

Package ‘graphite’

April 10, 2015

Version 1.12.0

Date 2014-10-02

Title GRAPH Interaction from pathway Topological Environment

Author Gabriele Sales <gabriele.sales@unipd.it>, Enrica Calura
<enrica.calura@gmail.com>, Chiara Romualdi
<chiara.romualdi@unipd.it>

Maintainer Gabriele Sales <gabriele.sales@unipd.it>

Description Graph objects from pathway topology derived from Biocarta,
HumanCyc, KEGG, NCI, Panther, Reactome and SPIKE databases.

License AGPL-3

Depends R (>= 2.10)

Imports AnnotationDbi, graph, graphics, methods, org.Hs.eg.db, stats,
utils

Suggests BiocStyle, DEGraph (>= 1.4), hgu133plus2.db, RCytoscape (>=
1.6), SPIA (>= 2.2), topologyGSA (>= 1.4.0), clipper, ALL

LazyData yes

biocViews Pathways, ThirdPartyClient, GraphAndNetwork, Network

R topics documented:

biocarta	2
convertIdentifiers	3
cytoscapePlot	4
humancyc	4
kegg	5
nci	6
panther	7
pathway-class	8
pathwayGraph	9
prepareSPIA	10
reactome	10

runClipper	11
runClipperMulti	12
runDEGraph	14
runDEGraphMulti	15
runSPIA	16
runTopologyGSA	17
runTopologyGSAMulti	18
spike	20
Index	21

biocarta

BioCarta pathways

Description

A list with a [pathway](#) object for each BioCarta pathway.

<http://www.biocarta.com>

Pathway topology conversion

BioCarta pathways were retrieved in BioPax format from the PDI database web page.

We define a pathway for each BioPax tag “pathway”. Pathway nodes often correspond to multiple gene products. These can be divided into protein complexes (proteins linked by protein-protein interactions) and groups made of alternative members (genes with similar biochemical functions). Thus, when considering signal propagation these groups are considered differently. The first kind (hereafter group AND) should be expanded into a clique (all proteins connected to the others), while the second (hereafter group OR) should be expanded without connection among them. In the BioPax format only one type of group is allowed: protein complexes (group AND) with the tag ‘complex’. However, it often happens that the ‘protein’ tag contains multiple ‘xref’ pointing to alternative elements of the process (group OR).

Compound mediated interactions are interactions for which a compound acts as a bridge between two elements. Since chemical compounds are not usually measured with high-throughput technology, they should be removed from the network to analyse gene signals. However, the trivial elimination of the compounds, without signal propagation, will strongly bias the topology interrupting the signals that pass through them. If element ‘A’ is linked to compound ‘c’ and compound ‘c’ is linked to element ‘B’, element ‘A’ should be linked to element ‘B’. Not all compounds are considered for the propagation because some of them (for example: H₂O, ATP, ADP) are highly frequent in map descriptions and the signal propagation through them would lead to chains too long. Compounds not considered for propagation are not characteristic of a specific reaction, but act as secondary substrates/products widely shared among different processes.

graphite allows the user to see the single/multiple relation types that characterized an edge. The type of edges have been kept as much as possible similar to those annotated in the original data format. Some new types have been introduced due to topological conversion needs.

See Also

[pathway-class](#).

convertIdentifiers *Convert the node identifiers of a pathway.*

Description

Converts the node identifiers of a pathway to either Entrez Genes or Gene Symbols.

Usage

```
convertIdentifiers(pathway, type)
```

Arguments

pathway	a pathway from biocarta , kegg , nci or reactome .
type	the type of identifier: either "entrez" or "symbol".

Value

A pathway object.

See Also

[biocarta](#)

[kegg](#)

[nci](#)

[reactome](#)

[pathway-class](#)

Examples

```
convertIdentifiers(reactome$mTOR signalling, type="symbol")
```

cytoscapePlot

Plot a pathway graph in Cytoscape

Description

Renders the topology of a pathway as a Cytoscape graph.

Usage

```
cytoscapePlot(pathway, ...)
```

Arguments

pathway a pathway object.
... optional arguments to pass to [pathwayGraph](#).

Details

Requires the RCytoscape package.

See Also

[pathway-class](#)
[pathwayGraph](#)

Examples

```
## Not run: cytoscapePlot(convertIdentifiers(reactome$Unwinding of DNA, "symbol"))
```

humancyc*Humancyc pathways*

Description

A list with a [pathway](#) object for each HumanCyc pathway.

