Package 'BFDCA'

June 28, 2016

Type Package

Title A Comprehensive Tool of Using Bayes Factor for Differential Co-expression Analysis

Version 1.0

Date 2015-10-18

Author Duolin Wang

Maintainer Duolin Wang <deepduoduo@gmail.com>

Description BFDCA is a comprehensive tool of using Bayes factor for Differential Co-expression (DC) analysis. BFDCA contains three main functions: (1) clustering condition-specific genes into functional DC subunits; (2) quantitatively characterizing the regulatory impact of genes based on their differential connectivity within DC structures; and (3) providing a DC based prediction model to predict case/control conditions/phenotypes by taking DC significant gene pairs as markers.

License GPL (>= 2)

Imports Rcpp (>= 0.12.0), dynamicTreeCut (>= 1.62), flashClust,WGCNA,igraph

LinkingTo Rcpp

Depends R (>= 3.0),Matrix

NeedsCompilation yes

R topics documented:

`ALL` *ALL dataset.*

Description

Acute Lymphoblastic Leukemia (ALL) dataset. The ALL dataset was obtained from the ALL package [1] in Bioconductor [2], which uses microarray platform HG-U95Av2 with Affymetrix gene identifiers. Our test focused on the B-cell ALL, which consists of 37 samples with BCR/ABL mutation and 42 samples with no cytogenetic abnormalities. Columns correspond to genes and rows to samples.

Usage

```
data(ALL)
```

Format

A list contains class information and expression. ALL$class is a vector containing class information, with "1" indicates BCR/ABL mutation and "2" indicates no cytogenetic abnormalities. ALL$data is a data frame containing expression data. It consist of 79 samples and 8638 genes.

References

[1] Chiaretti S, Li X, Gentleman R, Vitale A, Vignetti M, Mandelli F, et al. Gene expression profile of adult T-cell acute lymphocytic leukemia identifies distinct subsets of patients with different response to therapy and survival. Blood. 2004;103:2771-8.

[2] Gentleman RC, Carey VJ, Bates DM, Bolstad B, Dettling M, Dudoit S, et al. Bioconductor: open software development for computational biology and bioinformatics. Genome Biol. 2004;5:R80.

Examples

```
data(ALL)
```

`BFplot` *Plot DC gene pairs*

Description

Plot gene expression patterns of DC gene pairs.

Usage

```
BFplot(dataExp,
       class,
       classlabel,
       gene1,
       gene2
       )
```

Arguments

dataExp	a data frame or matrix containing expression data. Columns correspond to genes and rows to samples. Such as in Compute_bf.
class	a numeric or character vector contains the corresponding class information of dataExp. By far, package only accepts binary classes.
classlabel	a numeric or character vector contains only two values. The first used to label class 1 and the second used to label class 2. The order of these two elements must be kept the same throughout the analyses.
gene1	a numeric or character indicates gene indexes or gene ids for gene1
gene2	a numeric or character indicates gene indexes or gene ids for gene2. gene1 and gene2 are end-nodes which form one DC gene pair.

Details

The function is used to plot gene expression patterns of DC gene pairs. It will generate a dot plot for the DC gene pairs with x-axis and y-axis represent gene expression levels for each gene in the DC gene pair. Dots in different colors and shapes represent samples from different classes. Circles in different colors represent the 95% contours of the estimated bivariate normal density from different classes.

BFplot2files *Plot multiple DC gene pairs into a file in pdf format*

Description

The function takes multiple DC gene pairs from a specified file as input, and for each pair, it plot one figure to show the gene expression patterns of the gene pair and merge these figures into a pdf file.

Usage

```
BFplot2files(dataExp,
       class,
       classlabel,
       edgefilename,
       plotfilename="Plotedges.pdf"
       )
```

Arguments

dataExp	a data frame or matrix containing expression data. Columns correspond to genes and rows to samples. Such as in Compute_bf.
class	a numeric or character vector contains the corresponding class information of dataExp. By far, package only accepts binary classes.
classlabel	a numeric or character vector contains only two values. The first used to label class 1 and the second used to label class 2. The order of these two elements must be kept the same throughout the analyses.

