Package 'TDARACNE'

March 26, 2013

Type Package

Title Network reverse engineering from time course data.

Version 1.8.0

Date 2009-11-11

Author Zoppoli P.,Morganella S., Ceccarelli M.

Maintainer Zoppoli Pietro <zoppoli.pietro@gmail.com>

Depends GenKern, Rgraphviz, Biobase

biocViews Microarray, TimeCourse

Description

To infer gene networks from time-series measurements is a current challenge into bioinformatics research area. In order to detect dependencies between genes at different time delays, we propose an approach to infer gene regulatory networks from time-series measurements starting from a well known algorithm based on information theory. The proposed algorithm is expected to be useful in reconstruction of small biological directed networks from time course data.

License GPL-2

collate IcEfx.R PercentileC.R RangeRank2.R CalcMI_time2.R bootstrap.R saveTime.R MItimeIcE2.R MItimeThreshperm2.R DPI_TDAracne.R DPI2_TDAracne.R ToTheGraph_timeShiftmax2.R TDARACNE.R TDARACNEdataPublished.R plotRgraphviz.R

LazyLoad yes

R topics documented:

---

bootstrap *bootstrap*

Description

make a block bootstrap. See the reference paper

Usage

```
bootstrap(TS)
```

Arguments

TS	TS is the time series that have to be bootstrapped

CalcMI_time2 *CalcMI_time2*

Description

Compute the d-delayed Mutual information

Usage

```
CalcMI_time2(l, t, delta)
```

Arguments

l	one gene profile
t	another gene profile
delta	maximum delay

dataIRMAoff *dataIRMAoff*

Description

data used to infer the IRMAoff network

Usage

```
data(dataIRMAoff)
```

Format

The format is: Formal class 'ExpressionSet' [package "Biobase"] with 7 slots ..@ assayData :<environment: 0x115feb540> ..@ phenoData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 21 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "sampleNames" "sampleColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ featureData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 5 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "featureNames" "featureColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ experimentData :Formal class 'MIAME' [package "Biobase"] with 13 slots .. .. ..@ name : chr "" .. .. ..@ lab : chr "" .. .. ..@ contact : chr "" .. .. ..@ title : chr "" .. .. ..@ abstract : chr "" .. .. ..@ url : chr "" .. .. ..@ pubMedIds : chr "" .. .. ..@ samples : list() .. .. ..@ hybridizations : list() .. .. ..@ normControls : list() .. .. ..@ preprocessing : list() .. .. ..@ other : list() .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 0 0 ..@ annotation : chr(0) ..@ protocolData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 21 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "sampleNames" "sampleColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. .. ..@ .Data:List of 4 .. ..$: int [1:3] 2 11 0 .. ..$: int [1:3] 2 8 0 .. ..$: int [1:3] 1 3 0 .. ..$: int [1:3] 1 0 0

Details

gene on the rows and time points on the columns

dataIRMAon *dataIRMAon*

Description

data used to infer the IRMA network

Usage

data(dataIRMAon)

Format

The format is: Formal class 'ExpressionSet' [package "Biobase"] with 7 slots ..@ assayData :<environment: 0x1159767c8> ..@ phenoData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 16 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "sampleNames" "sampleColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ featureData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 5 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "featureNames" "featureColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ experimentData :Formal class 'MIAME' [package "Biobase"] with 13 slots .. .. ..@ name : chr "" .. .. ..@ lab : chr "" .. .. ..@ contact : chr "" .. .. ..@ title : chr "" .. .. ..@ abstract : chr "" .. .. ..@ url : chr "" .. .. ..@ pubMedIds : chr "" .. .. ..@ samples : list() .. .. ..@ hybridizations : list() .. .. ..@ normControls : list() .. .. ..@ preprocessing : list() .. .. ..@ other : list() .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 0 0 ..@ annotation : chr(0) ..@ protocolData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 16 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "sampleNames" "sampleColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. .. ..@ .Data:List of 4 .. ..$: int [1:3] 2 11 0 .. ..$: int [1:3] 2 8 0 .. ..$: int [1:3] 1 3 0 .. ..$: int [1:3] 1 0 0

Details

gene on the rows and time points on the columns

---

dataSOSmean *dataSOSmean*

Description

data used to infer the E.coli SOS network

Usage

data(dataSOSmean)