Caspi R, Altman T, Dale JM, Dreher K, Fulcher CA, Gilham F, Kaipa P, Karthikeyan AS, Kothari A, Krummenacker M, Latendresse M, Mueller LA, Paley S, Popescu L, Pujar A, Shearer AG, Zhang P, Karp PD. The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. Nucleic Acids Research 38:D473-9 2010.

<http://humancyc.org>

Pathway topology conversion

HumanCyc pathways were retrieved in the BioPax format downloaded by the Pathway Commons web page <http://www.pathwaycommons.org>.

We define a pathway for each BioPax tag “pathway”. Pathway nodes often correspond to multiple gene products. These can be divided into protein complexes (proteins linked by protein-protein interactions) and groups made of alternative members (genes with similar biochemical functions). Thus, when considering signal propagation these groups are considered differently. The first kind (hereafter group AND) should be expanded into a clique (all proteins connected to the others), while the second (hereafter group OR) should be expanded without connection among them. In the BioPax format only one type of group is allowed: protein complexes (group AND) with the tag ‘complex’. However, it often happens that the ‘protein’ tag contains multiple ‘xref’ pointing to alternative elements of the process (group OR).

Compound mediated interactions are interactions for which a compound acts as a bridge between two elements. Since chemical compounds are not usually measured with high-throughput technology, they should be removed from the network to analyse gene signals. However, the trivial elimination of the compounds, without signal propagation, will strongly bias the topology interrupting the signals that pass through them. If element ‘A’ is linked to compound ‘c’ and compound ‘c’ is linked to element ‘B’, element ‘A’ should be linked to element ‘B’. Not all compounds are considered for the propagation because some of them (for example: H₂O, ATP, ADP) are highly frequent in map descriptions and the signal propagation through them would lead to chains too long. Compounds not considered for propagation are not characteristic of a specific reaction, but act as secondary substrates/products widely shared among different processes.

graphite allows the user to see the single/multiple relation types that characterized an edge. The type of edges have been kept as much as possible similar to those annotated in the original data format. Some new types have been introduced due to topological conversion needs.

See Also

[pathway-class](#).

kegg

KEGG pathways

Description

A list with a [pathway](#) object for each KEGG pathway.

Ogata H, Goto S, Sato K, Fujibuchi W, Bono H, Kanehisa M. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.* 1999 Jan 1;27(1):29-34.

<http://www.genome.jp/kegg/>

Pathway topology conversion

KEGG pathway were retrieved in KGML format from the KEGG ftp site.

KEGG database provides separate xml files, one for each pathway. A pathway is therefore define by all the reactions described within each file.

Pathway nodes often correspond to multiple gene products. These can be divided into protein complexes (proteins linked by protein-protein interactions) and groups made of alternative members (genes with similar biochemical functions). Thus, when considering signal propagation these groups are considered differently. The first kind (hereafter group AND) should be expanded into a clique (all proteins connected to the others), while the second (hereafter group OR) should be expanded without connection among them. In the KGML format there are two ways of defining nodes with multiple elements: protein complexes (group AND defined by entry type="group") and groups with alternative members (group OR defined by entry type="gene").

Compound mediated interactions are interactions for which a compound acts as a bridge between two elements. Since chemical compounds are not usually measured with high-throughput technology, they should be removed from the network to analyse gene signals. However, the trivial elimination of the compounds, without signal propagation, will strongly bias the topology interrupting the signals that pass through them. If element 'A' is linked to compound 'c' and compound 'c' is linked to element 'B', element 'A' should be linked to element 'B'. Within the KGML format there are two different ways of describing a compound mediated interaction: i) direct interaction type="PPrel" ('A' interacts with 'B' through compound 'c') and ii) indirect one type="PCrel" ('A' interacts to compound 'c' and 'c' interacts with 'B').

Not all compounds are considered for the propagation because some of them (for example: H₂O, ATP, ADP) are highly frequent in map descriptions and the signal propagation through them would lead to chains too long. Compounds not considered for propagation are not characteristic of a specific reaction, but act as secondary substrates/products widely shared among different processes.

graphite allows the user to see the single/multiple relation types that characterized an edge. The type of edges have been kept as much as possible similar to those annotated in the original data format. Some new types have been introduced due to topological conversion needs.

