Package 'ToPASeq'

April 10, 2015

Type Package

Title Package for Topology-based Pathway Analysis of RNASeq data

Version 1.0.1

Date 2014-27-05

Author Ivana Ihnatova, Eva Budinska

Maintainer Ivana Ihnatova <ihnatova@iba.muni.cz>

Description Implementation of seven methods for topology-based pathway analysis of both RNASeq and microarray data: SPIA, DEGraph, TopologyGSA, TAPPA, TBS, PWEA and a visualization tool for a single pathway.

Depends graphite, gRbase, graph, locfit

Imports R.utils, edgeR, DESeq2, GenomicRanges, igraph, DESeq, fields, limma, TeachingDemos, SPIA, clipper, topologyGSA

Suggests RUnit, BiocGenerics, gageData, Rgraphviz, DEGraph

LazyData yes

License AGPL-3

biocViews Software, GeneExpression, NetworkEnrichment, GraphAndNetwork, RNASeq, Visualization, Microarray, Pathways, DifferentialExpression,

R topics documented:

`ToPASeq-package` *Package for topology-based pathway analysis of microarray and RNASeq data*

Description

The package implementats several methods for topology-based pathway analysis of microarray data. The methods present in here are: SPIA, TopologyGSA, DEGraph, Clipper, PWEA, TAPPA, TBS. SPIA, PWEA and TBS were also adapted for RNASeq data.

Details

Package:	ToPASeq
Type:	Package
Version:	1.0
Date:	2014-03-04
License:	AGPL-3

Author(s)

Ivana Ihnatova

Maintainer: Ivana Ihnatova <ihnatova@iba.muni.cz>

Examples

```
## Not run:
if (require(DEGraph)) {
  data("Loi2008_DEGraphVignette")
  pathways<-biocarta[1:10]
  SPIA(exprLoi2008, classLoi2008,pathways , type="MA", logFC.th=-1, IDs="entrez")
  DEGraph(exprLoi2008, classLoi2008, pathways, type="MA")
  TAPPA(exprLoi2008, classLoi2008, pathways, type="MA")
  TopologyGSA(exprLoi2008, classLoi2008, pathways, type="MA", nperm=200)
  Clipper( exprLoi2008, classLoi2008+1, pathways,type="MA", test="mean")
  PWEA(exprLoi2008, classLoi2008, pathways,  type="MA", nperm=100)
  TBS( exprLoi2008, classLoi2008, pathways, type="MA",  logFC.th=-1, nperm=100)
}
```

```
if (require(gageData)) {

 data(hnrnp.cnts)
 group<-c(rep("sample",4), rep("control",4))
 SPIA( hnrnp.cnts, group, biocarta[1:10], type="RNASeq", logFC.th=-1, IDs="entrez", test="limma")
 DEGraph(hnrnp.cnts, group, biocarta[1:10], type="RNASeq", norm.method="TMM")
 TAPPA( hnrnp.cnts, group, biocarta[1:10], type="RNASeq", norm.method="TMM")
 TopologyGSA(hnrnp.cnts, group, biocarta[1:10], type="RNASeq",nperm=200, norm.method="TMM")
 Clipper(hnrnp.cnts, group,biocarta[1:10], type="RNASeq",  norm.method="TMM")
 PWEA(hnrnp.cnts, group, biocarta[1:10],  type="RNASeq", test="limma", nperm=100)
 TBS(hnrnp.cnts, group, biocarta[1:10], type="RNASeq", logFC.th=-1, nperm=100, test="limma")
 }

## End(Not run)
```

`AdjacencyMatrix2Pathway` *Function to coerce an adjacency matrix to a pathway*

Description

The function coerces an adjacency matrix to a pathway. Two types of matrices are allowed. The first one, where `1` denotes an edge between to nodes and `0` otherwise. This matrix is coerced into a simply pathway were type of all edges is set to `"process"`. There are three values present in the second type: `1` for an activation, `-1` for an inhibition and `0` otherwise (=no edge between two nodes). In this case, activations are set to `"process(activation)"` and inhibition to `"process(inhibition)"`. The symetricity of the matrix is used to decide between directed and undirected graph. Symmetric matrix is expected for undirected graph.

