

Package ‘ARNetMiT’

January 6, 2017

Type Package

Title Association Rules based Gene Network of miRNA Targets

Version 0.2

Date 2017-01-06

Author M. Ozgur Cingiz

Maintainer M. Ozgur Cingiz <ozgurcingiz@gmail.com>

Depends R (>= 2.10),
hash,
RedeR, GAnet

Description This package infers gene-gene interactions from miRNA-target gene interactions dataset using association rules based algorithm (Eclat).

License GPL (>= 2.10)

LazyData true

RoxygenNote 5.0.1

R topics documented:

buildmiRNATargetTable	2
buildNetworkbyARules	2
createMirnaListfromDisease	3
disease	4
diseaseMirnaData	5
inferNet2BinaryFormat	6
mirnaGeneData	7
prostateData	8
visualizeNet	8
Index	10

`buildmiRNATargetTable` *Create miRNA-targets table for gene network inference*

Description

`buildmiRNATargetTable` Take miRNA target interactions in binary format then converts it into machine processable format.

Usage

```
buildmiRNATargetTable(dataset)
```

Arguments

<code>dataset</code>	miRNA-target gene interactions dataset. The first column must contain miRNAs and the second column must contain target genes.
----------------------	---

Value

`buildmiRNATargetTable` returns miRNA-targets table

See Also

[buildNetworkbyARules](#), [visualizeNet](#), [inferNet2BinaryFormat](#), [createMirnaListfromDisease](#)

Examples

```
data(prostateData)
table=buildmiRNATargetTable(prostateData)
```

`buildNetworkbyARules` *All in one function to infer network of miRNA targets using association rules*

Description

`buildNetworkbyARules` takes a miRNA-target gene dataset as input and then infers gene co-expression networks using hash based association rules based algorithm.

Usage

```
buildNetworkbyARules(dataset,support,confidenceVal,supportType="rank", filtering="s")
```

Arguments

dataset	miRNA-target gene interactions dataset. The first column must contain miRNAs and the second column must contain target genes.
support	The support value of association rules based algorithm. Users can determine the support value when supportType parameter is selected as "rank"
confidenceVal	The confidence value of association rules based algorithm.
supportType	It enables users to choose the support value types. It can be "rank", "firstQuartile", "median", "thirdQuartile" and "mean". If this parameter is determined as "rank" then the support parameter become active.
filtering	It is a filtering (or pruning) parameter to eliminate some weak associations to build robust GCN. This parameter can be "s" that indicates usage of only support value for interaction pruning or it can be "sc" that indicates usage of support and confidence value for interaction pruning.

Value

buildNetworkbyARules returns environment res that returns final interactions as \$finalNetwork, the common miRNAs of gene-gene interactions are given as \$geneMirna, the lift values of the interactions are given as \$liftTable, the confidence values of the interactions are given as \$confidenceTable, the support values of the interactions are given as \$binaryInteractions and the filtered genes of the first elimination steps are given as \$firstStep.

See Also

[createMirnaListfromDisease](#), [visualizeNet](#), [inferNet2BinaryFormat](#), [buildmiRNATargetTable](#)

Examples

```
data(prostateData)

res <- buildNetworkbyARules(prostateData,0.3,0.5,supportType="rank", filtering="s")
net <- res$finalNetwork
```

```
createMirnaListfromDisease
```

To obtain disease related miRNA-target genes dataset

Description

createMirnaListfromDisease It returns the disease related miRNA-target genes dataset.

Usage

```
createMirnaListfromDisease(mirnaGeneData,diseaseMirnaData,disease,diseaseIndice)
```

Arguments

- mirnaGeneData miRNAs-target genes dataset which consists of two columns. First column contains miRNAs and the second columns contains target genes.
- diseaseMirnaData miRNAs-diseases dataset which consists of two columns. First column contains miRNAs and the second columns contains diseases.
- disease It shows the disease list.
- diseaseIndice It indicates the disease indice of the disease list. ex: the prostate cancer indice is 11 in disease list

Value

createMirnaListfromDisease Return the disease related miRNA-target genes dataset.

See Also

[buildNetworkbyARules](#), [visualizeNet](#), [inferNet2BinaryFormat](#), [buildmiRNATargetTable](#)

Examples

```
data(mirnaGeneData)
data(diseaseMirnaData)
data(disease)

prostateData= createMirnaListfromDisease(mirnaGeneData,diseaseMirnaData,disease,11);
```

disease	<i>The disease list</i>
---------	-------------------------

Description

disease It returns the disease list

Usage

```
data( disease )
```

Arguments

- disease It is a dataset that contains disease list

Details

It consists of unique 152 diseases that are obtained from miR2Disease dataset.

Value

disease returns the disease list

References

Jiang, Q., Wang, Y., Hao, Y., et al. miR2Disease: a manually curated database for microRNA deregulation in human disease, Nucleic Acids Res., 2009, 37, (Suppl 1), pp. D98- D104

Examples

```
data(disease)
```

diseaseMirnaData	<i>miRNA-disease relations</i>
------------------	--------------------------------

Description

diseaseMirnaData Experimentally validated miRNA-disease relations

Usage

```
data( diseaseMirnaData )
```

Arguments

diseaseMirnaData

It is a dataset that contains experimentally validated miRNA-disease relations

Details

It consists of unique 2429 miRNA-disease relations. These relations are obtained from 477 unique miRNAs and 152 unique diseases. This dataset is derived from "miR2Disease".