See Also

[pathway-class](#).

nci

NCI pathways

Description

A list with a [pathway](#) object for each NCI pathway.

Schaefer CF, Anthony K, Krupa S, Buchoff J, Day M, Hannay T, Buetow KH. PID: the Pathway Interaction Database. *Nucleic Acids Res.* 2009 Jan;37(Database issue):D674-9. Epub 2008 Oct 2.

<http://pid.nci.nih.gov/>

Pathway topology conversion

NCI pathways were retrieved in BioPax format from the PDI database web page.

We define a pathway for each BioPax tag "pathway". Pathway nodes often correspond to multiple gene products. These can be divided into protein complexes (proteins linked by protein-protein interactions) and groups made of alternative members (genes with similar biochemical functions).

Thus, when considering signal propagation these groups are considered differently. The first kind (hereafter group AND) should be expanded into a clique (all proteins connected to the others), while the second (hereafter group OR) should be expanded without connection among them. In the BioPax format only one type of group is allowed: protein complexes (group AND) with the tag 'complex'. However, it often happens that the 'protein' tag contains multiple 'xref' pointing to alternative elements of the process (group OR).

Compound mediated interactions are interactions for which a compound acts as a bridge between two elements. Since chemical compounds are not usually measured with high-throughput technology, they should be removed from the network to analyse gene signals. However, the trivial elimination of the compounds, without signal propagation, will strongly bias the topology interrupting the signals that pass through them. If element 'A' is linked to compound 'c' and compound 'c' is linked to element 'B', element 'A' should be linked to element 'B'. Not all compounds are considered for the propagation because some of them (for example: H₂O, ATP, ADP) are highly frequent in map descriptions and the signal propagation through them would lead to chains too long. Compounds not considered for propagation are not characteristic of a specific reaction, but act as secondary substrates/products widely shared among different processes.

graphite allows the user to see the single/multiple relation types that characterized an edge. The type of edges have been kept as much as possible similar to those annotated in the original data format. Some new types have been introduced due to topological conversion needs.

See Also

[pathway-class](#).

panther

Panther pathways

Description

A list with a [pathway](#) object for each Panther pathway.

PANTHER in 2013: modeling the evolution of gene function, and other gene attributes, in the context of phylogenetic trees. Huaiyu Mi, Anushya Muruganujan and Paul D. Thomas Nucl. Acids Res. (2012) doi: 10.1093/nar/gks1118

www.pantherdb.org/pathway/

Pathway topology conversion

Panther pathways were retrieved in the BioPax format downloaded by the Pathway Commons web page <http://www.pathwaycommons.org>.

We define a pathway for each BioPax tag "pathway". Pathway nodes often correspond to multiple gene products. These can be divided into protein complexes (proteins linked by protein-protein interactions) and groups made of alternative members (genes with similar biochemical functions). Thus, when considering signal propagation these groups are considered differently. The first kind (hereafter group AND) should be expanded into a clique (all proteins connected to the others), while the second (hereafter group OR) should be expanded without connection among them. In

the BioPax format only one type of group is allowed: protein complexes (group AND) with the tag 'complex'. However, it often happens that the 'protein' tag contains multiple 'xref' pointing to alternative elements of the process (group OR).

Compound mediated interactions are interactions for which a compound acts as a bridge between two elements. Since chemical compounds are not usually measured with high-throughput technology, they should be removed from the network to analyse gene signals. However, the trivial elimination of the compounds, without signal propagation, will strongly bias the topology interrupting the signals that pass through them. If element 'A' is linked to compound 'c' and compound 'c' is linked to element 'B', element 'A' should be linked to element 'B'. Not all compounds are considered for the propagation because some of them (for example: H₂O, ATP, ADP) are highly frequent in map descriptions and the signal propagation through them would lead to chains too long. Compounds not considered for propagation are not characteristic of a specific reaction, but act as secondary substrates/products widely shared among different processes.

graphite allows the user to see the single/multiple relation types that characterized an edge. The type of edges have been kept as much as possible similar to those annotated in the original data format. Some new types have been introduced due to topological conversion needs.

See Also

[pathway-class](#).

pathway-class	Class "pathway"
---------------	-----------------

Description

This class represents a biological pathway.

