Package ‘discordant’

December 7, 2019

Version 1.10.0
Date 2016-10-21
Title The Discordant Method: A Novel Approach for Differential Correlation
Author Charlotte Siska [cre,aut], Katerina Kechris [aut]
Depends R (>= 3.4)
Maintainer Charlotte Siska <siska.charlotte@gmail.com>
Description Discordant is a method to determine differential correlation of molecular feature pairs from -omics data using mixture models. Algorithm is explained further in Siska et al.
Encoding latin1
biocViews ImmunoOncology, BiologicalQuestion, StatisticalMethod, mRNAMicroarray, Microarray, Genetics, RNASeq
Suggests BiocStyle, knitr
Imports Biobase, stats, biwt, gtools, MASS, tools
License GPL (>= 2)
URL https://github.com/siskac/discordant
NeedsCompilation yes
VignetteBuilder knitr
git_url https://git.bioconductor.org/packages/discordant
git_branch RELEASE_3_10
git_last_commit 11e6f2c
git_last_commit_date 2019-10-29
Date/Publication 2019-12-06

R topics documented:

createVectors .......................................................... 2
discordantRun .......................................................... 3
fishersTrans ............................................................ 5
splitMADOutlier .......................................................... 5
TCGA_Breast_miRNASeq ................................................. 6
TCGA_Breast_miRNASeq_voom ......................................... 7
createVectors

TCGA_Breast_RNASeq ......................................................... 8
TCGA_Breast_RNASeq_voom .................................................... 8
TCGA_GBM_miRNA_microarray .................................................. 9
TCGA_GBM_transcript_microarray ............................................. 10

createVectors

Create Pearson’s correlation coefficient vectors based on bivariate data

Description

Calculates correlation coefficients based on two groups of omics bivariate data. Currently, only two groups of samples can be specified. Used to make input for discordantRun().

Usage

createVectors(x, y = NULL, groups, cor.method = c("spearman"))

Arguments

x  ExpressionSet of -omics data
y  optional second ExpressionSet of -omics data, induces dual -omics analysis
groups  n-length vector of 1s and 2s matching samples belonging to groups 1 and 2
cor.method  correlation method to measure association. Options are "spearman", "pearson", "bwmc" and "sparcc"

Details

Creates vectors of correlation coefficients based on feature pairs within x or between x and y. The names of the vectors are the feature pairs taken from x and y.

Value

v1  List of correlation coefficients for group 1
v2  List of correlation coefficients for group 2

Author(s)

Charlotte Siska <siska.charlotte@gmail.com>

References

Examples

```r
## load data
data("TCGA_GBM_miRNA_microarray") # loads matrix called TCGA_GBM_miRNA_microarray
data("TCGA_GBM_transcript_microarray") # loads matrix called TCGA_GBM_transcript_microarray
print(colnames(TCGA_GBM_transcript_microarray)) # look at groups

groups <- c(rep(1,10), rep(2,20))

# transcript-transcript pairs
vectors <- createVectors(TCGA_GBM_transcript_microarray, groups = groups, cor.method = c("pearson"))

# miRNA-transcript pairs
vectors <- createVectors(TCGA_GBM_transcript_microarray, TCGA_GBM_miRNA_microarray, groups = groups)
```

discordantRun

Run Discordant Algorithm

Description

Runs discordant algorithm on two vectors of correlation coefficients.

Usage

discordantRun(v1, v2, x, y = NULL, transform = TRUE, subsampling = FALSE, subSize = dim(x)[1], iter = 100, components = 3)

Arguments

v1 Vector of Pearson correlation coefficients in group 1
v2 Vector of Pearson correlation coefficients in group 2
x ExpressionSet of -omics data
y ExpressionSet of -omics data, induces dual -omics analysis
transform If TRUE v1 and v2 will be Fisher transformed
subsampling If TRUE subsampling will be run
subSize Indicates how many feature pairs to be used for subsampling. Default is the feature size in x
iter Number of iterations for subsampling. Default is 100
components Number of components in mixture model.

