Package ‘CoDiNA’

March 6, 2019

Type Package
Title Co-Expression Differential Network Analysis
Version 1.1.1
Author Deisy Morselli Gysi, Tiago de Miranda Fragoso, Eivind Almaas and Katja Nowick.
Maintainer Deisy Morselli Gysi <deisy@bioinf.uni-leipzig.de>
Description Categorize links and nodes from multiple networks in 3 categories: Common links (alpha) specific links (gamma), and different links (beta). Also categorizes the links into subcategories and groups. The package includes a visualization tool for the networks. More information about the methodology can be found at: Gysi et. al., 2018 <arXiv:1802.00828>.
License GPL-2
LazyData TRUE
Depends R (>= 3.1)
Imports data.table, igraph, magrittr, plyr, visNetwork
RoxygenNote 6.1.1
Suggests knitr, rmarkdown, wTO
VignetteBuilder knitr
NeedsCompilation no
Repository CRAN
Date/Publication 2019-03-06 15:32:53 UTC

R topics documented:

  .Random.seed .......................................................... 2
  as.igraph ............................................................. 2
  AST ................................................................. 3
  ClusterNodes .......................................................... 4
  CTR ................................................................. 4
  GLI ................................................................. 5
  MakeDiffNet .......................................................... 6
  normalize ........................................................... 7
  OLI ................................................................. 7
### Description

Random numbers generated by set.seed(123)

### Usage

```r
diffnet <- makediffnet (data = list (net1, net2, net3), Code = c ("Net1", "Net2", "Net3"))
Graph <- plot (x = DiffNet,
               layout = NULL, smooth.edges = TRUE,
               path = NULL, MakeGroups = FALSE, Cluster = FALSE,
               ...)```

### Example

```r
suppressWarnings(RNGversion("3.5.0"))
Nodes = LETTERS[1:10]
Net1 = data.frame (Node.1 = sample (Nodes), Node.2 = sample (Nodes), wTO = runif (10, -1, 1))
Net2 = data.frame (Node.1 = sample (Nodes), Node.2 = sample (Nodes), wTO = runif (10, -1, 1))
Net3 = data.frame (Node.1 = sample (Nodes), Node.2 = sample (Nodes), wTO = runif (10, -1, 1))
DiffNet = MakeDiffNet (Data = list (Net1, Net2, Net3), Code = c ("Net1", "Net2", "Net3"))
Graph = plot (x = DiffNet,
              layout = NULL, smooth.edges = TRUE,
              path = NULL, MakeGroups = FALSE, Cluster = FALSE,
              ...)```
Description

This data.table contains node and the weighted topological overlap (wTO) of Transcription Factors (TFs), from GSE4290 (Sun, 2006) for 50 brain samples with ASTgodendroglomas. The wTO was calculated using the package wTO.

Usage

data("AST")

Format

A data frame with 3488761 observations on the following 3 variables.

Node.1  a factor with levels. TF names
Node.2  a factor with levels. TF names
cor    a numeric vector. wTO values calculated using only the TFs

Source


References


Examples

data(AST)
str(AST)
ClusterNodes

Description
Categorize the Nodes into Phi and Phi tilde.

Usage
ClusterNodes(DiffNet, cutoff.external = 0.8, cutoff.internal = 0.5)

Arguments
- DiffNet: The Differential network from MakeDiffNet
- cutoff.external: The cut-off between the clusters (delta from the center to the edge coordinates), the closer to 1, the better.
- cutoff.internal: The cut-off inside the clusters (delta from the theoretical cluster to the edge coordinates), the closer to zero, the better.

Examples
DiffNet = MakeDiffNet (Data = list(CTR, AST), Code = c('CTR', 'AST'))
Genes_Phi = ClusterNodes(DiffNet, cutoff.external = 0.5, cutoff.internal = 0.25)
table(Genesis_Phi$Phi_tilde)

CTR

Description
This data.table contains node and the weighted topological overlap (wTO) of Transcription Factors (TFs), from GSE4290 (Sun, 2006) for 23 brain samples with Controls. The wTO was calculated using the package wTO.