Usage

```
AdjacencyMatrix2Pathway(adjmat, name = "pathway", ident = "unknown", database = "unknown", species = "u
```

Arguments

`adjmat`	An adjacency matrix describing the pathway topology
`name`	A character, name of the pathway. Defaults to `"pathway"`
`ident`	A character, type of the identificators, e.g `"gene symbol"`
`database`	A character, the name of the database the topology comes from
`species`	A character, the species to which the topology belong
`date`	A date, the date the topology was created

Value

An object of class `pathway`

Author(s)

Ivana Ihnatova

See Also

pathway-class

Examples

```
genes<-paste("gene", 1:10, sep="")
adjmat<-matrix(sample(c(0,0,0,0,1), 100, TRUE),10,10, dimnames=list(genes,genes))
p<-AdjacencyMatrix2Pathway(adjmat)
head(edges(p))

adjmat<-matrix(sample(c(0,0,0,0,1,-1), 100, TRUE),10,10, dimnames=list(genes,genes))
p<-AdjacencyMatrix2Pathway(adjmat)
head(edges(p))
```

Clipper *Function to use clipper method on microarray or RNA-Seq data*

Description

clipper is a method for topological gene set analysis. It implements a two-step empirical approach based on the exploitation of graph decomposition into a junction tree to reconstruct the most relevant signal path. In the first step clipper selects significant pathways according to statistical tests on the means and the concentration matrices of the graphs derived from pathway topologies. Then, it "clips" the whole pathway identifying the signal paths having the greatest association with a specific phenotype.

Usage

```
Clipper(x, group, pathways, type = "MA", convert = TRUE, IDs = "entrez", both.directions=TRUE,
        test="mean", testCliques=FALSE, nperm=1000, norm.method = NULL)
```

Arguments

x	An ExpressionSet object or a gene expression data matrix or count matrix, rows refer to genes, columns to samples
group	Name or number of the phenoData column or a character vector or factor that contains required class assigments
pathways	A list of pathways in a form from graphite package
type	Type of the data, "MA" for microarray and "RNASeq" for RNA-Seq
convert	Logical, should nodes' labels be converted?
IDs	Character, which gene IDs should be used. One of "entrez" or "symbol"

`both.directions`
: Logical, should undirected edges be substitued by two directed edges (one in each direction) or only by one directed edge oriented as in `edges()`

`test`
: Character, a test to perform on the cliques. It can be mean or variance

`testCliques`
: Logical, should also be individual cliques for each pathway tested? Please, be aware that this calculation may take quite a lot of time

`nperm`
: Numeric, number of permutations in tests

`norm.method`
: Character, type of normalization method to be applied on RNA-Seq data. One of `"TMM"`, `"DESeq2"` or `"none"`. If `"none"`, normalization is not performed. TMM is coupled with voom transformation and DESeq2 with variance stabilizing transformation.

Value

A list,

`res`
: A data frame, where rows refer to pathways and columns are organized as follows: 1 - P-value from comparing the covariance matricesof whole pathway 2 - P-value from comparing the mean expression of whole pathway 3 - maximum score of the paths 4 - percentage of activation of the maximal scoring path 5 - impact of the maximal scoring path on the entire pathway 6 - genes forming the significant cliques 7 - genes forming the maximal scoring path)

`topo.sig`
: if `testCliques=TRUE`, a list where each slot contains the pvalues and a list of cliques in one pathway. `NULL` otherwise

`degtest`
: A numeric vector of gene-level differential expression statistics

Note

If only NA's are returned for a pathway, then less than 2 genes are present in the data. If there are NA's only in columns 3 to 7, then a junction tree could not be formed.

Author(s)

Ivana Ihnatova

References

Martini P, Sales G, Massa MS, Chiogna M, Romualdi C. Along signal paths: an empirical gene set approach exploiting pathway topology. Nucleic Acids Res. 2013 Jan 7;41(1):e19. doi: 10.1093/nar/gks866. Epub 2012 Sep 21. PubMed PMID: 23002139; PubMed Central PMCID: PMC3592432.

