Package ‘clipper’

March 22, 2017

Version 1.15.0
Date 2016-09-04
Title Gene Set Analysis Exploiting Pathway Topology
Author Paolo Martini <paolo.cavei@gmail.com>, Gabriele Sales <gabriele.sales@unipd.it>, Chiara Romualdi <chiara.romualdi@unipd.it>
Maintainer Paolo Martini <paolo.cavei@gmail.com>
Description Implements topological gene set analysis using a two-step empirical approach. It exploits graph decomposition theory to create a junction tree and 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.
Depends R (>= 2.15.0), Matrix, graph
Imports methods, Biobase, Rcpp, igraph, gRbase (>= 1.6.6), qpgraph, KEGGgraph, corpcor, RBGL
Suggests RUnit, BiocGenerics, graphite, ALL, hgu95av2.db, MASS, BiocStyle
Enhances RCytoscape (>= 1.6.3)
License AGPL-3
NeedsCompilation no

R topics documented:

clipper .................................................. 2
clipperAllRoots ...................................... 4
cliqueMeanTest ...................................... 5
cliqueMixedTest .................................... 7
cliquePairedTest ................................... 8
cliqueVarianceTest ................................. 9
deleteEdge .......................................... 10
easyClip ............................................ 11
easyLook ........................................... 12
getGraphEntryGenes ............................... 13
getJunctionTreePaths .............................. 14
nameCliques ....................................... 14
Dissect the pathway to find the path with the greatest association with phenotype.

**Description**

Basing on either variance or mean clique test, this function identifies the paths that are mostly related with the phenotype under study.

**Usage**

```r
clipper(expr, classes, graph, method=c("variance","mean", "both", "paired"), nperm=100, alphaV=0.05, b=100, root=NULL, trZero=0.001, signThr=0.05, maxGap=1, permute=TRUE, alwaysShrink=FALSE)
```

**Arguments**

- `expr`: an expression matrix or ExpressionSet with column names for samples and row name for genes.
- `classes`: vector of 1,2 indicating the classes of samples (columns).
- `graph`: a graphNEL object.
- `method`: the kind of test to perform on the cliques. It could be mean, variance, mixed (the best between variance and mean) or paired mean.
- `nperm`: number of permutations. Default = 100.
- `alphaV`: pvalue threshold for variance test to be used during mean test. Default = 0.05.
- `b`: number of permutations for mean analysis. Default = 100.
- `root`: nodes by which rip ordering is performed (as far as possible) on the variables using the maximum cardinality search algorithm.
- `trZero`: lowest pvalue detectable. This threshold avoids that -log(p) goes infinite.
- `signThr`: significance threshold for clique pvalues.
- `maxGap`: allow up to maxGap gaps in the best path computation. Default = 1.
- `permute`: always performs permutations in the concentration matrix test. If FALSE, the test is made using the asymptotic distribution of the log-likelihood ratio. This option should be use only if samples size is >=40 per class.
- `alwaysShrink`: always perform the shrinkage estimates of variance.

**Details**

The both method combines the results obtained from the mean and variance test. In particular it assign to the cliques the minimum of mean and variance p-values.
**Value**

A matrix with a row for each paths. Columns are organized as follows:

1. Index of the starting clique
2. Index of the ending clique
3. Index of the clique where the maximum value is reached
4. Length of the path
5. Maximum score of the path
6. Average score along the path
7. Percentage of path activation
8. Impact of the path on the entire pathway
9. Cliques involved and significant
10. Cliques forming the path
11. Genes forming the significant cliques
12. Genes forming the path

**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**

`cliqueVarianceTest, cliqueMeanTest, getJunctionTreePaths`

**Examples**

```r
if (require(graphite) & require(ALL)){
  kegg  <- pathways("hsapiens", "kegg")
  graph <- pathwayGraph(convertIdentifiers(kegg$'Chronic myeloid leukemia', "entrez"))
  genes <- nodes(graph)
  data(ALL)
  all <- ALL[1:length(genes),1:20]
  classes <- c(rep(1,10), rep(2,10))
  featureNames(all@assayData)<- genes
  graph <- subGraph(genes, graph)
  clipped <- clipper(all, classes, graph, "var", trZero=0.01, permute=FALSE)
  clipped[,1:5]
}
```
Dissect the pathway to find the path with the greatest association with phenotype.

