Science* (IDEAL 2018), vol 11314, pp. 117-124, 2018.

Examples

```
## Upper Cholesky factor sampling
# Generate a random acyclic digraph
dag <- rgraph(p = 3, d = 0.5, dag = TRUE)
igraph::print.igraph(dag)

# Generate an upper Cholesky factor complying with such zero pattern
mh_u(dag = dag)
# We may also generate it with no zero pattern (full upper triangular)
mh_u()
## Hemisphere sampling
# 3D hemisphere from a density proportional to the square of the first coordinate
mh_sphere(N = 4, k = 3, i = 2)
```

`rgraph` *Random generation of acyclic digraphs and undirected graphs*

Description

Wrapper of functionality from package `igraph` for random generation of graphs.

Usage

```
rgraph(p, d, dag = FALSE, ordered = TRUE)
```

Arguments

`p`	Number of vertices of the sampled graph.
`d`	Proportion of edges in the generated graph.
`dag`	Whether the generated graph should be acyclic directed.
`ordered`	When generating an acyclic directed graph, whether the nodes should follow the ancestral order 1, ..., p.

Details

When `dag = FALSE`, the graph is sampled from an Erdos-Renyi model. In the case where `dag = TRUE`, the upper triangle of the adjacency matrix of an Erdos-Renyi model is taken as the adjacency matrix for the acyclic digraph. This preserves the proportion of edges `d`.

Value

The generated graph.

Examples

```
## Random undirected graph with 3 nodes and 50% density of edges
rgraph(p = 3, d = 0.5)

## Random directed acyclic graphs
# Following the natural ancestral order 1, ..., p
dag <- rgraph(p = 6, d = 0.5, dag = TRUE)
igraph::topo_sort(dag)

# Following a random ancestral order
dag <- rgraph(p = 6, d = 0.5, dag = TRUE, ordered = FALSE)
igraph::topo_sort(dag)
```

set_cond_number *Set the condition number of the matrices in a sample of covariance/correlation matrices*

Description

Set the condition number of the matrices in a sample of covariance/correlation matrices

Usage

```
set_cond_number(sample, k)
```

Arguments

`sample`	Array, the p x p x N matrix sample.
`k`	Condition number to be set.

Value

A p x p x N array containing the matrices with the fixed condition number.

uchol *Get the upper factor of the upper Cholesky decomposition of a symmetric positive definite matrix.*

Description

Get the upper factor of the upper Cholesky decomposition of a symmetric positive definite matrix.

Usage

```
uchol(m)
```

Arguments

`m`	Matrix to factorize.

Details

The upper factor `U` such that `m = U %*% t(U)`. `U` is equal to the transpose with respect to the anti-diagonal of the standard Cholesky factor `L` in `m_rev = L %*% t(L)`, where `m_rev` is the matrix resulting from reverting the order of rows and columns in `m` (see Córdoba et al., 2019, Section 2.2 for more details). The function uses the base `chol` method.

Value

A `p*p` upper triangular matrix.

References

Córdoba I., Varando G., Bielza C., Larrañaga P., Generating random Gaussian graphical models, *arXiv*:1909.01062, 2019.

`ug-constrained correlation matrices`

Simulation of correlation matrices.

Description

Sample correlation matrices, possibly with a zero pattern constrained by an undirected graph.

Usage

```
port(N = 1, p = 3, d = 1, ug = NULL, rfun = stats::rnorm, ...)

port_chol(N = 1, p = 3, d = 1, ug = NULL, ...)

diagdom(N = 1, p = 3, d = 1, ug = NULL, rfun = stats::rnorm, ...)
```

Arguments

`N`	Number of samples.
`p`	Matrix dimension. Ignored if ug is provided.
`d`	Number in [0,1], the proportion of non-zero entries in the sampled matrices. Ignored if ug is provided.
`ug`	An igraph undirected graph specifying the zero pattern in the sampled matrices.
`rfun`	Function that generates the random entries in the initial factors, except for `port_chol()` which uses `mh_u()` to obtain it.
`...`	Additional parameters to be passed to `rfun` or to `mh_u()`.

Details

Function port() uses the method described in Córdoba et al. (2018). In summary, it consists on generating a random matrix Q and performing row-wise orthogonalization such that if i and j are not adjacent in ug, then the rows corresponding to such indices are orthogonalized, without violating previous orthogonalizations and without introducing unwanted independences. The resulting matrix after the process has finished is the cross product of Q.

Function port_chol() uses the method described in Córdoba et al. (2019), combining uniform sampling with partial orthogonalization as follows. If the graph provided is not chordal, then a chordal cover is found using gRbase::triangulate(). Then uniform sampling for the upper Choleksy factor corresponding to such chordal cover is performed with mh_u(). Finally, it uses partial orthogonalization as port() to add the missing zeros (corresponding to fill-in edges in the chordal cover). The behaviour of this function is the same as port().

We also provide an implementation of the most commonly used in the literature diagdom(). By contrast, this method produces a random matrix M with zeros corresponding to missing edges in ug, and then enforces a dominant diagonal to ensure positive definiteness. Matrices produced by diagdom usually are better conditioned than those by port; however, they typically suffer from small off-diagonal entries, which can compromise model validation in Gaussian graphical models. This is avoided by port.

Value

A three-dimensional array of length p x p x N.

References

Córdoba, I., Varando, G., Bielza, C. and Larrañaga, P. A partial orthogonalization method for simulation covariance and concentration graph matrices. *Proceedings of Machine Learning Research* (PGM 2018), vol. 72, pp. 61 - 72, 2018.

Córdoba, I., Varando, G., Bielza, C. and Larrañaga, P. On generating random Gaussian graphical models. *International Journal of Approximate Reasoning* , vol. 125, pp.240 - 250, 2020.

Examples

```
## Partial orthogonalization
# Generate a full matrix (default behaviour)
port()

# Generate a matrix with a percentage of zeros
port(d = 0.5)

# Generate a random undirected graph structure
ug <- rgraph(p = 3, d = 0.5)
igraph::print.igraph(ug)

# Generate a matrix complying with the predefined zero pattern
port(ug = ug)
## Diagonal dominance
# Generate a full matrix (default behaviour)
diagdom()
```

```
# Generate a matrix with a percentage of zeros
diagdom(d = 0.5)

# Generate a matrix complying with the predefined zero pattern
igraph::print.igraph(ug)
diagdom(ug = ug)
```

ug_to_dag *Moral DAG from non chordal UG*

Description

Find the DAG with no v-structures whose skeleton is a triangulation of a given undirected graph.

Usage

```
ug_to_dag(ug)
```

Arguments

`ug`	An igraph undirected graph.

Value

An igraph acyclic directed graph orientation.

vectorize *Vectorize a sample of covariance/correlation matrices*

Description

Vectorize a sample of covariance/correlation matrices

Usage

```
vectorize(sample)
```

Arguments

`sample`	Array, the p x p x N sample to vectorize.

Details

Note that if the sample is of covariance matrices, as returned by `port()` and `diagdom()`, the diagonal is omitted from the vectorization process.

Value

A p*(p - 1)/2 x N matrix containing the vectorized sample.

Index