Examples

```
if (require(clipper)) {
if (require(DEGraph)) {
  data("Loi2008_DEGraphVignette")
  Clipper( exprLoi2008, classLoi2008, kegg[1],type="MA")
}
```

```
}
## Not run:
if (require(gageData)) {

 data(hnrnp.cnts)
 group<-c(rep("sample",4), rep("control",4))
 Clipper(hnrnp.cnts, group,kegg[1:10], type="RNASeq",  norm.method="TMM")
 }

## End(Not run)
```

`convertIdentifiersByVector` *Function to convert identifiers in pathways by user specified vector*

Description

The function converts identifiers of nodes in a pathway. It uses the user specified named vector for the conversion.

Usage

```
convertIdentifiersByVector(pathway, conv.table, id.type)
```

Arguments

`pathway`	An object of class `pathway`
`conv.table`	A named vector in which names correspond to the identifiers present in the pathway and values are the new identifiers to which conversion happens
`id.type`	A character, the type of the identifiers provided e.g `"TAIR"` for TAIR numbers. This is for informative purposes only.

Value

A `pathway` in which identifiers have been converted

Author(s)

Ivana Ihnatova

See Also

`convertIdentifiers`

Examples

```
g<-kegg[["Asthma"]]
conv<-setNames(paste("gene", 1:length(nodes(g)), sep=""), nodes(g))
gc<-convertIdentifiersByVector(g, conv, "dummy")
nodes(gc)
edges(gc)
```

`defaultEdgeAttrs` *Auxiliary data needed for SPIA method*

Description

This a list containing auxiliary data needed in SPIA method for conversion between edge types and dividing interaction into three categories: positive, negative and neutral

Usage

```
data(defaultEdgeAttrs)
```

Format

The format is: List of 2 $ graphite2SPIA: chr [1:26, 1:2] "binding" "control(In(ACTIVATION))" "control(In(INHIBITION))" "control(Out(ACTIVATION))" ... ..- attr(*, "dimnames")=List of 2 .. ..$: NULL .. ..$: chr [1:2] "type" "spiaType" $ beta :'data.frame': 25 obs. of 2 variables: ..$ rel : chr [1:25] "activation" "compound" "binding/association" "expression" ... ..$ beta: num [1:25] 1 0 0 1 -1 1 0 -1 -1 0 ...

Details

The first slot called `graphite2SPIA` contains a mapping table between edge types in topologies from `graphite` and edge types which are used in the implementation of SPIA in `SPIA` package. All of the edge types present in the topologies must be also covered by this table otherwise the method could not be applied.

The second slot called `beta` divides the 25 interaction types into three categories: positive (`beta=1`), negative (`beta=-1` and neutral (`beta=0`) in the sense of gene regulation. Only user familiar with all the details of SPIA should change this.

Value

A list of two data frames explained in the *Details*

Source

The data were cerated from the unexported objects from `graphite` package version 1.10.1.

Examples

```
data(defaultEdgeAttrs)
str(defaultEdgeAttrs)
```

DEGraph *Function to use DEGraph method on microarray or RNA-Seq data*

Description

DEGraph implements recent hypothesis testing methods which directly assess whether a particular gene network is differentially expressed between two conditions. In employs Graph Laplacian, Fourier transformation and multivariate T2-statistic

Usage

```
DEGraph(x, group, pathways, type = "MA", convert = TRUE, IDs = "entrez",
     gene.stat="logFC",both.directions=TRUE, overall="biggest", useInteractionSigns=TRUE, norm.meth
```

Arguments

x	An ExpressionSet object or a gene expression data matrix or count matrix, rows refer to genes, columns to samples
group	Name or number of the phenoData column or a character vector or factor that contains required class assigments
pathways	A list of pathways in a form from graphite package
type	Type of the data, "MA" for microarray and "RNASeq" for RNA-Seq
convert	Logical, should nodes' labels be converted?
IDs	Character, which gene IDs should be used. One of "entrez" or "symbol"
gene.stat	Character, which gene-level statistic should be exported? "logFC" stands for log fold-change and "stats" for the modified t-statistic from limma package
both.directions	Logical, should undirected edges be substitued by two directed edges (one in each direction) or only by one directed edge oriented as in edges()
overall	Character, how should the overall p-value for a pathway be calculated. The possible values are: "mean", "min", "biggest". "biggest" returns the p-value of the biggest connected component.
useInteractionSigns	Logical, should types of interaction be included in the analysis?
norm.method	Character, type of normalization method to be applied on RNA-Seq data. One of "TMM" , "DESeq2" or "none". If "none", normalization is not performed. TMM is coupled with voom transformation and DESeq2 with variance stabilizing transformation.