Slots

title: Object of class "vector".

nodes: Object of class "vector".

edges: Object of class "data.frame". The data.frame has one row for each edge and four columns: source node, destination node, edge direction and type.

ident: Object of class "vector". The type of identifiers used to label the nodes. The value of "native" means that the identifiers are those present in the original pathway definition.

See [convertIdentifiers](#) to convert them.

database: Object of class "vector". The name of the database from which the pathway was retrieved.

species: Object of class "character". The name of the species the pathway belongs to.

timestamp: Object of class "Date". The date when the pathway was retrieved.

Methods

edges signature(object = "pathway"): retrieves the data.frame describing the pathway edges.

nodes signature(object = "pathway"): retrieves the vector enumerating the identifiers of the pathway nodes.

pathwayGraph

Graph representing the topology of a pathway

Description

Builds a graphNEL object representing the topology of a pathway.

Usage

```
pathwayGraph(pathway, edge.types=NULL)
```

Arguments

pathway a pathway object.

edge.types keep only the edges matching the type names in this vector.

Value

A graphNEL object.

See Also

[pathway-class](#)

[graphNEL-class](#)

Examples

```
pathwayGraph(reactome$mTOR signalling, edge.types=c("binding"))
```

prepareSPIA	<i>Prepare pathway dataset needed by runSPIA.</i>
-------------	---

Description

Prepare pathway dataset needed by runSPIA. See [runSPIA](#) and [spia](#) for more details.

Usage

```
prepareSPIA(db, pathwaySetName, print.names = FALSE)
```

Arguments

db	A list of pathways like biocarta , kegg , nci or reactome .
pathwaySetName	Name of the pathway set in output.
print.names	The verbose option.

Details

It is required to run this function before [runSPIA](#) to prepare the pathway set as required by runSPIA.

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.

See Also

[runSPIA](#)

reactome	<i>Reactome pathways</i>
----------	--------------------------

Description

A list with a [pathway](#) object for each Reactome pathway.

Matthews L, Gopinath G, Gillespie M, Caudy M, Croft D, de Bono B, Garapati P, Hemish J, Hermjakob H, Jassal B, Kanapin A, Lewis S, Mahajan S, May B, Schmidt E, Vastrik I, Wu G, Birney E, Stein L, D'Eustachio P. Reactome knowledgebase of human biological pathways and processes. *Nucleic Acids Res*. 2009 Jan;37(Database issue):D619-22. Epub 2008 Nov 3.

<http://www.reactome.org>

Pathway topology conversion

Reactome pathways were retrieved in the BioPax format from the Reactome database web page.

We define a pathway for each BioPax tag “pathway”. Pathway nodes often correspond to multiple gene products. These can be divided into protein complexes (proteins linked by protein-protein interactions) and groups made of alternative members (genes with similar biochemical functions). Thus, when considering signal propagation these groups are considered differently. The first kind (hereafter group AND) should be expanded into a clique (all proteins connected to the others), while the second (hereafter group OR) should be expanded without connection among them. In the BioPax format only one type of group is allowed: protein complexes (group AND) with the tag 'complex'. However, it often happens that the 'protein' tag contains multiple 'xref' pointing to alternative elements of the process (group OR).

Compound mediated interactions are interactions for which a compound acts as a bridge between two elements. Since chemical compounds are not usually measured with high-throughput technology, they should be removed from the network to analyse gene signals. However, the trivial elimination of the compounds, without signal propagation, will strongly bias the topology interrupting the signals that pass through them. If element 'A' is linked to compound 'c' and compound 'c' is linked to element 'B', element 'A' should be linked to element 'B'. Not all compounds are considered for the propagation because some of them (for example: H₂O, ATP, ADP) are highly frequent in map descriptions and the signal propagation through them would lead to chains too long. Compounds not considered for propagation are not characteristic of a specific reaction, but act as secondary substrates/products widely shared among different processes.

graphite allows the user to see the single/multiple relation types that characterized an edge. The type of edges have been kept as much as possible similar to those annotated in the original data format. Some new types have been introduced due to topological conversion needs.

See Also

[pathway-class](#).

runClipper

Run a topological analysis on an expression dataset using the clipper.

Description

clipper is a package 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

```
runClipper(pathway, expr, classes, method, ...)
```

Arguments

pathway	One of the pathways contained in biocarta , kegg , nci , reactome or spike .
expr	A matrix (size: number p of genes x number n of samples) of gene expression.
classes	A vector (length: n) of class assignments.
method	The kind of test to perform on the cliques. It could be either mean or variance.
...	Additional options see for details easyClip .

