Package 'scorelingam'

December 5, 2022

Type Package

Title What the package does (short line)

Version 1.0

Date 2022-12-04

Author Gabriel Ruiz

Maintainer Gabriel Ruiz

Description Implements the sorting procedure outlined in Ruiz et. al (2022+).

URL https://gabriel-ruiz.github.io/scorelingam/, https://openreview.net/forum?id=4pCjIGIjrt

BugReports https://github.com/gabriel-ruiz/scorelingam/issues

Imports Rcpp (>= 1.0.9)

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 7.2.0

R topics documented:

scorelingam-package 2
checkSortingErrors 2
corrMat 3
dlap 3
generateData 4
getParents 4
getWeights 5
moralize 6
randomWeightedAdjacencyMatrix 6
rlap 7
scorelingam 7
skeleton 8

Index **9**

scorelingam-package *ScoreLiNGAM*

Description

Implements the sorting procedure outlined in Ruiz et. al (2022+).

Details

The main procedure is `scorelingam`. See its documentation below for more info.

Author(s)

Gabriel Ruiz

References

G Ruiz, OH Madrid-Padilla, Q Zhou. "Sequentially learning the topological ordering of directed acyclic graphs with likelihood ratio scores." Transactions of Machine Learning Research, 2022+. [Link]

See Also

`scorelingam`

Examples

```
See the examples at https://github.com/gabriel-ruiz/scorelingam.
```

checkSortingErrors *Sorting Error.*

Description

Sorting Error.

Usage

```
checkSortingErrors(estOrder, A)
```

Arguments

`estOrder:`	A length p vector with the estimated topological ordering. Its unique elements are a re-ordering of the indices 1,2,...,p.
`A:`	The true DAG adjacency matrix.

Value

Returns proportion of parents sorted after children. See Equation (11) in the paper, which defines the output of this function: $\mathrm{ERR}(\hat{\pi})$.

`corrMat` *Quickly estimate a Pearson correlation matrix using RcppArmadillo.*

Description

Quickly estimate a Pearson correlation matrix using RcppArmadillo.

Usage

```
corrMat(X)
```

Arguments

`X`	An n by p data matrix which is used to estimate the correlation matrix.

Value

A p by p estimate of the correlation matrix.

`dlap` *laplace density of x*

Description

laplace density of x

Usage

```
dlap(x, center = mean(x), shape = mean(abs(x - center)), log = F)
```

`generateData` *Generate Linear (laplace, logistic, or scaled-t error family) Structural Equation Model Data*

Description

Generate Linear (laplace, logistic, or scaled-t error family) Structural Equation Model Data

Usage

```
generateData(
  B,
  n = 1000,
  family = "laplace",
  scale = rep(1, ncol(B)),
  perm = topoSort(as.matrix(B)),
  df = 5
)
```

Arguments

`B`	A p by p weighted adjacency matrix for an acyclic linear SEM.
`n`	The desired sample size. Default: `n=1000`.
`family`	Currently allow 't', 'laplace' (default), or 'logistic'.
`scale`	Length p vector s.t. $\mathrm{var}[\epsilon_j] = (\mathrm{scale}[j])^2 \times \mathrm{var}[\epsilon_0]$ with $\epsilon_0 \sim \mathrm{family}(1)$ (the chosen family with scale parameter 1). Default: `scale=rep(1,ncol(B))`.
`df`	Degrees of freedom for scaled-t-distributed noise. Default: `df=5`.

Value

An n by p matrix object.

`getParents` *Get parent sets given Markov blanket and topological ordering*

Description

Get parent sets given Markov blanket and topological ordering

Usage

```
getParents(mb, ordering)
```

Arguments

mb	A length p list object whose j-th entry gives the Markov blanket of node j. The j-th entry can also be some other set that defines the possible support for the parent set.
ordering:	A length p vector with the specified topological ordering of the nodes in the underlying DAG. Its unique elements are a re-ordering of the indices 1,2,...,p.

Value

A list object of length p; entry j denotes the parent set of node j in the underlying DAG.

`getWeights` *Get weighted adjancency matrix for linear SEM. Assumes that* `X` *is zero-centered, which can be done via* `X=scale(Xoriginal,center=T,scale=F)`. *Estimation is done via Ordinary Least Squares (OLS) regression.*

Description

Get weighted adjancency matrix for linear SEM. Assumes that `X` is zero-centered, which can be done via `X=scale(Xoriginal,center=T,scale=F)`. Estimation is done via Ordinary Least Squares (OLS) regression.