Value

A list:

res	Results from analysis of individual pathways. The first column refers to the overall p-value for a pathway. Then groups of four columns follows. One group refers to one connected component and contains a pair of p-values (without and with Fourier transformation), graph and number of Fourier componets used in the test. The number of groups is equal to the highest number of components in analysed pathways. Components are sorted in the decreasing order of their nodes number.
topo.sig	NULL, present for the compatibility with outputs from other methods
degtest	A numeric vector of gene-level statistics of all genes in the dataset

Warning

The method is applied witout interaction signs. Argument useInteractionSigns in the original function testOneGraph from DEGraph package is set to FALSE.

Author(s)

Ivana Ihnatova

References

L. Jacob, P. Neuvial, and S. Dudoit. Gains in power from structured two-sample tests of means on graphs. Technical Report arXiv:q-bio/1009.5173v1, arXiv, 2010.

Examples

```
if (require(DEGraph)) {
  data("Loi2008_DEGraphVignette")

  DEGraph(exprLoi2008, classLoi2008, biocarta[1:10], type="MA")
}
## Not run:
if (require(gageData)) {

 data(hnrnp.cnts)
 group<-c(rep("sample",4), rep("control",4))
 DEGraph(hnrnp.cnts, group, biocarta[1:10], type="RNASeq", norm.method="TMM")
}

## End(Not run)
```

---

graphNEL2Pathway *Function to coerce a graphNEL to a pathway*

Description

The function coerces a graphNEL to a pathway. It attempts to recover the edge types from "edgeType" attribute of edgeData. If it is not present "process(indirect effect)" is used in order to preserve directionality.

Usage

```
graphNEL2Pathway(graph, name = "pathway", ident = "unknown", database = "unknown", species = "unknown",
```

Arguments

graph	A graphNEL object to be coerced.
name	A character, name of the pathway. Defaults to "pathway"
ident	A character, type of the identificators, e.g "gene symbol"
database	A character, the name of the database the topology comes from
species	A character, the species to which the topology belong
date	A date, the date the topology was created

Value

A coerced pathway

Note

Source and destination nodes may be swaped when coercing pathway to graphNEL via pathwayGraph and back.

Author(s)

Ivana Ihnatova

See Also

pathway-class, pathwayGraph

Examples

```
kegg[[1]]
pathwayGraph(kegg[[1]])
graphNEL2Pathway(pathwayGraph(kegg[[1]]))

set.seed(123)
rg <- randomEGraph(LETTERS[1:20], edges = 30)
p<-graphNEL2Pathway(rg)
p
head(edges(p))
```

PWEA *Function to use PWEA method on microarray or RNA-Seq data*

Description

The function runs PWEA method (please see References for the details) on gene expression data matrix, vector specifing to which group a sample belongs and a list of pathway graphs. Briefly, it is a weighted GSEA-like method. The weightes are based on the distance and Pearson's correlation between genes in a pathway.

Usage

```
PWEA(x, group, pathways, type = "MA", test=NULL, convert = TRUE, IDs = "entrez",
     gene.stat="logFC", both.directions=TRUE, alpha = 0.05, nperm = 5000)
```

Arguments

x	An ExpressionSet object or a gene expression data matrix or count matrix, rows refer to genes, columns to samples
group	Name or number of the phenoData column or a character vector or factor that contains required class assigments
pathways	A list of pathways in a form from graphite package
type	Type of the data, "MA" for microarray and "RNASeq" for RNA-Seq
test	A character, which approach should be used to calculate differentially expressed genes from RNASeq data. One of "vstlimma", "voomlimma", "limmaMA" or "DESeq2". "limmaMA" refers to to limma-method for microarray data. "vstlimma" referes to limma with variance stabilization normalization, "voomlimma" refers to limma with voom transformation, DESeq2 method takes much longer and therefore it is not recommended.
convert	Logical, should nodes' labels be converted?
IDs	Character, which gene IDs should be used. One of "entrez" or "symbol"
gene.stat	Character, which gene-level statistic should be exported? "logFC" stands for log fold-change and "stats" for the modified t-statistic from limma package
both.directions	Logical, should undirected edges be substitued by two directed edges (one in each direction) or only by one directed edge oriented as in edges()
alpha	Numeric, a parameter used when calculating genes weights
nperm	Numeric, number of permutations