Usage
data("CTR")

Format
A data frame with 3488761 observations on the following 3 variables.
- Node.1: a factor with levels. TF names
- Node.2: a factor with levels. TF names
- cor: a numeric vector. wTO values calculated using only the TFs
Source


References


Examples

data(CTR)
str(CTR)

data(GLI)
str(GLI)

Description

This data.table contains node and the weighted topological overlap (wTO) of Transcription Factors (TFs), from GSE4290 (Sun, 2006) for 81 brain samples with glioblastomas. The wTO was calculated using the package wTO.

Usage

data("GLI")

Format

A data frame with 3488761 observations on the following 3 variables.

Node.1  a factor with levels. TF names
Node.2  a factor with levels. TF names
cor     a numeric vector. wTO values calculated using only the TFs

Source


References


Examples

data(GLI)
str(GLI)
Description

Categorize links into Phi categories, calculate the distance to the center and also normalize the distance into some categories: Phi and Phi tilda, group and all.

Usage

MakeDiffNet(Data, Code, cutoff = 0.33, stretch = TRUE)

Arguments

Data: List of data.frames containing Node.1, Node.2 and the correlation value.
Code: Name of each one of the networks.
cutoff: By default, the cutoff is 0.33. If the user wants to use another value, it has to be cited on the description of the used methodology that the cutoff was changed.
stretch: Should the input data be normalized? Default to TRUE.

Value

Returns a data.table containing: Nodes names, correlation value for each network (the input values), the k means cluster that link belongs, the Phi groups (Phi and Phi tilda), the signed group that link belongs to, the unsigned group. The distance to the center, and the distance normalized by: Phi_tilda, Phi, signed group or all data.

Author(s)

Deisy Morselli Gysi <deisy at bioinf.uni-leipzig.de>

Examples

suppressWarnings(RNGversion("3.5.0"))
Nodes = LETTERS[1:20]
Net1 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10, -1, 1))
Net2 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10, -1, 1))
Net3 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10, -1, 1))
DiffNet = MakeDiffNet (Data = list(Net1, Net2, Net3), Code = c('Net1', 'Net2', 'Net3'))
print(DiffNet)
**normalize**

Normalize a given variable.

**Usage**

```r
normalize(m)
```

**Arguments**

- `m`: variable to be normalized in the interval [0,1]

**Author(s)**

Deisy Morselli Gysi <deisy at bioinf.uni-leipzig.de>

**Examples**

```r
Z = runif(10,-10,10)
normalize(Z)
```

---

**OLI**

This data.table contains node and the weighted topological overlap (wTO) of Transcription Factors (TFs), from GSE4290 (Sun, 2006) for 50 brain samples with oligodendrogliomas. The wTO was calculated using the package wTO.

**Usage**

```r
data("OLI")
```

**Format**

A data frame with 3488761 observations on the following 3 variables.

- `Node.1`: a factor with levels. TF names
- `Node.2`: a factor with levels. TF names
- `cor`: a numeric vector. wTO values calculated using only the TFs
Source


References


Examples

data(OLI)
str(OLI)

<table>
<thead>
<tr>
<th>OrderNames</th>
<th>OrderNames</th>
</tr>
</thead>
</table>

Description

Sorts each link’s Nodes by the smallest value. Removes links that both nodes are the same.

Usage

OrderNames(M)

Arguments

M data.frame to have the names ordered. Node.1, Node.2 and correlation value.

Value

a data.table with Node.1 and Node.2, sorted by the smallest value between both.

Author(s)

Deisy Morselli Gysi <deisy@bioinf.uni-leipzig.de>

Examples

Nodes = LETTERS[1:10]
Z = data.frame(Node.1 = sample(Nodes),
Node.2 = sample(Nodes), cor = runif(10,-1,1))
OrderNames(Z)
Description

Categorize the Nodes into Phi and groups categories. Also, creates an interactive view of the CoDiNA network.