Details

The expression data and the pathway have to be annotated in the same set of identifiers.

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.

See Also

[clipper](#)

Examples

```
if (require(clipper) & require(ALL)){
  path <- convertIdentifiers(kegg$Chronic myeloid leukemia, "entrez")
  genes <- nodes(path)
  data(ALL)
  all <- as.matrix(exprs(ALL[1:length(genes),1:20]))
  classes <- c(rep(1,10), rep(2,10))
  rownames(all) <- genes
  runClipper(path, all, classes, "mean", pathThr=0.1)
}
```

runClipperMulti	<i>Run topological analyses on an expression dataset, calling clipper over all the pathways in a list.</i>
-----------------	--

Description

clipper is a package 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. This function easily provide pathway analysis using clipper on an entire pathway database.

Usage

```
runClipperMulti(pathways, expr, classes, method, maxNodes=150, ...)
```

Arguments

pathways	A list of pathways, such as biocarta , kegg,nci , reactome or spike .
expr	A matrix (size: number p of genes x number n of samples) of gene expression.
classes	A vector (length: n) of class assignments.
method	The kind of test to perform on the cliques. It could be either mean or variance.
maxNodes	Ignore pathways with more than "maxNodes" nodes. Set to "NULL" to disable the filter.
...	Additional options for details see clipper .

Details

The expression data and the pathway have to be annotated in the same set of identifiers.

Value

A list with two elements:

- results: a list with one entry for each successfully analyzed pathway;
- errors: a vector containing the error messages of failed analyses.

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.

See Also

[clipper](#)

Examples

```
if (require(clipper) & require(ALL)){
  paths <- lapply(kegg[1:5], function(x) convertIdentifiers(x, "entrez"))
  genes <- unlist(lapply(paths, nodes))
  data(ALL)
  all <- as.matrix(exprs(ALL[1:length(genes),1:20]))
  classes <- c(rep(1,10), rep(2,10))
  rownames(all) <- genes
  clipped <- runClipperMulti(paths, all, classes, "mean", pathThr=0.1)
}
```

runDEGraph	<i>Run a topological analysis on an expression dataset using DEGraph package.</i>
------------	---

Description

DEGraph implements recent hypothesis testing methods which directly assess whether a particular gene network is differentially expressed between two conditions.

Usage

```
runDEGraph(pathway, expr, classes)
```

Arguments

pathway	One of the pathways contained in biocarta , kegg.nci or reactome .
expr	A matrix (size: number p of genes x number n of samples) of gene expression.
classes	A vector (length: n) of class assignments.

Details

The expression data and the pathway have to be annotated in the same set of identifiers.

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.

See Also

[testOneGraph](#)

Examples

```
if (require(DEGraph)) {  
  data("Loi2008_DEGraphVignette")  
  
  p <- convertIdentifiers(biocarta[["actions of nitric oxide in the heart"]], "entrez")  
  runDEGraph(p, exprLoi2008, classLoi2008)  
}
```

runDEGraphMulti	<i>Run topological analyses on an expression dataset, calling DEGraph over all the pathways in a list.</i>
-----------------	--

Description

DEGraph implements recent hypothesis testing methods which directly assess whether a particular gene network is differentially expressed between two conditions.

Usage

```
runDEGraphMulti(pathways, expr, classes, maxNodes=150)
```

Arguments

pathways	A list of pathways, such as biocarta , kegg,nci or reactome .
expr	A matrix (size: number p of genes x number n of samples) of gene expression.
classes	A vector (length: n) of class assignments.
maxNodes	Ignore pathways with more than "maxNodes" nodes. Set to "NULL" to disable the filter.

Details

The expression data and the pathway have to be annotated in the same set of identifiers.

Value

A list with two elements:

- results: a list with one entry for each successfully analyzed pathway;
- errors: a vector containing the error messages of failed analyses.

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.

See Also

[testOneGraph](#)

Examples

```

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

  ps <- lapply(biocarta[1:3],
              function(p) convertIdentifiers(p, "entrez"))
  runDEGraphMulti(ps, exprLoi2008, classLoi2008)
}

```

runSPIA

*Run SPIA analysis***Description**

Run a topological analysis on an expression dataset using SPIA.