Value

A list

res	A data frame, rows refer to pathways. It contains: Enrichment score for a pathway, p-value and p-value adjusted for multiple hypothesis testing by Benjamini-Hochberg's FDR method. NA's if less than 2 nodes are present in the data
topo.sig	A list, topology influence factors for the genes in individual pathways. NULL if less than 2 nodes are present in the data
degtest	A named vector of statistics from testing the differential expression

Author(s)

Ivana Ihnatova

References

Hung, JH., Whitfield, T. W., Yang, TH., Hu, Z., Weng, Z., DeLisi, Ch. (2010) Identification of functional modules that correlate with phenotypic difference: the influence of network topology, Genome Biology, 11:R23

Examples

```
## Not run:
if (require(DEGraph)) {
  data("Loi2008_DEGraphVignette")
  PWEA(exprLoi2008, classLoi2008, biocarta[1:10],  type="MA", nperm=100)
}


if (require(gageData)) {
 data(hnrnp.cnts)
 group<-c(rep("sample",4), rep("control",4))
 PWEA(hnrnp.cnts, group, biocarta[1:10],  type="RNASeq", test="vstlimma", nperm=100)
}

## End(Not run)
```

---

reduceGraph *Function to reduce the pathway graph*

Description

Function simplifies a pathway graph topology. It merges a user specified nodes into a one. The specified set of nodes must be either a gene family or a protein complex. By a gene family we mean a set of genes with same outgoing or incoming edges. On the other hand, a protein complex is a set of nodes with only undirected binding edges between them and the number of edges is equal to the complex size.

Usage

```
reduceGraph(graph, reduction)
```

Arguments

graph	An object of class pathway, a pathway to be reduced
reduction	A named list of reductions to be maded.

Value

A pathway

Author(s)

Ivana Ihnatova

Examples

```
gr<-convertIdentifiers(kegg[["Prolactin signaling pathway"]],"symbol")
red<-list(RAS=c("NRAS","KRAS","HRAS"), SHC=c("SHC1", "SHC4","SHC2","SHC3"))
reduced<-reduceGraph(gr, red)
reduced
par(mfrow=c(1,2))

nA<-list(fillcolor=c(NRAS="red", KRAS="red", HRAS="red", SHC1="green", SHC4="green", SHC2="green", SHC3="green")
plot(pathwayGraph(gr), nodeAttrs=nA, attrs=list(node=list(fontsize=30, height=40)), main="Before")

plot(pathwayGraph(reduced),
 nodeAttrs=list(fillcolor=c(RAS="red", SHC="green")), attrs=list(node=list(fontsize=30, height=40)), main="After

#this throws an error, "RELA", "FOS","NFKB1" is not correct set of genes
## Not run:
gr<-convertIdentifiers(kegg[["Prolactin signaling pathway"]],"symbol")
red<-list(RAS=c("NRAS","KRAS","HRAS"), SHC=c("RELA", "FOS","NFKB1"))
reduced<-reduceGraph(gr, red)

## End(Not run)

gr<-convertIdentifiers(kegg[["p53 signaling pathway"]], "symbol")
reduced<-reduceGraph(gr, list(com=c("CCNE1", "CDK2", "CCNE2")))

par(mfrow=c(1,2))

nA<-list(fillcolor=c(CCNE1="red", CDK2="red", CCNE2="red"))
plot(pathwayGraph(gr), nodeAttrs=nA, attrs=list(node=list(fontsize=30, height=40)), main="Before")

plot(pathwayGraph(reduced),
 nodeAttrs=list(fillcolor=c(com="red")), attrs=list(node=list(fontsize=30, height=40)), main="After")
```

res *Function to extract parts of object*

Description

Function extracts part of an object named "res", "topo.sig", "degtable"

Usage

```
res(object)
topo.sig(object)
degtable(object)
```

Arguments

object	Object of defined class. Methods for topResult are available in this package

Value

Extracted parts of an object. Data type varies between parts and the origin of the object

Author(s)

Ivana Ihnatova

SPIA *Function to use SPIA method on microarray or RNA-Seq data*

Description

The function runs SPIA method on microarray or RNA-Seq data. The implementatio includes the identification of differentially expressed genes and transformation of pathways' topologies to an appropriate form. The SPIA method combines two independent p-values. One p-value comes from overrepresentation analysis and the other is so called pertubation factor.