Details

The discordant algorithm is based on a Gaussian mixture model. If there are three components, correlation coefficients are clustered into negative correlations (-), positive correlations (+) and no correlation (0). If there are five components, then there are two more classes for very negative correlation (--) and very positive correlations (++) . All possible combinations for these components are made into classes. If there are three components, there are 9 classes. If there are five components, there are 25 classes.
The posterior probabilities for each class are generated and outputted into the value probMatrix. The value probMatrix is a matrix where each column is a class and each row is a feature pair. The values discordPPVector and discordPPMatrix are the summed differential correlation posterior probability for each feature pair. The values classVector and classMatrix are the class with the highest posterior probability for each feature pair.

**Value**

- **discordPPVector**: Vector of differentially correlated posterior probabilities.
- **discordPPMatrix**: Matrix of differentially correlated posterior probabilities where rows and columns reflect features.
- **classVector**: Vector of classes that have the highest posterior probability.
- **classMatrix**: Matrix of classes that have the highest posterior probability where rows and columns reflect features.
- **probMatrix**: Matrix of posterior probabilities where rows are each molecular feature pair and columns are nine different classes.
- **loglik**: Final log likelihood.

**Author(s)**

Charlotte Siska <siska.charlotte@gmail.com>

**References**


**Examples**

```r
## load Data

data(TCGA_GBM_miRNA_microarray) # loads matrix called TCGA_GBM_miRNA_microarray
data(TCGA_GBM_transcript_microarray) # loads matrix called TCGA_GBM_transcript_microarray
print(colnames(TCGA_GBM_transcript_microarray)) # look at groups
groups <- c(rep(1,10), rep(2,20))

## DC analysis on only transcripts pairs

vectors <- createVectors(TCGA_GBM_transcript_microarray, groups = groups)
result <- discordantRun(vectors$v1, vectors$v2, TCGA_GBM_transcript_microarray)

## DC analysis on miRNA-transcript pairs

vectors <- createVectors(TCGA_GBM_transcript_microarray, TCGA_GBM_miRNA_microarray, groups = groups, cor.method = c("pearson"))
result <- discordantRun(vectors$v1, vectors$v2, TCGA_GBM_transcript_microarray, TCGA_GBM_miRNA_microarray)
```
fishersTrans

Fisher Transformation of Pearson Correlation Coefficients to Z Scores

Description
Transforms Pearson's correlation coefficients into z scores using Fisher's method.

Usage
fishersTrans(rho)

Arguments
rho Integer or numeric vector of Pearson's correlation coefficients

Details
Fisher's transformation is when correlation coefficients are transformed into a z score. These z scores have an approximately normal distribution.

Value
Returns Fisher-transformed correlation coefficients

References
Fisher, R.A. (1915). "Frequency distribution of the values of the correlation coefficient in samples of an indefinitely large population". Biometrika (Biometrika Trust) 10 (4).

Examples
## Create integer or list of Pearson's correlation coefficients.
library(MASS)
rhoV <- as.vector(cor(t(mvrnorm(10, rep(3, 100), diag(100)))))
## Determine Fisher-Transformed z scores of rho
zV <- fishersTrans(rhoV)

splitMADOutlier

Outliers using left and right MAD

Description
Identify features with outliers using left and right median absolute deviation (MAD).

Usage
splitMADOutlier(mat, filter0 = TRUE, threshold = 2)
Arguments

mat  
mxn matrix of -omics data, where rows are features and columns samples.

filter0  
Option to filter out features if they have at least one 0 value. Default is TRUE.

threshold  
Threshold of how many MADs outside the left or right median is used to determine features with outliers.

Details

The purpose of this function is to determine outliers in non-symmetric distributions. The distribution is split by the median. Outliers are identified by being however many median absolute deviations (MAD) from either split distribution.

Value

mat.filtered  
Input matrix where features with outliers filtered out.

index  
Index of features that have no outliers.

References


Examples

```r
# Simulate matrix of continuous -omics data.
data(TCGA_Breast_miRNASeq)

# Filter matrix based on outliers.
mat.filtered <- splitMADOutlier(TCGA_Breast_miRNASeq)$mat.filtered
```

TCGA_Breast.miRNASeq  
TCGA Breast Cancer miRNASeq Sample Dataset

Description

This dataset contains TMM normalized miRNA count values from miRNASeq that was taken from the Cancer Genome Atlas, or TCGA. The dataset has 100 miRNA and 57 samples. The original dataset has 212 miRNA and 57 samples.