Value

diseaseMirnaData returns experimentally validated miRNA-disease relations

References

Jiang, Q., Wang, Y., Hao, Y., et al. miR2Disease: a manually curated database for microRNA deregulation in human disease, Nucleic Acids Res., 2009, 37, (Suppl 1), pp. D98- D104

Examples

```
data(diseaseMirnaData)
```

inferNet2BinaryFormat *Convert inferred interactions to binary interaction format*

Description

inferNet2BinaryFormat Convert inferred interactions to binary interaction format.

Usage

```
inferNet2BinaryFormat (net,genes,filtering="s")
```

Arguments

net	It can be either n*n matrix of support values of interactions or n*n matrix of confidence values of interactions
genes	The data contains gene names.
filtering	It is a filtering (or pruning) parameter to eliminate some weak associations to build robust GCN. This parameter can be "s" that indicates usage of only support value for interaction pruning or it can be "sc" that indicates usage of support and confidence value for interaction pruning.

Value

inferNet2BinaryFormat returns environment res that returns gene-gene interactions in binary format as res\$interactions and it also returns indices of interactions as res\$indices.

See Also

[buildNetworkbyARules](#), [visualizeNet](#), [createMirnaListfromDisease](#), [buildmiRNATargetTable](#)

Examples

```
data(prostateData)

temp <- buildNetworkbyARules(prostateData,0.3,0.5,supportType="rank", filtering="s")
net <- temp$binaryInteractions;
genes <- temp$firstStep;

res <- inferNet2BinaryFormat (net,genes,filtering="s");
binaryFormat= res$interactions;
```

mirnaGeneData	<i>miRNA-target gene interactions dataset</i>
---------------	---

Description

mirnaGeneData Experimentally validated miRNA-target gene interactions

Usage

```
data( mirnaGeneData )
```

Arguments

mirnaGeneData	It is a dataset that contains experimentally validated miRNA-target gene interactions
---------------	---

Details

mirnaGeneData consists of two validated miRNA-target gene datasets. First datasets is the release 6.1 of miRTarBase that contains the experimentally validated microRNA-target interactions database. It has 322,389 unique miRNA-target gene interactions from 410,620 miRNA- target gene interactions. The second data set is miR2Disease and it has 637 unique miRNA-target gene interactions from 805 miRNA- target gene interactions. The integrated data consists of experimentally validated 322,994 unique miRNA-target gene interactions.

Value

mirnaGeneData returns miRNA-target gene interactions whose first column contains miRNAs and the second column contains target genes

References

Hsu, S.D., Tseng, Y.T., Shrestha, S., et al.: miRTarBase update 2014: an information resource for experimentally validated miRNA-target interactions, *Nucleic Acids Res.*, 2014, 42, (D1), pp. D78- D85
Jiang, Q., Wang, Y., Hao, Y., et al. miR2Disease: a manually curated database for microRNA deregulation in human disease, *Nucleic Acids Res.*, 2009, 37, (Suppl 1), pp. D98- D104

Examples

```
data(mirnaGeneData)
```

prostateData	<i>The prostate cancer related miRNA-target gene dataset</i>
--------------	--

Description

prostateData It returns The prostate cancer related miRNA-target gene dataset

Usage

```
data(prostateData)
```

Arguments

prostateData The prostate cancer related miRNA-target gene dataset

Details

This dataset is derived from miRTarBase and miR2Disease datasets by using "createMirnaList-fromDisease" function.

Value

prostateData returns the prostate cancer related miRNA-target gene dataset

References

Hsu, S.D., Tseng, Y.T., Shrestha, S., et al.: miRTarBase update 2014: an information resource for experimentally validated miRNA-target interactions, *Nucleic Acids Res.*, 2014, 42, (D1), pp. D78-D85
 Jiang, Q., Wang, Y., Hao, Y., et al. miR2Disease: a manually curated database for microRNA deregulation in human disease, *Nucleic Acids Res.*, 2009, 37, (Suppl 1), pp. D98- D104

Examples

```
data(prostateData)
```

visualizeNet	<i>Plot the target gene network</i>
--------------	-------------------------------------

Description

visualizeNet Plot the target network

Usage

```
visualizeNet(network)
```


Arguments

network It is the binary format network that has edge weights.

Value

visualizeNet It only plots the gene network and it returns void.

See Also

[buildNetworkbyARules](#), [inferNet2BinaryFormat](#), [createMirnaListfromDisease](#), [buildmiRNATargetTable](#)

Examples

```
data(prostateData)

temp <- buildNetworkbyARules(prostateData,0.3,0.5,supportType="rank", filtering="sc")
net <- temp$finalNetwork;

visualizeNet(net);
```

Index

*Topic **misc**

- buildmiRNATargetTable, [2](#)
- buildNetworkbyARules, [2](#)
- createMirnaListfromDisease, [3](#)
- disease, [4](#)
- diseaseMirnaData, [5](#)
- inferNet2BinaryFormat, [6](#)
- mirnaGeneData, [7](#)
- prostateData, [8](#)
- visualizeNet, [8](#)

- buildmiRNATargetTable, [2](#), [3](#), [4](#), [6](#), [9](#)
- buildNetworkbyARules, [2](#), [2](#), [4](#), [6](#), [9](#)
- createMirnaListfromDisease, [2](#), [3](#), [3](#), [6](#), [9](#)
- disease, [4](#)
- diseaseMirnaData, [5](#)
- inferNet2BinaryFormat, [2–4](#), [6](#), [9](#)
- mirnaGeneData, [7](#)
- prostateData, [8](#)
- visualizeNet, [2–4](#), [6](#), [8](#)