Usage

```
SPIA(x, group, pathways, type="MA", convert=TRUE, IDs = "entrez",
   gene.stat="logFC", both.directions=TRUE, logFC.th = 2, p.val.th = 0.05, test = NULL, edgeAttrs=defau
```

Arguments

x	An ExpressionSet object or a gene expression data matrix or count matrix, rows refer to genes, columns to samples
group	Name or number of the phenoData column or a character vector or factor that contains required class assigments
pathways	A list of pathways in a form from graphite package
type	Type of the data, "MA" for microarray and "RNASeq" for RNA-Seq
convert	Logical, should nodes' labels be converted?
IDs	Character, which gene IDs should be used. One of "entrez" or "symbol"
gene.stat	Character, which gene-level statistic should be exported? "logFC" stands for log fold-change and "stats" for the modified t-statistic from limma package
both.directions	Logical, should undirected edges be substitued by two directed edges (one in each direction) or only by one directed edge oriented as in edges()
logFC.th	A numeric, threshold to select differentially expressed genes according to log Fold-Change. Use negative, if this criterion should not be used.
p.val.th	A numeric, threshold to select differentially expressed genes according to p-value from moderated t-test. Use 1 if this criterion should not be applied
test	Character, which test should be used for differential expression analysis of RNA-Seq data. A character, which approach should be used to calculate differentially expressed genes from RNASeq data. One of "vstlimma", "voomlimma", "limmaMA" or "DESeq2". "limmaMA" refers to to limma-method for microarray data. "vstlimma" referes to limma with variance stabilization normalization, "voomlimma" refers to limma with voom transformation.
edgeAttrs	Auxiliary data needed for SPIA method. See ?edgeAttrs for more details
...	Preserved for other arguments to spia. These should not be used.

Value

A list:

res	A matrix with columns as descibed below: pSize - Pathway size, number of genes, NDE - Number of differentially expressed genes, pNDE - P-value of the overrepresentation part of the method, tA - The observed total preturbation accumulation in the pathway, pPERT - P-value of the pertubation part of the method, p - Combined p-value (overrepresentation and pertubation), pFdr - False discovery rate adjusted p, pFWER - FWER adjusted p, Status - If a pathway was identified as Acivated or Inhibited
topo.sig	A list of accumulated pertubation factors and log fold-changes for genes in individual pathways
degtest	A numeric vector of gene-level differential expression statistics of all genes in the dataset

Author(s)

Ivana Ihnatova

References

Tarca AL, Draghici S, Khatri P, Hassan SS, Mittal P, Kim JS, Kim CJ, Kusanovic JP, Romero R. A novel signaling pathway impact analysis. Bioinformatics. 2009 Jan 1;25(1):75-82.

Adi L. Tarca, Sorin Draghici, Purvesh Khatri, et. al, A Signaling Pathway Impact Analysis for Microarray Experiments, 2008, Bioinformatics, 2009, 25(1):75-82.

Draghici, S., Khatri, P., Tarca, A.L., Amin, K., Done, A., Voichita, C., Georgescu, C., Romero, R.: A systems biology approach for pathway level analysis. Genome Research, 17, 2007.

Examples

```
if (require(DEGraph)) {
  data("Loi2008_DEGraphVignette")
  p<-biocarta[1:2]
  SPIA(exprLoi2008, classLoi2008,p, type="MA", logFC.th=-1, IDs="entrez")
}
## Not run:
if (require(gageData)) {

 data(hnrnp.cnts)
 group<-c(rep("sample",4), rep("control",4))
SPIA( hnrnp.cnts, group, biocarta[1:10], type="RNASeq", logFC.th=-1, IDs="entrez", test="vstlimma")
 }

## End(Not run)
```

TAPPA *Function to use TAPPA method on microarray or RNA-Seq data*

Description

The functions analyses the differential expression of pathways via TAPPA method. Expression is compared between two groups of samples by Mann-Whitney test. P-values are later adjusted for multiple hypothesis testing by Benjamini-Hochberg's FDR method.