Usage

TCGA_Breast.miRNASeq

Format

A matrix of miRNA count values
Value

Breast miRNA-Seq count data with 100 features and 57 samples.

Author(s)

Charlotte Siska <siska.charlotte@gmail.com>

References


---

**TCGA Breast miRNASeq_voom**

**TCGA Breast Cancer miRNASeq Sample Dataset**

Description

This dataset contains TMM normalized voom-transformed miRNA count values from miRNASeq that was taken from the Cancer Genome Atlas, or TCGA. The dataset has 100 miRNA and 57 samples. The original dataset has 212 miRNA and 57 samples.

Usage

TCGA_Breast_miRNASeq_voom

Format

A matrix of miRNA count values

Value

Breast miRNA-Seq voom-transformed count data with 100 features and 57 samples.

Author(s)

Charlotte Siska <siska.charlotte@gmail.com>

References

<table>
<thead>
<tr>
<th>TCGA_Breast_RNASeq</th>
<th>TCGA Breast Cancer RNASeq Sample Dataset</th>
</tr>
</thead>
</table>

**Description**

This dataset contains TMM normalized RNA count values from RNASeq that was taken from the Cancer Genome Atlas, or TCGA. It has 100 features and 57 samples. The original dataset had 17972 features and 57 samples.

**Usage**

TCGA_Breast_RNASeq

**Format**

A matrix of RNA count values

**Value**

Breast RNA-Seq count data with 100 features and 57 samples.

**Author(s)**

Charlotte Siska <siska.charlotte@gmail.com>

**References**


<table>
<thead>
<tr>
<th>TCGA_Breast_RNASeq_voom</th>
<th>TCGA Breast Cancer RNASeq Sample Dataset</th>
</tr>
</thead>
</table>

**Description**

This dataset contains TMM normalized voom-transformed RNA count values from RNASeq that was taken from the Cancer Genome Atlas, or TCGA.

**Usage**

TCGA_Breast_RNASeq_voom

**Format**

A matrix of RNA count values

**Value**

Breast RNA-Seq voom-transformed count data with 100 features and 57 samples.
Description

This dataset contains miRNA expression values from a microarray that was taken from the Cancer Genome Atlas, or TCGA. It has 10 features and 30 samples. The original dataset had 331 features and 30 samples.

Usage

TCGA_GBM_miRNASample

Format

A matrix of miRNA expression values

Value

GBM miRNA microarray data with 10 features and 30 samples.

Author(s)

Charlotte Siska <siska.charlotte@gmail.com>

References

TCGA_GBM_transcript_microarray

TCGA Glioblastoma Multiforme Transcript Sample Dataset

Description

This dataset contains transcript expression values from a microarray that was taken from the Cancer Genome Atlas, or TCGA. It has 10 features and 30 samples. The original dataset had 72656 features and 30 samples.

Usage

TCGA_GBM_transcript_microarray

Format

A matrix of transcript expression values

Value

GBM transcript microarray data with 10 features and 30 samples.

Author(s)

Charlotte Siska <siska.charlotte@gmail.com>

References

Index

*Topic datagen
  createVectors, 2

*Topic datasets
  TCGA_Breast_miRNASeq, 6
  TCGA_Breast_miRNASeq_voom, 7
  TCGA_Breast_RNASeq, 8
  TCGA_Breast_RNASeq_voom, 8
  TCGA_GBM_miRNA_microarray, 9
  TCGA_GBM_transcript_microarray, 10

*Topic methods
  fishersTrans, 5
  splitMADOutlier, 5

*Topic model
  discordantRun, 3
  createVectors, 2
  discordantRun, 3
  fishersTrans, 5
  splitMADOutlier, 5
  TCGA_Breast_miRNASeq, 6
  TCGA_Breast_miRNASeq_voom, 7
  TCGA_Breast_RNASeq, 8
  TCGA_Breast_RNASeq_voom, 8
  TCGA_GBM_miRNA_microarray, 9
  TCGA_GBM_transcript_microarray, 10