`edgefilename` an input file giving the DC gene pairs which are required to be plotted. Consists of at least two columns, the first column giving ids (or indexes) for gene1 and the second column giving ids (or indexes) for gene2. gene1 and gene2 are end-nodes which form one DC gene pair. Columns are splited by 'Tab'.

`plotfilename` specify a file (in pdf format) to output the multiple plots for all the gene pairs. The default file is "Plotedges.pdf".

Details

The function takes multiple DC gene pairs from a file specified by `edgefilename` as input, and for each pair, it plot one figure to show the gene expression patterns of the gene pair. Refer to BFplot for details.

BFtest *Predict case/control conditions or phenotypes for testing data.*

Description

Predict case/control conditions or phenotypes for a user provided testing data using the DC-based prediction model trained previously on a set of training data.

Usage

```
BFtest(testdata,
       model)
```

Arguments

`testdata` a data frame or matrix containing expression of testing data. Columns correspond to genes and rows to samples. The columns must be consistent with the columns in the training data.

`model` DCmodel object represents the prediction model trained by training data.

Details

The function predict case/control conditions or phenotypes for a user provided testing data using the DC-based prediction model trained previously on a set of training data.

Value

A vector contains the predicting class labels for the testing data.

BFtrain *Train a DC-based prediction model*

Description

Train a DC-based prediction model on training data.

Usage

```
BFtrain(dataExp,
        class,
        classlabel,
        edges,
        bfthr=6)
```

Arguments

dataExp	a data frame or matrix containing expression of training data. Columns correspond to genes and rows to samples. Such as in Compute_bf.
class	a numeric or character vector contains the corresponding class information of dataExp. By far, package only accepts binary classes.
classlabel	a numeric or character vector contains only two values. The first used to label class 1 and the second used to label class 2. the order of these two elements must be kept the same throughout the analyses.
edges	a data frame or a matrix containing user specified gene pairs, which are the significant DC gene pairs selected by sigDCpair_st1 and sigDCpair_SFS. It consists of at least two columns, the first column indicates ids (or indexes) for gene1, the second column indicates ids (or indexes) for gene2. Ids or indexes of genes must exist in dataExp.
bfthr	Bayes factor threshold. Only gene pairs with Bayes factor higher than bfthr threshold can be considered in the prediction model. The default value is 6.

Details

The function trains a DC-based prediction model on training data by using the significant DC gene pairs selected by sigDCpair_st1 and sigDCpair_SFS as features. It generates a model which contains all the estimated parameters for each gene pair under different classes.

Value

A DCmodel object indicates the DC-based prediction model containing all the estimated parameters for each gene pair under different classes.

`BF_output_networks` *Generate DC networks*

Description

Generate differential co-expression modules and information of gene-gene interactions.

Usage

```
BF_output_networks(
  dataExp,
  class,
  classlabel,
  BFobt,
  mst2file="MST2.txt",
  bfthr=6,
  corthres=0.3,
  echo=FALSE)
```

Arguments

`dataExp`	a data frame or matrix containing expression data. Columns correspond to genes and rows to samples. Such as in `Compute_bf`.
`class`	a numeric or character vector contains the corresponding class information of dataExp. By far, package only accepts binary classes.
`classlabel`	a numeric or character vector contains only two values. The first used to label class 1 and the second used to label class 2. the order of these two elements must be kept the same throughout the analyses.
`BFobt`	a `BFobt` object generated by `BF_WGCNA`.
`mst2file`	specify a file to output the generated union of First and Second Minimal Spanning tree (MST2). The default file name is "MST2.txt".
`bfthr`	Bayes factor threshold. Only gene pairs with Bayes factor higher than `bfthr` threshold can be considered. The default value is 6.
`corthres`	Pearson Correlation Coefficient (PCC) threshold. It's used to classify edges into up, down and unchange classes. Edges with PCC between -corthres to +corthres is labeled as unchange, edges with PCC higher than corthres is labeled as up, and edges with PCC smaller than corthres is labeled as down.
`echo`	a logical variable, indicates whether print processing information or not. If `TRUE`, it will print processing information for every 1000 gene pairs.