Usage

```
runSPIA(de, all, pathwaySetName, ...)
```

Arguments

de	A named vector containing log ₂ fold-changes of the differentially expressed genes. The names of this numeric vector are Entrez gene IDs.
all	A vector with the Entrez IDs in the reference set. If the data was obtained from a microarray experiment, this set will contain all genes present on the specific array used for the experiment. This vector should contain all names of the 'de' argument.
pathwaySetName	A list of pathways like kegg , nci or reactome .
...	Additional options to pass to spia .

Details

The spia option "organism" is internally used. It is an error use it in the additional options.

Value

The same of spia, without KEGG links. A data frame containing the ranked pathways and various statistics: pSize is the number of genes on the pathway; NDE is the number of DE genes per pathway; tA is the observed total perturbation accumulation in the pathway; pNDE is the probability to observe at least NDE genes on the pathway using a hypergeometric model; pPERT is the probability to observe a total accumulation more extreme than tA only by chance; pG is the p-value obtained by combining pNDE and pPERT; pGFdr and pGFWER are the False Discovery Rate and respectively Bonferroni adjusted global p-values; and the Status gives the direction in which the pathway is perturbed (activated or inhibited).

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.

See Also

[spia](#)

Examples

```
if (require(SPIA) && require(hgu133plus2.db)) {
  data(colorectalcancer)

  x <- hgu133plus2ENTREZID
  top$ENTREZ <- unlist(as.list(x[top$ID]))
  top <- top[!is.na(top$ENTREZ), ]
  top <- top[!duplicated(top$ENTREZ), ]
  tg1 <- top[top$adj.P.Val < 0.05, ]

  DE_Colorectal = tg1$logFC
  names(DE_Colorectal) <- as.vector(tg1$ENTREZ)
  ALL_Colorectal <- top$ENTREZ

  prepareSPIA(biocarta[1:20], "biocartaEx")
  runSPIA(de=DE_Colorectal, all=ALL_Colorectal, "biocartaEx")
}
```

runTopologyGSA

Run a topological analysis on an expression dataset using topologyGSA.

Description

Use graphical models to test the pathway components highlighting those involved in its deregulation.

Usage

```
runTopologyGSA(pathway, test, exp1, exp2, alpha, ...)
```

Arguments

pathway	One of the pathways contained in biocarta , kegg,nci or reactome .
test	Either "var" and "mean". Determine the type of test used by topologyGSA.
exp1	Experiment matrix of the first class, genes in columns.
exp2	Experiment matrix of the second class, genes in columns.
alpha	Significance level of the test.
...	Additional parameters.

Details

This function produces a warning and returns NULL when the number of genes in common between the expression matrices and the pathway is less than 3.

References

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

See Also

[pathway.var.test](#) [pathway.mean.test](#)

Examples

```
if (require(topologyGSA)) {
  data(examples)

  p <- convertIdentifiers(kegg[["Fc epsilon RI signaling pathway"]], "symbol")
  runTopologyGSA(p, "var", y1, y2, 0.05)
}
```

runTopologyGSAMulti *Run topological analyses on an expression dataset, calling topologyGSA over all the pathways in a list.*

Description

Use graphical models to test the pathway components highlighting those involved in its deregulation.

Usage

```
runTopologyGSAMulti(pathways, test, exp1, exp2, alpha, maxNodes=150, ...)
```

Arguments

pathways	A list of pathways, such as biocarta , kegg , nci or reactome .
test	Either "var" and "mean". Determine the type of test used by topologyGSA.
exp1	Experiment matrix of the first class, genes in columns.
exp2	Experiment matrix of the second class, genes in columns.
alpha	Significance level of the test.
maxNodes	Ignore pathways with more than "maxNodes" nodes. Set to "NULL" to disable the filter.
...	Additional parameters for topologyGSA.

Details

This function produces a warning and whenever the number of genes in common between the expression matrices and a pathway is less than 3.

Value

A list with two elements:

- results: a list with one entry for each successfully analyzed pathway;
- errors: a vector containing the error messages of failed analyses.

References

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

See Also

[pathway.var.test](#) [pathway.mean.test](#)

Examples

```
if (require(topologyGSA)) {
  data(examples)

  ps <- lapply(kegg[c("Acute myeloid leukemia", "Fc epsilon RI signaling pathway")],
              function(p) convertIdentifiers(p, "symbol"))
  runTopologyGSAMulti(ps, "var", y1, y2, 0.05)
}
```

spike

SPIKE pathways

Description

A list with a [pathway](#) object for each SPIKE pathway.