**Description**

Basing on either variance or mean clique test, this function identifies the paths that are mostly related with the phenotype under study.

**Usage**

```r
clipperAllRoots(expr, classes, graph, method=c("variance","mean", "both", "paired"), nperm=100, alphaV=0.05, b=100, trZero=0.001, signThr=0.05, maxGap=1, permute=TRUE, alwaysShrink=FALSE)
```

**Arguments**

- `expr` an expression matrix or ExpressionSet with colnames for samples and row name for genes.
- `classes` vector of 1,2 indicating the classes of samples (columns).
- `graph` a graphNEL object.
- `method` the kind of test to perform on the cliques. It could be mean, variance, mixed (the best between variance and mean) or paired mean.
- `nperm` number of permutations. Default = 100.
- `alphaV` pvalue threshold for variance test to be used during mean test. Default = 0.05.
- `b` number of permutations for mean analysis. Default = 100.
- `trZero` lowest pvalue detectable. This threshold avoids that -log(p) goes infinite.
- `signThr` significance threshold for clique pvalues.
- `maxGap` allow up to maxGap gaps in the best path computation. Default = 1.
- `permute` always performs permutations in the concentration matrix test. If FALSE, the test is made using the asymptotic distribution of the log-likelihood ratio. This option should be use only if samples size is >=40 per class.
- `alwaysShrink` always perform the shrinkage estimates of variance.

**Details**

The both method combines the results obtained from the mean and variance test. In particular it assign to the cliques the minimum of mean and variance p-values.

**Value**

A matrix with a row for each paths. Rownames have the form: roots-paths.

Columns are organized as follows:

1. Index of the starting clique
2. Index of the ending clique
cliqueMeanTest

Mean test for cliques.

Description

It decomposes the graph in cliques and performs the mean test in every one.

Usage

cliqueMeanTest(expr, classes, graph, nperm, alphaV=0.05, b=100, root=NULL, permute=TRUE, alwaysShrink=FALSE)

References


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

- **expr**: an expression matrix or ExpressionSet with colnames for samples and row name for genes.
- **classes**: vector of 1,2 indicating the classes of samples (columns).
- **graph**: a graphNEL object.
- **nperm**: number of permutations.
- **alphaV**: pvalue threshold for variance test to be used during mean test.
- **b**: number of permutations for mean analysis.
- **root**: nodes by which rip ordering is performed (as far as possible) on the variables using the maximum cardinality search algorithm.
- **permute**: always performs permutations in the concentration matrix test. If FALSE, the test is made using the asymptotic distribution of the log-likelihood ratio. This option should be use only if samples size is >=40 per class.
- **alwaysShrink**: always perform the shrinkage estimates of variance.

### Value

A list with alphas (vector of cliques pvalues based on the mean test) and cliques (list of the cliques and related elements).

### 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

`cliqueVarianceTest`.

### Examples

```r
if (require(graphite) & require(ALL)){
  kegg <- pathways("hsapiens", "kegg")
  graph <- pathwayGraph(convertIdentifiers(kegg$"Chronic myeloid leukemia", "entrez"))
  genes <- nodes(graph)
  data(ALL)
  all <- ALL[1:length(genes),1:20]
  classes <- c(rep(1,10), rep(2,10))
  featureNames(all@assayData)<- genes
  graph <- subGraph(genes, graph)
  cliqueMeanTest(all, classes, graph, nperm=100, permute=FALSE)$alpha
}
```
cliqueMixedTest

Mean test for cliques.

Description
It decomposes the graph in cliques and performs the combination of mean and variance test in every one.