Usage

```
getWeights(X, pa)
```

Arguments

X	An n by p data matrix which is used to estimate the weighted adjacency matrix.
pa	A length p list object whose j-th entry gives the support for node j's parent set.

Value

A p by p weighted adjacency matrix for the linear SEM.

`moralize` *Moralize a DAG. Obtain Markov Blanket, the union of all children, parents, co-parents.*

Description

Moralize a DAG. Obtain Markov Blanket, the union of all children, parents, co-parents.

Usage

```
moralize(B)
```

Arguments

`B`	The p by p weighted adjacency matrix. Assumes `B[j,k]!=0` if $j \in PA_k$.

Value

A list of length p. Each entry j corresponds to the Markov Blanket of node j according to adjacency matrix `B`.

`randomWeightedAdjacencyMatrix` *Generate a random weighted adjacency matrix with num.roots number of root nodes. Each non-root node will have between pa.min and pa.max number of parents. The parent weight (in absolute value) is between pa.wt.min and pa.wt.max. The parent weight is positive with probability prob.pos (else, negative).*

Description

Generate a random weighted adjacency matrix with num.roots number of root nodes. Each non-root node will have between pa.min and pa.max number of parents. The parent weight (in absolute value) is between pa.wt.min and pa.wt.max. The parent weight is positive with probability prob.pos (else, negative).

Usage

```
randomWeightedAdjacencyMatrix(
  p,
  num.roots = p - 1,
  pa.min = 1,
  pa.max = 1,
  pa.wt.min = 1,
  pa.wt.max = 1,
  prob.pos = 0.5,
  perm = sample(x = p, size = p, replace = F)
)
```

Arguments

p	Number of nodes in underlying DAG.
num.roots	Number of root nodes in underlying DAG. Default: num.roots=p-1.
pa.min	Minimum number of parents a node can have. Default: pa.min=1.
pa.max	Maximum number of parents a node can have. Default: pa.max=1.
pa.wt.min	The minimum (in absolute value) coefficient value for the parent to a child in the linear SEM. Default: pa.wt.min=1.
pa.wt.max	The maximum (in absolute value) coefficient value for the parent to a child in the linear SEM. Default: pa.wt.max=1.
prob.pos	The probability of a positive coefficient value for the parent to a child in the the linear SEM. Default: \cdoeprob.pos=0.5.
perm	A topological ordering for underlying DAG. Default is perm=sample(p,p,replace=F).

Value

A list with two elements: perm (the topological ordering used) and B (the weighted adjacency matrix).

---

rlap *laplace r.v.*

Description

laplace r.v.

Usage

```
rlap(n = 1, mu = 0, b = 1)
```

---

scorelingam *The ScoreLiNGAM sorting procedure.*

Description

The ScoreLiNGAM sorting procedure.

Usage

```
scorelingam(X, mb, numUpdates = 5L, family = "laplace", df = 10)
```

Arguments

X	An n by p data matrix which is used to estimate the permutation.
mb	A length p list object whose j-th entry gives the Markov blanket of node j (or its superset).
numUpdates	The number of updates to give while sorting. Default: numUpdates=5.
family	Currently allows 't', 'laplace' (default), or 'logistic'.
df	Degrees of freedom for scaled-t-distributed noise. Default: df=10.

Value

A length p vector with the specified topological ordering of the nodes in the underlying DAG. Its unique elements are a re-ordering of the indices 1,2,...,p.

---

skeleton — *Adjacency matrix for an undirected (moral) graph based on Markov Blanket (or neighborhood estimates) of each node.*

Description

Adjacency matrix for an undirected (moral) graph based on Markov Blanket (or neighborhood estimates) of each node.

Usage

```
skeleton(mb)
```

Arguments

mb	The Markov blanket or neighborhood estiamtes. A list of length p.

Value

A p by p adjacency matrix for an undirected graph. 0 indicates no undirected edge in the graph, while 1 indicates an directed edge in the graph.

Index

∗ **package**
scorelingam-package, 2

checkSortingErrors, 2
corrMat, 3

dlap, 3

generateData, 4
getParents, 4
getWeights, 5

moralize, 6

randomWeightedAdjacencyMatrix, 6
rlap, 7

scorelingam, 2, 7
scorelingam *(*scorelingam-package*)*, 2
scorelingam-package, 2
skeleton, 8