Details

The function returns a list containing (1) information for DC gene nodes, including module assignment and weight assignment. (2) information for DC gene pairs, including a full network built based on DC modules. And the essential links among genes represented by a union of First and Second Minimal Spanning Tree (MST2) will be outputed into a file specified by argument mst2file.

Value

A list containing `genegroups` and `network`.

`genegroups` containing following 4 columns:

1. gene: A character vector giving gene ids.
2. colors: A character vector giving color labels of DC modules.
3. labels: A numerical vector giving labels of DC modules, start from 0.
4. gene_weight: A numerical vector giving weight for each gene.

`network` containing following 16 columns:

1. geneid1: A character vector giving ids for gene1.
2. geneid2: A character vector giving ids for gene2.
3. groupid: A numerical vector giving labels of DC modules for gene1 and gene2, connected genes belong to same DC modules.
4. type: A character vector indicating which model is selected for gene1 and gene2.
5. gene1_weight: A numerical vector giving weight for gene1.
6. gene2_weight: A numerical vector giving weight for gene2.
7. bf.value: A numerical vector giving pair-wise Bayes factors for gene1 and gene2.
8. pc1: A numerical vector giving Pearson Correlation Coefficient of gene1 and gene2 under class 1.
9. pc2: A numerical vector giving Pearson Correlation Coefficient of gene1 and gene2 under class 2.
10. cortypes: A character vector indicating the direction of correlation changing in terms of Pearson Correlation Coefficient. Compare the difference of `pc1` and `pc2` with `corthres`, it gives values from ("down","up","unchange").
11. mean_g1_c1: A numerical vector giving the mean expression for gene1 under class 1.
12. mean_g1_c2: A numerical vector giving the mean expression for gene1 under class 2.
13. mean_foldchange_gene1: A numerical vector giving the fold change of mean expression for gene1 between class 1 and class 2.
14. mean_g2_c1: A numerical vector giving the mean expression for gene2 under class 1.
15. mean_g2_c2: A numerical vector giving the mean expression for gene2 under class 2.
16. mean_foldchange_gene2: A numerical vector giving the fold change of mean expression for gene2 between class 1 and class 2.

BF_similarity *Convert pair-wise Bayes factors into network adjacency*

Description

Convert pair-wise Bayes factors into network adjacency (values between 0 and 1) which can be used as an input for other methods.

Usage

```
BF_similarity(bfmatrix,softPower,bfthr=6,keepedges)
```

Arguments

bfmatrix	a data frame containing information of pair-wise bayes factors generated by Compute_bf. Refer to Compute_bf for details.
softPower	soft thresholding power. Such as the argument power in package WGCNA
bfthr	Bayes factor threshold. The default value is 6.
keepedges	a numerical variable indicating how many top edges to be remained in terms of Bayes factor. keepedges is invalid when is set as 0. When keepedges is not 0, it is prior to bfthr. The default value is the number of genes in dataExp.

Details

The function generates an adjacency matrix indicating the pair-wise differential co-expression under two different classes measured by pair-wise Bayes factors. It transfers pair-wise Bayes factor matrix bfmatrix into an adjacency matrix which can be used as an input for other methods. The arguments bfthr and keepedges are considered as hard threshold. When the hard threshold is set, the argument softPower for soft thresholding is usually set as 1 and only gene pairs which satisfy the bfthr threshold or keepedges threshold are considered in the clustering procedure.

Value

An adjacency matrix. If bfmatrix contains n different genes, the adjacency matrix is of dimensions n times n.

BF_WGCNA *Identify differential co-expression modules*

Description

Apply WGCNA [1] to identify differential co-expression modules through pair-wise Bayes factors.

