Package ‘NetComp’

February 19, 2015

Version 1.6

Date 2012-8-31

Title Network Generation and Comparison

Author Shannon M. Bell <`bell.shannonm@gmail.com`>, Lyle D. Burgoon <`burgoon.lyle@gmail.com`>

Maintainer Shannon M. Bell <`bell.shannonm@gmail.com`>

Depends R (>= 2.12.1)

Imports gdata

LazyLoad yes

Description This package contains functions to carry out high throughput data analysis and to conduct data set comparisons. Similarity matrices from high throughput phenotypic data containing uninformative (e.g. wild type) or missing data can be calculated to report similarity of response. A suite of graph comparisons using an adjacency or correlation matrix format are included to facilitate quick network analysis.

License GPL (>= 3)

Repository CRAN

Date/Publication 2012-08-31 17:17:51

NeedsCompilation yes

R topics documented:

`matrix_threshold` *Remove values from Matrix based on threshold*

Description

Returns matrix with values above or below the `threshold` are replaced. Can be used for converting to a binary matrix or for thresholding an adjacency matrix.

Usage

```
matrix_threshold(matrix, threshold=NULL, minval=0, maxval=NULL, abs=TRUE, rmna=FALSE,...)
```

Arguments

`matrix`	matrix
`threshold`	Integer
`minval`	Integer. Replaces values below threshold.
`maxval`	Integer. Replaces values above or equal to `threshold`.
`abs`	Logical. If TRUE the absolute value will be considered.
`rmna`	Logical. If TRUE NA's will be replaced with `minval`.
`...`	Other parameters.

Details

If `rm.na` is TRUE and `minval` is NULL, then NA's will be replaced with 0.

Value

`max_threshold` returns an matrix with values removed according to parameters.

Author(s)

Shannon M. Bell

Examples

```
 #using USArrest dataset
 arrestCor<-cor(t(USArrests))
arrestCor.7<-matrix_threshold(arrestCor, threshold=0.7, minval=0, abs=TRUE, rm.na=FALSE)
 arrestCor[10:15,1:5]
 arrestCor.7[10:15,1:5]
```

`netClass` *Network from Class List*

Description

Returns an unweighted adjacency matrix with a '1' where observations are from the same class, '0' if from different classes based on a vector of class assingments.

Usage

```
netClass(x, labels=NULL)
```

Arguments

`x`	Numeric vector containing the class/cluster assignments
`labels`	Optional. Vector of labels for the individual observations

Details

Used for converting class/community assingments into an adjacency matrix.

Value

A square matrix is returned with the number of rows, columns equal to the length of `x` and in the same order. If `lables` is supplied, they will be used as the row/column names.

Author(s)

Shannon M. Bell

See Also

`hclust{cutree}`

Examples

```
#using the USArrest dataset
arrestCor<-cor(t(USArrests))
tree<-cutree(hclust(as.dist(1-arrestCor), method='ward'), k=10)
netClass(tree, labels=colnames(arrestCor))
```

`netDiff` *Network Difference*

Description

Returns the difference of the input matrices. Output object is an adjacency matrix where edges present in `matrix1` but not `matrix2` are returned. The edge value is maintained provided `cutoff` = NULL. Edges in both graphs failing to meet the `cutoff`, if provided, are set to zero before taking the graph difference.

Usage

```
netDiff(matrix1, matrix2, cutoff=NULL, ...)
```

Arguments

`matrix1`	Square matrix (e.g. correlation or adjacency) containing row/column labels
`matrix2`	Square matrix (e.g. correlation or adjacency) containing row/column labels
`cutoff`	The cutoff value. Edges less than this value (absolute value considered)are converted to zero.
`...`	Other parameters.

Details

Matrices must be square and have row and column labels. Output adjacency matrix can be used directly for creating a graph object.

Value

`netDiff` returns an adjacency matrix containing edges present in `matrix1` that are not present in `matrix2`. Edges below `cutoff` are set to zero.

Author(s)

Shannon M. Bell

Examples

```
#using the state.x77 and USArrest datasets
#remove data from states for illustration
ssArrest<-subset(t(USArrests), select=-c(Alabama,Colorado,Delaware))
ssState<-subset(t(state.x77), select=-c(Alabama, Arizona, Iowa))
arrestCor<-cor(ssArrest)
stateCor<-cor(ssState)
dataDiff<-netDiff(stateCor, arrestCor)
dataDiff[1:15,1:5]
#Setting a cutoff to remove any edges that are below 0.6
dataDiff.6<-netDiff(stateCor, arrestCor, cutoff=0.6)
dataDiff.6[1:15,1:5]
```

`netIntersect` *Network Intersection*

Description

Returns the intersection of the input matrices. Output object is an adjacency matrix where the edge weight is the average value of the absolute value of edges in BOTH matrices. Edges not present in either, or below the `cutoff` are removed or set to zero.