Usage

```
TAPPA(x, group, pathways, type = "MA", convert = TRUE, IDs = "entrez",
    gene.stat="logFC", both.directions=TRUE, normalize = TRUE, verbose = FALSE, norm.method = NULL)
```

Arguments

x	An ExpressionSet object or a gene expression data matrix or count matrix, rows refer to genes, columns to samples
group	Name or number of the phenoData column or a character vector or factor that contains required class assigments
pathways	A list of pathways in a form from graphite package

type	Type of the data, "MA" for microarray and "RNASeq" for RNA-Seq
convert	Logical, should nodes' labels be converted?
IDs	Character, which gene IDs should be used. One of "entrez" or "symbol"
gene.stat	Character, which gene-level statistic should be exported? "logFC" stands for log fold-change and "stats" for the modified t-statistic from limma package]
both.directions	Logical, should undirected edges be substitued by two directed edges (one in each direction) or only by one directed edge oriented as in edges()
normalize	Logical, should data be normalized?
verbose	Logical, if TRUE names of the pathways are printed as they are analysed
norm.method	Character, type of normalization method to be applied on RNA-Seq data. One of "TMM" , "DESeq2" or "none". If "none", normalization is not performed. TMM is coupled with voom transformation and DESeq2 with variance stabilizing transformation.

Value

A list,

res	A data frame, rows refer to pathways. Columns contain: number of valid PCI-scores, median, min and max of the PCI scores for each group of samples (denoted with 1 or 2), p-value of the Mann-Whitney test (p.val) and adjusted p-value (p.adj). If less than two nodes are present in the data, the function puts NA's in all columns.
topo.sig	NULL, it is preserved for the compatibility with other methods implemented in this package
degtest	A numeric vector of gene-level differential expression statistics

Author(s)

Ivana Ihnatova

References

Gao, S. and Wang, X. (2007) TAPPA: topological analysis of pathway phenotype association. Bioinformatics, 23, pages 3100-3102

Examples

```
if (require(DEGraph)) {
  data("Loi2008_DEGraphVignette")
  TAPPA(exprLoi2008, classLoi2008, biocarta[1:10], type="MA")
}

## Not run:
if (require(gageData)) {
```

```
 data(hnrnp.cnts)
 group<-c(rep("sample",4), rep("control",4))
 TAPPA( hnrnp.cnts, group, biocarta[1:10], type="RNASeq", norm.method="TMM")
}

## End(Not run)
```

TBS *Function to use TBS method on microarray or RNA-Seq data*

Description

A function runs TBS method on a gene expression data matrix or count matrix and vector dividing samples into two groups and a set of pathways from `graphite` package. The TBS methods (please see Reference for the details) was adapted to `graphite`'s graphs where each node is represented only by one gene.

Usage

```
TBS(x, group, pathways, type = "MA", test = NULL, convert = TRUE, IDs = "entrez",
   gene.stat="logFC", both.directions=TRUE, logFC.th = 2, p.val.th = 0.05, nperm = 1000)
```

Arguments

`x`	An `ExpressionSet` object or a gene expression data matrix or count matrix, rows refer to genes, columns to samples
`group`	Name or number of the phenoData column or a character vector or factor that contains required class assigments
`pathways`	A list of pathways in a form from `graphite` package
`type`	Type of the data, `"MA"` for microarray and `"RNA-Seq"` for RNA-Seq
`test`	A character, which approach should be used to calculate differentially expressed genes from RNASeq data. One of `"vstlimma"`, `"voomlimma"`, `"limmaMA"` or `"DESeq2"`. `"limmaMA"` refers to to `limma`-method for microarray data. `"vstlimma"` referes to `limma` with variance stabilization normalization, `"voomlimma"` refers to `limma` with `voom` transformation
`convert`	Logical, should nodes' labels be converted?
`IDs`	Character, which gene IDs should be used. One of `"entrez"` or `"symbol"`
`gene.stat`	Character, which gene-level statistic should be exported? `"logFC"` stands for log fold-change and `"stats"` for the modified t-statistic from `limma` package
`both.directions`	Logical, should undirected edges be substitued by two directed edges (one in each direction) or only directed edges are considered NOTE: this is different from the other methods implemented in this package
`logFC.th`	Numeric, threshold for log fold-change of a gene to identify the gene as differentially expressed. Use negative if you don't want any threshold to be applied

p.val.th	Numeric, threshold for p-values of moderated t-test (from limma package) of a gene to identify the gene as significantly differentially expressed. Use 1 if you don't want any threshold to be applied
nperm	Numeric, number of permutations