Usage

```
BF_WGCNA(
        dataExp,
        bfmatrix,
        bfthr=6,
        keepedges=dim(dataExp)[2],
        softPower=(ifelse(keepedges != 0, 1,6)),
        plotTree=TRUE,
        plotfile="Gene_dendrogram_and_module_colors.pdf",
        trueModule=NULL,
        ...)
```

Arguments

dataExp	a data frame or matrix containing expression data. Columns correspond to genes and rows to samples. Must be kept the same throughout the analyses.
bfmatrix	a bfmatrix object containing information of pair-wise Bayes factors generated by Compute_bf. Refer to Compute_bf for details.
bfthr	Bayes factor threshold. The default value is 6
keepedges	a numerical variable indicating how many top edges to be remained in terms of Bayes factor. keepedges is invalid when is set as 0. When keepedges is not 0, it is prior to bfthr. The default value is the number of genes in dataExp.
softPower	soft thresholding power. Such as the argument power in package WGCNA. The default value is 1 when keepedges is not 0, otherwise is 6.
plotTree	if TRUE, plots a hierarchical clustering dendrogram and color annotation(s) of module(s) in file specified by argument plotfile.
plotfile	if plotTree is TRUE, specify a file (in pdf format) to output the dendrogram plot. The default file name is "Gene_dendrogram_and_module_colors.pdf".
trueModule	a vector of color names in the order of genes in dataExp. If not NULL, it represents the true module assignment for each gene, can be used as a comparison.
...	Arguments to be passed to method cutreeDynamic in package dynamicTreecut. see "https://CRAN.R-project.org/package=dynamicTreeCut" Arguments, like minClusterSize, cutHeight, deepSplit and method are very crucial for the clustering result. 1. minClusterSize: minimum cluster size. The default value is 20. 2. cutHeight: the maximum joining heights that will be considered. see see "https://CRAN.R-project.org/package=dynamicTreeCut". The default value is NULL. 3. deepSplit: provides a rough control over sensitivity to cluster splitting. see "https://CRAN.R-project.org/package=dynamicTreeCut" 4. method: chooses the method to use. Recognized values are "hybrid" and "tree".

Details

We apply WGCNA [1], an approach for co-expression analysis, to identify differential co-expression modules. It transfers pair-wise Bayes factor matrix into adjacency matrix first, then transfer this adjacency matrix into topological overlap matrix by TOMdist function from package WGCNA, then do hierarchical clustering using cutreeDynamic function from package dynamicTreeCut [2]. The

arguments bfthr and keepedges are considered as hard thresholding. When the hard thresholding is set, the argument softPower for soft thresholding is usually set as 1 and only gene pairs which satisfy the bfthr thresholding or keepedges thresholding are considered in the clustering procedure.

Value

A BFobt object. It's a list consist of one data frame component indicating module assignment ($modules), one matrix component indicating adjacency matrix ($adjacencies), one matrix component indicating Bayes factor matrix ($bf), one matrix component indicating which model is selected for gene1 and gene2 ($type), and one component indicating a hierarchical clustering dendrogram as produced by 'hclust' ($geneTree).

References

[1] Langfelder P, Horvath S. WGCNA: an R package for weighted correlation network analysis. BMC bioinformatics. 2008;9:1. [2] Langfelder P, Zhang B, Horvath S. Defining clusters from a hierarchical cluster tree: the Dynamic Tree Cut package for R. Bioinformatics. 2008;24:719-20.

Compute_bf *Calculation of Bayes factors*

Description

Estimate the stength of pair-wise differential co-expression by calculating Bayes factors between all genes from a given expression matrix.

Usage

```
Compute_bf(dataExp, class,classlabel,bfthr=6,echo=FALSE)
```

Arguments

dataExp	a data frame or matrix containing expression data. Columns correspond to genes and rows to samples. Must be kept the same throughout the analyses.
class	a numeric or character vector contains the corresponding class information of dataExp. By far, package only accepts binary classes.
classlabel	a numeric or character vector contains only two values. The first used to label class 1 and the second used to label class 2. the order of these two elements must be kept the same throughout the analyses.
bfthr	Bayes factor threshold. The default value is 6. If it's NULL, all the gene pairs are remained.
echo	a logical variable, indicates whether print processing information or not. If TRUE, it will print processing information for every 5000 gene pairs.