Format

The format is: Formal class 'ExpressionSet' [package "Biobase"] with 7 slots ..@ assayData :<environment: 0x1159ba7f0> ..@ phenoData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 14 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "sampleNames" "sampleColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"]

with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ featureData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 8 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "featureNames" "featureColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ experimentData :Formal class 'MIAME' [package "Biobase"] with 13 slots .. .. ..@ name : chr "" .. .. ..@ lab : chr "" .. .. ..@ contact : chr "" .. .. ..@ title : chr "" .. .. ..@ abstract : chr "" .. .. ..@ url : chr "" .. .. ..@ pubMedIds : chr "" .. .. ..@ samples : list() .. .. ..@ hybridizations : list() .. .. ..@ normControls : list() .. .. ..@ preprocessing : list() .. .. ..@ other : list() .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 0 0 ..@ annotation : chr(0) ..@ protocolData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 14 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "sampleNames" "sampleColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. .. ..@ .Data:List of 4 .. ..$: int [1:3] 2 11 0 .. ..$: int [1:3] 2 8 0 .. ..$: int [1:3] 1 3 0 .. ..$: int [1:3] 1 0 0

Details

gene on the rows and time points on the columns

---

dataYeast *dataYeast*

Description

data used to infer a partial Yeast G1/M cell cycle network

Usage

```
data(dataYeast)
```

Format

The format is: Formal class 'ExpressionSet' [package "Biobase"] with 7 slots ..@ assayData :<environment: 0x100e32508> ..@ phenoData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 16 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "sampleNames" "sampleColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ featureData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 11 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "featureNames" "featureColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ experimentData :Formal class 'MIAME' [package "Biobase"] with 13 slots .. .. ..@ name : chr "" .. .. ..@ lab : chr "" .. .. ..@ contact : chr "" .. .. ..@ title : chr "" .. .. ..@ abstract : chr "" .. .. ..@ url : chr "" .. .. ..@ pubMedIds : chr "" .. .. ..@ samples : list() .. .. ..@ hybridizations : list() .. .. ..@ normControls : list() .. .. ..@ preprocessing : list() .. .. ..@ other : list() .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List

of 1 .. ..$: int [1:3] 1 0 0 ..@ annotation : chr(0) ..@ protocolData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots .. .. ..@ varMetadata :'data.frame': 0 obs. of 1 variable: .. ..$ labelDescription: chr(0) .. .. ..@ data :'data.frame': 16 obs. of 0 variables .. .. ..@ dimLabels : chr [1:2] "sampleNames" "sampleColumns" .. .. ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. ..@ .Data:List of 1 .. ..$: int [1:3] 1 1 0 ..@ .__classVersion__:Formal class 'Versions' [package "Biobase"] with 1 slots .. .. ..@ .Data:List of 4 .. ..$: int [1:3] 2 11 0 .. ..$: int [1:3] 2 8 0 .. ..$: int [1:3] 1 3 0 .. ..$: int [1:3] 1 0 0

Details

gene on the rows and time points on the columns

DPI2_TDAracne *DPI2_TDAracne*

Description

MAke the second DPI

Usage

```
DPI2_TDAracne(MItab, tolerance)
```

Arguments

MItab	MItab is the adjacency matrix before the second DPI
tolerance	tolerance is the DPI tolerance.

DPI_TDAracne *DPI_TDAracne*

Description

MAake the first DPI

Usage

```
DPI_TDAracne(MItab, delta, tolerance)
```

Arguments

MItab	MItab is the adjacency matrix before DPI
delta	delta is the maximum time delay allowed to infer connections.
tolerance	tolerance is the DPI tolerance.

IcEfx *IcEfx*

Description

Select the point of Initial change Expression of the genes

Usage

IcEfx(z, likehood, logarit)

Arguments

z	z is the data matrix
likehood	likehood is the fold change used as threshold to state the initial change expression
logarit	if z is log put logarithm == 0;

MItimeIcE2 *MItimeIcE2*

Description

Compute the d-delayed Mutual information all over the whole set of genes

Usage

MItimeIcE2(z, N, delta, norm, threshold, ksd, IcE)

Arguments

z	z is the data matrix
N	N is respectively the number of bins in percentile normalization or in rank normalization
delta	delta is the maximum time delay allowed to infer connections
norm	if you want column percentile normalization put norm == 1; if you want Rank normalization put norm == 2;
threshold	the Influence threshold. if you have a threshold and a SD put them here in this format: c(thresh,SD) if you don't have threshold put 0 in thresh;
ksd	ksd is the standard deviation multiplier;
IcE	the IcE value

MItimeThreshperm2 *MItimeThreshperm2*

Description

Compute the threshold of the d-delayed Mutual information

Usage

MItimeThreshperm2(z, N, delta, norm)

Arguments

z	z is the data matrix
N	N is respectively the number of bins in percentile normalization or in rank normalization
delta	delta is the maximum time delay allowed to infer connections
norm	if you want column percentile normalization put norm == 1; if you want Rank normalization put norm == 2;