Arnon Paz, Zippora Brownstein, Yaara Ber, Shani Bialik, Eyal David, Dorit Sagir, Igor Ulitsky, Ran Elkou, Adi Kimchi, Karen B. Avraham, Yosef Shiloh and Ron Shamir SPIKE: a database of highly curated human signaling pathways. Nucleic Acids Research, 2011, Vol. 39, Database issue

<http://www.cs.tau.ac.il/~spike/>

Pathway topology conversion

SPIKE pathways were retrieved in the BioPax format from the SPIKE database web site.

We define a pathway for each BioPax tag “pathway”. Pathway nodes often correspond to multiple gene products. These can be divided into protein complexes (proteins linked by protein-protein interactions) and groups made of alternative members (genes with similar biochemical functions). Thus, when considering signal propagation these groups are considered differently. The first kind (hereafter group AND) should be expanded into a clique (all proteins connected to the others), while the second (hereafter group OR) should be expanded without connection among them. In the BioPax format only one type of group is allowed: protein complexes (group AND) with the tag ‘complex’. However, it often happens that the ‘protein’ tag contains multiple ‘xref’ pointing to alternative elements of the process (group OR).

Compound mediated interactions are interactions for which a compound acts as a bridge between two elements. Since chemical compounds are not usually measured with high-throughput technology, they should be removed from the network to analyse gene signals. However, the trivial elimination of the compounds, without signal propagation, will strongly bias the topology interrupting the signals that pass through them. If element ‘A’ is linked to compound ‘c’ and compound ‘c’ is linked to element ‘B’, element ‘A’ should be linked to element ‘B’. Not all compounds are considered for the propagation because some of them (for example: H₂O, ATP, ADP) are highly frequent in map descriptions and the signal propagation through them would lead to chains too long. Compounds not considered for propagation are not characteristic of a specific reaction, but act as secondary substrates/products widely shared among different processes.

graphite allows the user to see the single/multiple relation types that characterized an edge. The type of edges have been kept as much as possible similar to those annotated in the original data format. Some new types have been introduced due to topological conversion needs.

See Also

[pathway-class](#).

Index

- *Topic **DEGraph**
 - runDEGraph, 14
 - runDEGraphMulti, 15
- *Topic **analysis**
 - runClipper, 11
 - runClipperMulti, 12
 - runDEGraph, 14
 - runDEGraphMulti, 15
 - runSPIA, 16
 - runTopologyGSA, 17
 - runTopologyGSAMulti, 18
- *Topic **classes**
 - pathway-class, 8
- *Topic **clipper**
 - runClipper, 11
 - runClipperMulti, 12
- *Topic **spia**
 - runSPIA, 16
- *Topic **topologyGSEA**
 - runTopologyGSA, 17
 - runTopologyGSAMulti, 18
- *Topic **topology**
 - runClipper, 11
 - runClipperMulti, 12
 - runDEGraph, 14
 - runDEGraphMulti, 15
 - runSPIA, 16
 - runTopologyGSA, 17
 - runTopologyGSAMulti, 18
- biocarta, 2, 3, 10, 12–15, 18, 19
- clipper, 12, 13
- convertIdentifiers, 3, 8
- cytoscapePlot, 4
- easyClip, 12
- edges, pathway, ANY-method
(pathway-class), 8
- humancyc, 4
- kegg, 3, 5, 10, 12–16, 18, 19
- nci, 3, 6, 10, 12–16, 18, 19
- nodes, pathway-method (pathway-class), 8
- panther, 7
- pathway, 2, 4–7, 10, 20
- pathway-class, 8
- pathway.mean.test, 18, 19
- pathway.var.test, 18, 19
- pathwayGraph, 4, 9
- prepareSPIA, 10
- reactome, 3, 10, 10, 12–16, 18, 19
- runClipper, 11
- runClipperMulti, 12
- runDEGraph, 14
- runDEGraphMulti, 15
- runSPIA, 10, 16
- runTopologyGSA, 17
- runTopologyGSAMulti, 18
- show, pathway-method (pathway-class), 8
- spia, 10, 16, 17
- spike, 12, 13, 20
- testOneGraph, 14, 15