Value

A list,

res	A data frame with normalized score, p-value and FDR-adjusted p-value for each pathway
topo.sig	A list with log fold-changes and number of downstream differentially expressed nodes for nodes of individual pathways
degtest	A named vector of statistics from testing the differential expression of genes

Author(s)

Ivana Ihnatova

References

Maysson Al-Haj Ibrahim, Sabah Jassim, Michael Anthony Cawthorne, and Kenneth Langlands. A Topology-Based Score for Pathway Enrichment, Journal of Computational Biology. May 2012, 19(5): 563-573

Examples

```
if (require(DEGraph)) {
  data("Loi2008_DEGraphVignette")
  TBS( exprLoi2008, classLoi2008, biocarta[1:10], type="MA",  logFC.th=-1, nperm=100)
}
## Not run:
if (require(gageData)) {

 data(hnrnp.cnts)
 group<-c(rep("sample",4), rep("control",4))

 TBS(hnrnp.cnts, group, biocarta[1:10], type="RNASeq", logFC.th=-1, nperm=100, test="vstlimma")
}

## End(Not run)
```

TopologyGSA *Function to use TopologyGSA method on microarray or RNA-Seq data*

Description

TopologyGSA method uses graphical models to test the differential expression of a pathway. It also highlights pathway componenets involved in the deregulation.

Usage

```
TopologyGSA(x, group, pathways, type = "MA", convert = TRUE, IDs = "entrez", both.directions=TRUE,
            test="mean", testCliques=FALSE, alpha=0.05, nperm = 10000,
            norm.method = NULL, maxNodes = 150)
```

Arguments

x	An ExpressionSet object or a gene expression data matrix or count matrix, rows refer to genes, columns to samples
group	Name or number of the phenoData column or a character vector or factor that contains required class assigments
pathways	A list of pathways in a form from graphite package
type	Type of the data, "MA" for microarray and "RNASeq" for RNA-Seq
convert	Logical, should nodes' labels be converted?
IDs	Character, which gene IDs should be used. One of "entrez" or "symbol"
both.directions	Logical, should undirected edges be substitued by two directed edges (one in each direction) or only by one directed edge oriented as in edges()
test	Either "var" and "mean". Determine the type of test used by topologyGSA.
testCliques	Logical, should also be individual cliques for each pathway tested? Please, be aware that this calculation may take quite a lot of time
alpha	Numeric, threshold for statistical significance
nperm	Numeric, Number of permutations
norm.method	Character, type of normalization method to be applied on RNA-Seq data. One of "TMM" , "DESeq2" or "none". If "none", normalization is not performed. TMM is coupled with voom transformation and DESeq2 with variance stabilizing transformation.
maxNodes	Pathways with more nodes than specified value will be omitted from the analysis

Value

A list

res	a list with one entry for each successfully analyzed pathway
topo.sig	if testCliques=TRUE, a list where each slot contains the pvalues and a list of cliques in one pathway. NULL otherwise
degtest	A numeric vector of gene-level differential expression statistics

Author(s)

Ivana Ihnatova

References

Massa MS, Chiogna M, Romualdi C. Gene set analysis exploiting the topology of a pathway. BMC System Biol. 2010 Sep 1;4:121.

Examples

```
## Not run:
if (require(DEGraph)) {
  data("Loi2008_DEGraphVignette")
  p<-biocarta[1:10]
 TopologyGSA(exprLoi2008, classLoi2008, p, type="MA", test="mean", alpha=0.05, nperm=200)
  }

if (require(gageData)) {

 data(hnrnp.cnts)
 group<-c(rep("sample",4), rep("control",4))
 TopologyGSA(hnrnp.cnts, group, biocarta[1:10], type="RNASeq",test="mean", alpha=0.05,
   nperm=200, norm.method="TMM")
}

## End(Not run)
```

Index