Usage

```
netIntersect(matrix1, matrix2, cutoff=NULL, absolute=TRUE, ...)
```

Arguments

`matrix1`	Square matrix (e.g. correlation or adjacency) containing row/column labels
`matrix2`	Square matrix (e.g. correlation or adjacency) containing row/column labels
`cutoff`	The cutoff value. Edges less than this value (absolute value considered)are converted to zero.
`absolute`	Logical, should the absolute values be considered for the cutoff value.
`...`	Other parameters.

Details

Matrices must be square and have row and column labels. Output adjacency matrix can be used directly for creating a graph object.

Value

`netIntersect` returns an adjacency matrix containing edged present in both graphs.

Author(s)

Shannon M. Bell

Examples

```
#using the state.x77 and USArrests datasets
arrestCor<-cor(t(USArrests))
stateCor<-cor(t(state.x77))
isect<-netIntersect(stateCor, arrestCor)
isect[1:15,1:5]
#Setting a cutoff to remove any edges that are below 0.6
isect.6<-netIntersect(stateCor, arrestCor, cutoff=0.6)
isect.6[1:15,1:5]
```

`netUnion` *Network Union*

Description

Returns the union of the input matrices. Output object is an adjacency matrix where the edge weight is 1 if an edge is present in both graphs, 0.5 if present in 1 graph, and 0 if present in neither graph. Edges in both graphs failing to meet the `cutoff`, if provided, are set to zero before performing the graph union.

Usage

```
netUnion(matrix1, matrix2, cutoff=NULL, ...)
```

Arguments

`matrix1`	Square matrix (e.g. correlation or adjacency) containing row/column labels
`matrix2`	Square matrix (e.g. correlation or adjacency) containing row/column labels
`cutoff`	The cutoff value. Edges less than this value (absolute value considered)are converted to zero.
`...`	Other parameters.

Details

Matrices must be square and have row and column labels. Output adjacency matrix can be used directly for creating a graph object.

Value

`netUnion` returns an adjacency matrix containing edges present either both graphs.

Author(s)

Shannon M. Bell

Examples

```
#using the state.x77 and USArrest datasets
#remove data from states for illustration
ssArrest<-subset(t(USArrests), select=-c(Alabama,Colorado,Delaware))
ssState<-subset(t(state.x77), select=-c(Alabama, Arizona, Iowa))
arrestCor<-cor(ssArrest)
stateCor<-cor(ssState)
dataUnion<-netUnion(stateCor, arrestCor)
dataUnion[1:15,1:5]
#Setting a cutoff to remove any edges that are below 0.6
dataUnion.6<-netUnion(stateCor, arrestCor, cutoff=0.6)
dataUnion.6[1:15,1:5]
```

`netVal` *Network Validation*

Description

Returns vector with the True Positives, True Negatives, False Postives, False Negatives, the Adjusted Rand Index, and the Balanced Accuracy of the test network (`matrix2`) compared to the true network (`matrix1`). Network communities are based on hierachical clustering, with communities defind by `k` or `h`.

Usage

```
netVal(matrix1, matrix2, method='ward', k=200, h=NULL,...)
```

Arguments

`matrix1`	Adjacency matrix containing the true topology
`matrix2`	Adjacency matrix containing the test topology
`method`	Hierarchical clustering method to be passed to `hclust`.
`k`	Number of groups. Passed to `cutree`
`h`	Heights where tree is to be cut. Passed to `cutree`
`...`	Other parameters.

Details

The input matrices must be square but do not need to contain the same edges.

Value

`netVal` returns an vector with values True Positives, True Negatives, False Postives, False Negatives, Adjusted Rand Index, and Balanced Accuracy.

Author(s)

Shannon M. Bell

See Also

`hclust{cutree}`

Examples

```
#using the state.x77 and USArrest datasets
arrestCor<-cor(t(USArrests))
stateCor<-cor(t(state.x77))
netVal(stateCor, arrestCor, method='ward', k=10)
netVal(stateCor, stateCor, method='ward', k=10)
```

`SimMeasure` *Generation of Similarity Matrix from Sparse/ Low Information Content Data*

Description

Returns a adjacency matrix with the similarity scores between individuals. If `threshold` is provided, values where the absolute value of the observation is less than the threshold are not considered. The similarity measure is based on the percent difference between the observations. Details of the algorithm can be found in the accompanying paper (see references).

Usage

```
SimMeasure(data,  threshold=NULL, ...)
```

Arguments

`data`	Matrix object containing observation data on which to calculate the similarity score.
`threshold`	The threshold value. Responses less than this value (absolute value considered)are not used in calculating the similarity score.
`...`	Other parameters.

Details

Data can contain NA but may not contain NULL values. This method is designed for datasets with high numbers of missing or uninformative values that can be removed by setting the `threshold` value. Note that the `threshold` value must be the same for all numbers.

Value

`SimMeasure` returns an adjacency matrix containing edges corresponding to the similarity of the observed values.

Author(s)

Shannon M. Bell

Examples

```
#using the state.x77
sim<-SimMeasure(t(state.x77), threshold=NULL)
sim[1:5,1:15]
```

Index