Usage
cliqueMixedTest(expr, classes, graph, nperm, alphaV=0.05, b=100, root=NULL, permute=TRUE, alwaysShrink=FALSE)

Arguments
- **expr**: an expression matrix or ExpressionSet with colnames for samples and row names for genes.
- **classes**: vector of 1,2 indicating the classes of samples (columns).
- **graph**: a graphNEL object.
- **nperm**: number of permutations.
- **alphaV**: p-value threshold for variance test to be used during mean test.
- **b**: number of permutations for mean analysis.
- **root**: nodes by which rip ordering is performed (as far as possible) on the variables using the maximum cardinality search algorithm.
- **permute**: always performs permutations in the concentration matrix test. If FALSE, the test is made using the asymptotic distribution of the log-likelihood ratio. This option should be used only if samples size is >=40 per class.
- **alwaysShrink**: always perform the shrinkage estimates of variance.

Details
The method combines the results obtained from the mean and variance test. In particular, it assigns to the cliques the minimum of mean and variance p-values.

Value
a list with alphas (vector of cliques p-values based on the variance test) and cliques (list of the cliques and related elements).

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
cliqueVarianceTest.
cliquePairedTest

Paired mean test for cliques.

Description

It decomposes the graph in cliques and performs the paired mean test in every one.

Usage

cliquePairedTest(expr, classes, graph, nperm, alphaV=0.05, b=100, root=NULL, permute=TRUE, alwaysShrink=FALSE)

Arguments

expr    an expression matrix or ExpressionSet with colnames for samples and row name for genes.
classes vector of 1,2 indicating the classes of samples (columns). It is assumed that class labels are ordered so that the first occurrence of class 2 is paired with the first occurrence of class 1 and so on.
graph   a graphNEL object.
nperm   number of permutations.
alphaV  pvalue threshold for variance test to be used during mean test.
b       number of permutations for mean analysis.
root    nodes by which rip ordering is performed (as far as possible) on the variables using the maximum cardinality search algorithm.
permute always performs permutations in the concentration matrix test. If FALSE, the test is made using the asymptotic distribution of the log-likelihood ratio. This option should be use only if samples size is >=40 per class.
alwaysShrink always perform the shrinkage estimates of variance.

Value

a list with alphas (vector of cliques pvalues based on the variance test) and cliques (list of the cliques and related elements).
cliqueVarianceTest

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
cliqueVarianceTest.

Examples
if (require(graphite) & require(ALL)){
  kegg <- pathways("hsapiens", "kegg")
  graph <- pathwayGraph(convertIdentifiers(kegg$'Chronic myeloid leukemia', "entrez"))
  genes <- nodes(graph)
  data(ALL)
  all <- ALL[1:length(genes),1:20]
  classes <- c(rep(1,10), rep(2,10))
  featureNames(all@assayData)<- genes
  graph <- subGraph(genes, graph)
  cliquePairedTest(all, classes, graph, nperm=100, permute=FALSE)$alpha
}

cliqueVarianceTest  
Variance test for cliques.

Description
It decomposes the graph in cliques and performs the variance test in every one.

Usage
cliqueVarianceTest(expr, classes, graph, nperm, alphaV=0.05,
b=100, root=NULL, permute=TRUE, alwaysShrink=FALSE)

Arguments
expr an expression matrix or ExpressionSet with colnames for samples and row name for genes.
classes vector of 1,2 indicating the classes of samples (columns).
graph a graphNEL object.
nperm number of permutations.
alphaV pvalue threshold for variance test to be used during mean test.
b number of permutations for mean analysis.
root nodes by which rip ordering is performed (as far as possible) on the variables using the maximum cardinality search algorithm.
permute always performs permutations in the concentration matrix test. If FALSE, the test is made using the asymptotic distribution of the log-likelihood ratio. This option should be use only if samples size is >=40 per class.
alwaysShrink always perform the shrinkage estimates of variance.
deleteEdge

Value

a list with alphas (vector of cliques p-values based on the variance test) and cliques (list of the cliques and related elements).

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

cliqueMeanTest.

Examples

```r
if (require(graphite) & require(ALL)){
  kegg <- pathways("hsapiens", "kegg")
  graph <- pathwayGraph(convertIdentifiers(kegg$"Chronic myeloid leukemia", "entrez"))
  genes <- nodes(graph)
  data(ALL)
  all <- ALL[1:length(genes),1:20]
  classes <- c(rep(1,10), rep(2,10))
  featureNames(all@assayData)<- genes
  graph <- subGraph(genes, graph)
  cliqueVarianceTest(all, classes, graph, nperm=100, permute=FALSE)$alpha
}
```

deleteEdge

Remove an edge from graphNEL object.