Details

The function estimate the stength of pair-wise differential co-expression by calculating Bayes factors between all genes from a given expression matrix. Gene pairs with Bayes factors higher than bfthr will be remained, if bfthr is not NULL.

Value

A bfmatrix object. It's a data frame containing the following elements:

geneid1	A character vector giving gene ids for gene1.
geneid2	A character vector giving gene ids for gene2.
bf.value	A numerical vector giving the values of Bayes factors for gene pair, gene1-gene2.
type	A character vector indicating which model is selected for the gene pair.

SelectPower *Choosing the soft-thresholding power by analysing of the network topology*

Description

Choose the soft thresholding power based on the criterion of approximate scale-free topology.

Usage

```
SelectPower(bfsimilarity,powers)
```

Arguments

bfsimilarity	an adjacency matrix based on pair-wise Bayes factor generated from BF_similarity.
powers	a vector of candidate soft-thresholding powers provided by user, soft-thresholding power will be chosen from it.

Details

This function plots a network topology under a set of soft-thresholding powers (a set of candidate softpower parameters for BF_WGCNA) provided by user. It constructs two panels in the plot, the left panel shows the scale-free fit index (y-axis) as a function of the soft-thresholding power (x-axis). The right panel shows the mean connectivity (degree, y-axis) as a function of the soft-thresholding power (x-axis). According to WGCNA [1] (https://labs.genetics.ucla.edu/horvath/htdocs/CoexpressionNetwork/Rpackages/WGCNA/Tutorials/FemaleLiver-02-pdf), they proposed to choose the lowest soft thresholding power and achieve an approximate scale-free topology (scale-free topology fit index reaches 0.9).

Value

None.

References

[1] B. Zhang and S. Horvath. A general framework for weighted gene co-expression network analysis. Statistical Applications in Genetics and Molecular Biology, 4(1):Article 17, 2005.

`sigDCpair_SFS` *The second step of selecting significant DC gene pairs.*

Description

It's for the second step of selecting significant DC gene pairs, after this procedure the final significant DC gene pairs are generated.

Usage

```
sigDCpair_SFS(
  dataExp,
  class,
  classlabel,
  sigDC,
  DC_acc="DC_pair_acc.txt",
  bfthr=6,
  by=50,
  LOOCV=TRUE,
  valid=NULL,
  valid_class=NULL
  )
```

Arguments

`dataExp`	a data frame or matrix containing expression data. Columns correspond to genes and rows to samples. Such as in `Compute_bf`.
`class`	a numeric or character vector contains the corresponding class information of dataExp. By far, package only accepts binary classes.
`classlabel`	a numeric or character vector contains only two values. The first used to label class 1 and the second used to label class 2. the order of these two elements must be kept the same throughout the analyses.
`sigDC`	a data frame contains the selected gene pairs generated from `sigDCpair_st1`.
`DC_acc`	specify a file to output the results. The default fila name is "DC_pair_acc.txt". Columns in the file are splited by 'Tab'.
`bfthr`	Bayes factor threshold. Only gene pairs with Bayes factor higher than `bfthr` are considered. The default value is 6.
`by`	an integer indicates the increment of gene pairs. The default value is 50.
`LOOCV`	logical: If `TRUE`, leave-one-out cross-validation (LOOCV) is used to monitor the accuracy of top N subset of gene pairs, otherwise a validation data must be supplied in argument `valid` and the accuracy is monitored on the validation data. The default value is `TRUE`.
`valid`	'NULL' (default) or data frame or matrix containing expression of validation data. Columns correspond to genes and rows to samples. Only used when `LOOCV` is `FALSE`. Must have compatible columns (order of columns and dimensions of columns) with `dataExp`.
`valid_class`	a character vector containing class information for validition data. It is sorted according to the rows of `valid`.

