Package ‘gsplom’

April 8, 2014

Type Package

Title Glyph Scatterplot Matrix (Glyph SPLOM)

Version 0.99.0

Author Andrew D. Yates

Maintainer Andrew D. Yates <yates.115@osu.edu>

Description Glyph SPLOM is a data visualization and exploration tool like an enhanced correlation heatmap that can be used to infer directed networks from unordered, numeric data. It combines a non-linear measure of dependency strength, distance correlation, and a four-quadrant dependency class to quantify dependencies in an all-pairs dependency matrix. gsplom provides tools to efficiently compute, visualize, cluster, and analyze this matrix.

License Artistic-2.0

Depends R (>= 2.14.0)

Suggests RUnit, BiocGenerics, fastcluster

URL https://github.com/andrewdyates/gsplom.rpackage

R topics documented:

`adaptiveRegress` *High/Low binary partition by adaptive regression.*

Description

Sort values from low to high, fit a step-up function, and return a partition. Removes `NA` values before computation.

Usage

```
adaptiveRegress(v)
```

Arguments

`v`	A numeric vector to quantize.

Value

A list containing two elements:

`thresh`: high/low threshold value

`k`: highest rank of low partition

Author(s)

Andrew D. Yates

Examples

```
v <- sample(1:100)
R <- adaptiveRegress(v)
```

`dcorMatrix` *Compute Distance Correlation Matrix.*

Description

Efficiently compute an all-pairs-rows distance correlation matrix from a data matrix.

Usage

```
dcorMatrix(M, verbose = TRUE)
```

Arguments

`M`	A numeric matrix where rows are variables and columns are samples with no missing values (`NA`).
`verbose`	Whether to output status information as the result is computed. Default is TRUE.

Value

A numeric matrix where the entry at row-i, column-j is the distance correlation between row-i and row-j in `M`. Rows and columns are labeled by the row labels of `M`.

Note

Uses absolute difference (Euclidan Distance in one dimension) as a distance function in the distance correlation computation. Does not support missing values (`NA`); see `dcorMatrixNA`.

Author(s)

Andrew D. Yates

Examples

```
## Load data so that samples (countries) are columns and econometric
##   variables are rows.
M <- t(LifeCycleSavings)
## Compute all pairs distance correlation between econometric variables.
DCOR <- dcorMatrix(M)
```

`dcorMatrixNA` *Compute Distance Correlation Matrix with Missing Values in Data.*

Description

A slower method of computing all-pairs-rows distance correlation that excludes samples with at least one missing value in each computation.

Usage

```
dcorMatrixNA(M, do.rank = FALSE, verbose = TRUE)
```

Arguments

`M`	A numeric matrix where rows are variables and columns are samples. May have missing values (`NA`).
`do.rank`	Whether to convert values to ranks after removing samples with missing values. Default is FALSE.
`verbose`	Whether to output status information as the result is computed. Default is TRUE.

Value

A list containing two matrices:

`DCOR`: A numeric matrix where the entry at row-i, column-j is the distance correlation between row-i and row-j in `M` after removing samples where there is at least one missing value in either row-i or row-j. If only one or zero samples remain after removing missing values, the entry value is `NA`. Rows and columns are labeled by the row labels of `M`.

`SIZE`: An integer matrix where the entry at row-i, column-j is the number of samples used to compute DCOR[i,j] after handling missing values.

Note

Uses absolute difference (Euclidan Distance in one dimension) as a distance function in the distance correlation computation. This function is much less efficient than `dcorMatrix` and should only be used when missing values are in the data.

Author(s)

Andrew D. Yates

Examples

```
## Load data so that samples (countries) are columns and econometric
##   variables are rows.
M <- t(LifeCycleSavings)
## Insert a missing value.
M[1,1] <- NA
## Compute all-pairs distance correlation between econometric variables.
Results <- dcorMatrixNA(M)
```

---

dcorSingle *Single Pair Distance Correlation.*

Description

Compute Distance Correlation between two 1-dimensional vectors.

Usage

```
dcorSingle(x, y)
```

Arguments

x	A numeric vector.
y	A numeric vector of the same length as `x`.

Value

Numeric value between 0 and 1 of the distance correlation between `x` and `y`. Returns `NA` if distance correlation is undefined.

Note

Uses absolute difference (Euclidan Distance in one dimension) as a distance function in the distance correlation computation. Does not support missing values; remove samples with at least one missing value from `x` and `y` prior to using `dcorSingle`.

Author(s)

Andrew D. Yates

Examples

```
## Load data so that samples (countries) are columns and econometric
##   variables are rows.
M <- t(LifeCycleSavings)
## Compute distance correlation between two econometric variables.
d <- dcorSingle(M[1,], M[2,])
```

`gsplom` *Glyph SPLOM Workflow Wrapper*

Description

Generate a Glyph SPLOM from a data matrix.

Usage

```
gsplom(M, ...)
```

Arguments

`M`	A numeric matrix where rows are variables and columns are samples.
`...`	Special plotting options.

Value

A list of intermediate results and the gsplom plot itself sent to the current open plotting device.

Note

This is a high level function that completes the entire Glyph SPLOM workflow on a data matrix using the recommended default values. See `gsplomCore` to plot a Glyph SPLOM from a pre-computed all-pairs distance correlation matrix and logical dependency class matrix. Other parameters are related to special Glyph SPLOM options.

Author(s)

Andrew D. Yates

Examples

```
## Not run:
## Load data so that samples (countries) are columns and econometric
##   variables are rows.
M <- t(LifeCycleSavings)
pdf("/mypath/gsplom.pdf")
R <- gsplom(M)
dev.off()

## End(Not run)
```

`logicClassMatrix` *Compute Logical Dependency Class Matrix.*

Description

Efficiently compute an all-pairs-rows logical dependency class matrix from a data matrix.

Usage

```
logicClassMatrix(M, thresholds=NULL, intervals=NULL, z=3, min.cnt=0, frac.conf=0.2)
```

Arguments

`M`	A numeric matrix where rows are variables and columns are samples.
`thresholds`	A vector of high/low thresholds corresponding to each row of `M`. Default is NULL. If NULL, then thresholds are computed automatically using `adaptiveRegress`.
`intervals`	A vector of uncertainty intervals corresponding to each row of `M`. Default is NULL. If NULL, then intervals are uniformly set to the 3rd percentile row standard deviation.
`z`	A parameter of how sparse a quadrant must be relative to the margins to be called as empty. Higher z means that a quadrant must be more relatively sparse to be called as sparse. Default is 3.
`min.cnt`	A parameter to force a quadrant to be called as sparse if a fewer or equal number of samples are in it.
`frac.conf`	A parameter of what fraction of points must be in a quadrant and not in the uncertainty region to not classifiy as logical dependency class *NA*.

Value

A numeric matrix where the entry at row-i, column-j is an integer enumeration of a logical dependency class between row-i and row-j in `M`. Rows and columns are labeled by the row labels of `M`.

Note

Supports missing values. Missing values in either variable per pair are called to be in the uncertainty region and ignored during classification.

Author(s)

Andrew D. Yates

Examples

```
M <- t(LifeCycleSavings)
CLS <- logicClassMatrix(M, intervals=apply(M,1,sd)/2, frac.conf=0.2)
M[1,1] <- NA
CLS.NA <- logicClassMatrix(M, intervals=apply(M,1,function(x)sd(x,na.rm=TRUE))/2, frac.conf=0.2)
```

Index