PercentileC *PercentileC*

Description

Percentile row normalization, each column goes from 0 to 1

Usage

PercentileC(z, N)

Arguments

z	z is the data matrix
N	N is respectively the number of bins in percentile normalization or in rank normalization

plotRgraphviz *plotRgraphviz*

Description

use Rgraphviz to plot the adj; bonus help function

Usage

plotRgraphviz(Influence)

Arguments

Influence	the adj matrix

RangeRank2 *RangeRank2*

Description

Column Rank discretization and normalization, each row goes from 0 to 1

Usage

```
RangeRank2(z, N)
```

Arguments

z	the data matrix
N	number of bins

saveTime *saveTime*

Description

make some useful check on the data

Usage

```
saveTime(newz, delta)
```

Arguments

newz	newz is the data matrix
delta	delta is the maximum time delay allowed to infer connections

TDARACNE *TDARACNE*

Description

Main function, see P. Zoppoli, S. Morganella, M. Ceccarelli. TimeDelay-ARACNE: Reverse engineering of gene networks from time-course data by an information theoretic approach. BMC Bioinformatics 2010, 11:154.

Usage

```
TDARACNE(eSet,N,delta=3,likehood=1.2,norm=2,logarithm=1,thresh=0,ksd=1,tolerance=0.15,plot=FALSE
```

Arguments

eSet	eSet is the ExpressionSet object
N	N is respectively the number of bins in percentile normalization or in rank normalization
delta	delta is the maximum time delay allowed to infer connections
likehood	likehood is the fold change used as threshold to state the initial change expression (IcE)
norm	if you want column percentile normalization put norm == 1; if you want Rank normalization put norm == 2;
logarithm	if z is log put logarithm == 0;
thresh	the Influence threshold. if you have a threshold and a SD put them here in this format: c(thresh,SD) if you don't have threshold put 0 in thresh;
ksd	ksd is the standard deviation multiplier;
tolerance	tolerance is the DPI tolerance; 0 means no tolerance 1 means no DPI 0.15 is the default ARACNE tolerance as it is for TDARACNE
plot	plot must be TRUE to obtain automatically the graph
dot	dot must be TRUE to obtain a .dot file
name	name must be written with quotation marks(like this:'examplename') and is the name of the .dot file produced;
adj	adj must be TRUE to obtain an adjacent matrix

Examples

```
## take paper data
library(TDARACNE)
data(dataIRMAon)
data(threshIRMAon)
## main function; in output gives to you and adj matrix and a .dot file
# eSet is the ExpressionSet object
# N is respectively the number of bins in percentile normalization or in rank normalization
# delta is the maximum time delay allowed to infer connections
# likehood is the fold change used as threshold to state the initial change expression (IcE)
# if you want column percentile normalization put norm == 1;
# if you want Rank normalization put norm == 2;
# if z is log put logarithm == 0;
# if you don't have threshold put 0 in thresh;
# ksd is the standard deviation multiplier;
# tolerance is the DPI tolerance;
# plot must be TRUE to obtain automatically the graph
# dot must be TRUE to obtain a .dot file
# name must be written with quotation marks(like this:'examplename') and is the name of the .dot file produced;
# adj must be TRUE to obtain an adjacent matrix

TDARACNE(dataIRMAon,11,"netIRMAon",delta=3,likehood=1.2,norm=2,logarithm=1,thresh=threshIRMAon,ksd=
```

TDARACNEdataPublished *TDARACNEdataPublished*

Description

main function with reference paper data. This reproduce the paper results. Simply run the function with no arguments to obtain the paper results.

Usage

```
TDARACNEdataPublished()
```

Examples

```
## take the paper data
library(TDARACNE)
data(dataYeast)
data(dataSOSmean)
data(dataIRMAon)
data(threshIRMAon)
data(threshSOSmean)
data(threshYeast)
## paper results
    TDARACNEdataPublished()
## see in your working directory for .dot files
```

threshIRMAon *threshIRMAon*

Description

IRMAon thresh

Usage

```
data(threshIRMAon)
```

Format

The format is: num [1:2] 0.593 0.309

threshSOSmean *threshSOSmean*

Description

SOS thresh

Usage

```
data(threshSOSmean)
```

Format

The format is: num [1:2] 0.428 0.311

threshYeast *threshYeast*

Description

Yeast threshold

Usage

```
data(threshYeast)
```

Format

The format is: num [1:2] 0.216 0.156

ToTheGraph_timeShiftmax2 *ToTheGraph_timeShiftmax2*

Description

make a .dot file of the adj

Usage

```
ToTheGraph_timeShiftmax2(network, name)
```

Arguments

network	the adj matrix
name	name for the .dot file

Index