Details

The function generates a file specified in argument DC_acc which can be used to decide the final significant DC gene pairs. In this step, a generalized sequential forward selection (SFS) algorithm [1] is used on the candidate significant DC gene pairs generated by sigDCpair_st1. The gene pairs in sigDC are sorted in descending order of their corresponding scores. Start with an initial subset of gene pairs with size N=1, the next by gene pairs are repeatedly added to the subset of gene pairs, N=N+by. The goodness of top N subset of gene pairs are characterized by the leave-one-out cross-validation (LOOCV) or a validation dataset to monitor the accuracy for the top N gene pairs, and the accuracy is calculated by DC-based prediction model. The output file consist of two columns, the first indicates the number of top gene pairs and the second indicates the accuracy. According to this file, an accuracy curve can be plotted, than the decision of number N can be made as the final number of selected DC gene pairs. The final significant DC gene pairs is the top N gene pairs in sigDC.

Note

The running time of this function is closely related to the total number of candidate gene pairs generated from sigDCpair_st1 and the argument by.

References

[1] Whitney AW. A direct method of nonparametric measurement selection. Computers, IEEE Transactions on. 1971;100:1100-3.

---

sigDCpair_st1 *First step of selecting significant DC gene pairs.*

Description

Generate the candidate gene pairs selected from the first step of selecting significant DC gene pairs.

Usage

```
sigDCpair_st1(
  BFobt,
  mst2="mst2",
  bfthr=6,
  sparse=6,
  weight_cutoff=0.8
  )
```

Arguments

BFobt	a BFobt object generated by BF_WGCNA.
mst2	an input file specified to provide the MST2 skeletons of the DC structures. It must be the resulting MST2 file of BF_output_networks.
bfthr	Bayes factor threshold. Only gene pairs with Bayes factor higher than bfthr are considered. The default value is 6.
sparse	a tuning parameter controls the end-node weights in order to make a sparser subsets of gene pairs passing the weight_cutoff threshold. The defualt value is 6.

`weight_cutoff`
: threshold of end-node weights. The default value is 0.8.

Details

The function generates the candidate gene pairs selected from the first step of selecting significant DC gene pairs. The result is a data frame contains three columns, the first is a character vector representing ids for gene1, the second is a character vector representing ids for gene2 and the last is a numeric vector representing assignment scores for each gene pair. The result is in a descending order according to the score. The score is caculated by the geometric mean of the two end-node weights multiplied by the Bayes factor corresponding to the gene pair. In this procedure, each edges of MST2 stored in the file specified by argument mst2 will be searched, and those that satisfy the two thresholds, `bfthr` and `weight_cutoff`, will be remained. Here, the tuning parameter `sparse` and `weight_cutoff` are crucial to select the gene pairs. A higher `sparse` and higher `weight_cutoff` will result in a smaller set of gene pairs with higher prediction power.

Value

a data frame contains the candidate gene pairs from the first step of selecting significant DC gene pairs. It contains 3 columns:

1. geneid1: A character vector representing ids for gene1.
2. genedi2: A character vector representing ids for gene2.
3. score: A numeric vector representing assignment scores for each gene pair.

The data frame is sorted in a descending order according to the score.

References

Rahmatallah Y, Emmert-Streib F, Glazko G. Gene Sets Net Correlations Analysis (GSNCA): a multivariate differential coexpression test for gene sets. Bioinformatics. 2014;30:360-8.

`SimulationSmall` *small simulation data for test.*

Description

This matrix gives class information and expression data generated by a simulation procedure.

Usage

```
data(SimulationSmall)
```

Format

A 200 x 241 matrix. The first column (class) lists class labels, 1 indicate class 1 and 2 indicate class 2. Other columns correspond to genes. The rows are correspond to samples. There are 7 co-expression modules in total, each contains 20 genes, and 100 randomly generated genes. Module1-module2 have the same distribution in both class 1 and class 2, so they are stable modules. Module3 and module4 represent the intra-module DC, genes in these modules have differential co-expression patterns under different classes. Genes in module3 only correlated under one class. Genes in module4 highly correlated under one class, while less correlated under the other class. Module5

represents the shift scenario of DC, in which the correlations of genes under different classes do not change while their expression levels have an overall increase or decrease. Module6 and module7 represent the inter-module DC. Genes within each module have conserved co-expression patterns under two classes, but have a cross scenario between module6 and module7.

Examples

```
data(SimulationSmall)
head(SimulationSmall)
```

Index