Usage

```r
## S3 method for class 'CoDiNA'
plot(x, cutoff.external = 0, cutoff.internal = 1,
cutoff.ratio = 1, layout = NULL, smooth.edges = TRUE,
path = NULL, MakeGroups = FALSE, Cluster = FALSE, legend = TRUE,
manipulation = FALSE, sort.by.Phi = FALSE, ...)
```

Arguments

- `x` Output from `MakeDiffNet`
- `cutoff.external` The cut-off between the clusters (delta from the center to the edge coordinates), the closer to 1, the better.
- `cutoff.internal` The cut-off inside the clusters (delta from the theoretical cluster to the edge coordinates), the closer to zero, the better.
- `cutoff.ratio` The cut-off for the ratio of both scores. Default is set to 1. The greater, the better.
- `layout` a layout from the igraph package.
- `smooth.edges` If the edges should be smoothed or not.
- `path` If the graph should be saved specify the name of the file.
- `MakeGroups` algorithm to find clusters. One of the followings: walktrap, optimal, spinglass, edge.betweenness, fast_greedy, infomap, louvain, label_prop, leading_eigen. Default to FALSE.
- `Cluster` TRUE or FALSE if the nodes should be clustered (double click to uncluster).
- `legend` TRUE or FALSE if the legend should appear.
- `manipulation` TRUE or FALSE if the graph should be editable.
- `sort.by.Phi` if the graph should be plotted in the Phi order
- `...` Additional plotting parameters.

Value

Returns a list containing: The nodes description, the Edges description and the network graph.
Author(s)
Deisy Morselli Gysi <deisy at bioinf.uni-leipzig.de>

Examples

suppressWarnings(RNGversion("3.5.0"))

Nodes = LETTERS[1:10]
Net1 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10,-1,1))
Net2 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10,-1,1))
Net3 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10,-1,1))
DiffNet = MakeDiffNet (Data = list(Net1,Net2,Net3), Code = c('Net1', 'Net2', 'Net3'))
Graph = plot(x = DiffNet,
  layout = NULL, smooth.edges = TRUE,
  path = NULL, MakeGroups = FALSE, Cluster = FALSE,
  legend = TRUE, manipulation = FALSE, sort.by.Phi = FALSE)
Graph

Description
Print on the screen the number of nodes and edges. To see the data.frame, call: data.frame().

Usage

## S3 method for class 'CoDiNA'
print(x, ...)

Arguments

x Output from MakeDiffNet
...

Additional plotting parameters.

Value
Print on the screen the number of nodes and edges.

Author(s)
Deisy Morselli Gysi <deisy at bioinf.uni-leipzig.de>
Examples

Nodes = LETTERS[1:10]
Net1 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10,-1,1))
Net2 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10,-1,1))
Net3 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10,-1,1))
DiffNet = MakeDiffNet (Data = list(Net1,Net2,Net3), Code = c('Net1', 'Net2', 'Net3'))
print(DiffNet)

summary.CoDiNA

Description

summary of the CoDiNA network.

Usage

## S3 method for class 'CoDiNA'
summary(object,...)

Arguments

object Output from MakeDiffNet
...

Additional plotting parameters.

Value

Returns a summary describing the network.

Author(s)

Deisy Morselli Gysi <deisy at bioinf.uni-leipzig.de>

Examples

Nodes = LETTERS[1:10]
Net1 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10,-1,1))
Net2 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10,-1,1))
Net3 = data.frame(Node.1 = sample(Nodes), Node.2 = sample(Nodes), wTO = runif(10,-1,1))
DiffNet = MakeDiffNet (Data = list(Net1,Net2,Net3), Code = c('Net1', 'Net2', 'Net3'))
summary(DiffNet)
Index

*Topic **datasets**
  Random.seed, 2
  AST, 3
  CTR, 4
  GLI, 5
  OLI, 7
  Random.seed, 2

as.igraph, 2
AST, 3

ClusterNodes, 4
CTR, 4

GLI, 5

MakeDiffNet, 6
normalize, 7

OLI, 7
OrderNames, 8

plot.CoDiNA, 9
print.CoDiNA, 10

summary.CoDiNA, 11