Package ’BN_Data_Generator’

December 24, 2014

Type Package

Title Data Generator based on Bayesian Network Model

Version 1.0

Date 2014-12-24

Depends R (>=3.1.2)

Suggests bnlearn

Author Jae-seong Yoo

Maintainer Jae-seong Yoo <praster1@gmail.com>

Description Previous tools suffer from serious trade-off between cost and complexity, restricting most studies relevant to Bayesian network to using only real data. To address such problem, a data generator based on Bayesian network model using R is built and introduced.

URL http://www.github.com/praster1/BN_Data_Generator

License GPL (>=2)

Repository CRAN

R topics documented:

BN_Data_Generator-package

Data Generator based on Bayesian Network Model

Details

Package:	BN_Data_Generator
Type:	Package
Version:	1.0
Date:	2014-12-24
License:	GPL (>=2)

Author(s)

Jae-seong Yoo <praster1@gmail.com>

References

Jae-seong Yoo, (2014), "A Study on Comparison of Bayesian Network Structure Learning Algorithms for Selecting Appropriate Models", M.S. thesis, Department of Statistics, Korea University, Seoul.

See Also

~~ Optional links to other man pages, e.g. ~~ ~~ <pkg> ~~

Examples

```
~~ simple examples of the most important functions ~~
```

big_letters

Description

the 26 lower-case letters of the Roman alphabet;

Usage

```
big_letters(len)
```

Arguments

`len`

BN_Data_Generator *Data Generator based on Bayesian Network Model*

Description

Data Generator based on Bayesian Network Model

Usage

```
BN_Data_Generator(arcs_mat, input_Probs, n, node_names = NULL, cardinalities = NULL)
```

Arguments

arcs_mat	A matrix that determines the arcs.
input_Probs	The conditional probabilities.
n	Data size.
node_names	The names of each nodes.
cardinalities	The cardinalities of each nodes.

References

Jae-seong Yoo, (2014), "A Study on Comparison of Bayesian Network Structure Learning Algorithms for Selecting Appropriate Models", M.S. thesis, Department of Statistics, Korea University, Seoul.

check_cardinalities

Usage

```
check_cardinalities(arcs_mat, node_names = NULL, cardinalities = NULL)
```

Arguments

arcs_mat	A matrix that determines the arcs.
node_names	The names of each nodes.
cardinalities	The cardinalities of each nodes.

C_M_WO_WC

Description

The existence of the known network structures allows us to define three important terms which indicate the performance of the algorithm (in terms of the number of graphical errors in the learnt structure).

Usage

```
C_M_WO_WC(target_arcs_mat, learnt_arcs_mat)
```

Arguments

`target_arcs_mat`
A matrix of known network structure.

`learnt_arcs_mat`
A matrix of learnt network structure.

Value

`C (Correct Arcs)`
Edges present in the original network and in the learnt network structure.

`M (Missing Arcs)`
Edges present in the original network but not in the learnt network structure.

`WO (Wrongly Oriented Arcs)`
Edges present in the learnt network structure, but having opposite orientation when compared with the corresponding edge in the original network structure.

`WC (Wrongly Corrected Arcs)`
Edges not present in the original network but included in the learnt network structure.

References

X.-w. Chen, G. Anantha, and X. Wang, (2006), An effective structure learning method for constructing gene networks, Bioinformatics, Vol. 22, 1367-1374.

fromto_to_mat

Usage

```
fromto_to_mat(fromto, node_names)
```

Arguments

fromto	aa
node_names	The names of each nodes.

gen_asia

Usage

```
gen_asia()
```

is_acyclic

Description

This function checks for each node in a DAG whether backtracing arcs leading to it results in an "infinite recursion" error indicating that there actually is a cyclic part in the DAG (which then obviously seems not to be a DAG).

Usage

```
is_acyclic(arcs_mat)
```

Arguments

arcs_mat	A matrix that determines the arcs.

Value

A list with two elements. acyclic is a boolean indicating whether the DAG is acyclic (=TRUE) or contains a cyclic component (=FALSE). nodewise is a vector containing 1 boolean per node in the DAG, TRUE indicating that backtracing from this node does not lead to a cyclic component, FALSE indicating that backtracing from this node leads to a cyclic component.

See Also

is_DAG

is_DAG

Description

This function tests whether the given graph is a DAG, a directed acyclic graph.

Usage

```
is_DAG(arcs_mat)
```

Arguments

`arcs_mat`	A matrix that determines the arcs.

Details

is_dag checks whether there is a directed cycle in the graph. If not, the graph is a DAG.

Value

A logical vector of length one.

See Also

`is_acyclic`

make_topology

Description

Bayesian Networks with varying topologies (DAGs) with number of nodes.

Usage

```
make_topology(nodes, topology = "Collapse", input_Probs = NULL, node_names = NULL, cardinalities = NUL
```

Arguments

`nodes`	The number of nodes.
`topology`	Geometric characteristic.
`input_Probs`	The conditional probabilities.
`node_names`	The names of each nodes.
`cardinalities`	The cardinalities of each nodes.

Details

The volume of the manifold is a geometric characteristic associated with the BN<e2><80><99>s topology. Each BN produces a different magnitude of the volume based on the BN<e2><80><99>s DAG. <e2><80><9c>Collapse<e2><80><9d>, <e2><80><9c>Line<e2><80><9d>, <e2><80><9c>Star<e2><80><9d>, <e2><80><9c>PseudoLoop<e2><80><9d>, <e2><80><9c>Diamond<e2><80><9d>, <e2><80><9c>Rhombus<e2><80><

References

Eitel J. M. L., (2008), An Information-geometric approach to learning Bayesian network topologies from data, Innovations in Bayesian Networks Studies in Computational Intelligence, Vol. 156, 187-217.

`mat_to_fromto`

Usage

```
mat_to_fromto(arcs_mat)
```

Arguments

`arcs_mat`	A matrix that determines the arcs.

`real_alarm`

Usage

```
real_alarm(n, rep = T)
```

Arguments

`n`	Data size.
`rep`	Should sampling be with replacement?

`real_asia`

Usage

```
real_asia(n, rep = T)
```

Arguments

`n`	Data size.
`rep`	Should sampling be with replacement?

real_hailfinder

Usage

```
real_hailfinder(n, rep = T)
```

Arguments

n	Data size.
rep	Should sampling be with replacement?

real_insurance

Usage

```
real_insurance(n, rep = T)
```

Arguments

n	Data size.
rep	Should sampling be with replacement?

real_lizards

Usage

```
real_lizards(n, rep = T)
```

Arguments

n	Data size.
rep	Should sampling be with replacement?

toss_value

Description

Sets up a sample space for the experiment of tossing a coin repeatedly with the outcomes "H" or "T".

Usage

```
toss_value(times, num_of_cases, makespace = FALSE)
```

Arguments

`times`	Number of tosses.
`num_of_cases`	Cardinality.
`makespace`	Logical.

Value

A data frame, with an equally likely probs column if makespace is TRUE.

Examples

```
toss_value(1, 3)
toss_value(2, 3)
toss_value(3, 4, makespace = TRUE)
```

Index