Description

Remove from a graphNEL object the edge specified.

Usage

deleteEdge(graph, from, to)

Arguments

graph a graphNEL object.
from a string with the name of the node where the edge start.
to a string with the name of the node where the edge end.

Value

a graphNEL object.
Examples

```r
if (require(graphite)) {
  kegg <- pathways("hsapiens", "kegg")
  graph <- pathwayGraph(convertIdentifiers(kegg$'Chronic myeloid leukemia', "entrez"))
  head(edges(graph))
  ## We are going to remove the edge 1026-1019
  head(edges(deleteEdge(graph, "1026", "1019")))
}
```

---

easyClip  
Easy clip analysis.

description

Easy clip function allows the full exploitation of Clipper Package features in a unique and easy to use function. Starting from an expression matrix and a pathway, these function extract the most transcriptionally altered portions of the graph.

Usage

```r
easyClip(expr, classes, graph, method=c("variance","mean"),
  pathThr=0.05, pruneLevel=0.2, nperm=100, alphaV=0.05, b=100,
  root=NULL, trZero=0.001, signThr=0.05, maxGap=1, permute=TRUE)
```

Arguments

expr  
an expression matrix or ExpressionSet with colnames for samples and row name for genes.

classes  
vector of 1,2 indicating the classes of samples (columns).

graph  
a graphNEL object.

method  
the kind of test to perform on the cliques. It could be either mean or variance.

pathThr  
The significance threshold of the whole pathway test. Default = 0.05

pruneLevel  
da dissimilarity threshold. NULL means no pruning.

nperm  
number of permutations. Default = 100.

alphaV  
pvalue threshold for variance test to be used during mean test. Default = 0.05.

b  
number of permutations for mean analysis. Default = 100.

root  
nodes by which rip ordering is performed (as far as possible) on the variables using the maximum cardinality search algorithm.

trZero  
lowest pvalue detectable. This threshold avoids that -log(p) goes infinite.

signThr  
significance threshold for clique pvalues.

maxGap  
allow up to maxGap gaps in the best path computation. Default = 1.

permute  
always performs permutations in the concentration matrix test. If FALSE, the test is made using the asymptotic distribution of the log-likelihood ratio. This option should be use only if samples size is >=40 per class.
Value

a matrix with row as the different paths. Columns are organized as follows: 1 - Index of the starting clique 2 - Index of the ending clique 3 - Index of the clique where the maximum value is reached 4 - length of the path 5 - maximum score of the path 6 - average score along the path 7 - percentage of path activation 8 - impact of the path on the entire pathway 9 - clique involved and significant 10 - clique forming the path 11 - genes forming the significant cliques 12 - genes forming the path.

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

cliqueVarianceTest, cliqueMeanTest, getJunctionTreePaths

Examples

```r
if (require(graphite) & require(ALL)){
  kegg <- pathways("hsapiens", "kegg")
  graph <- pathwayGraph(convertIdentifiers(kegg$'Chronic myeloid leukemia', "entrez"))
  genes <- nodes(graph)
  data(ALL)
  all <- ALL[1:length(genes),1:20]
  classes <- c(rep(1,10), rep(2,10))
  featureNames(all@assayData)<- genes
  graph <- subGraph(genes, graph)
  clipped <- easyClip(all, classes, graph, nperm=10)
  clipped[,1:5]
}
```

```r
easyLook Summarize clipper output.
```

Description

Summarization of the result for a quick look of clipper function.

Usage

easyLook(clipped)

Arguments

collapsed the output of either clipper or easyClip.

Value

Nice formatted output of clipper function.
getGraphEntryGenes

References


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

description

It extracts the possible entry point of the graph. Entry points are defined as nodes with no entering edges.

Usage

getGraphEntryGenes(graph, byCliques=FALSE, root=NULL)

Arguments

graph a graphNEL object.
byCliques when TRUE it returns a list where entry point are organized by cliques.
root nodes by which rip ordering is performed (as far as possible) on the variables using the maximum cardinality search algorithm.

Value

a vector of gene names representing the entry point of graph.

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

if (require(graphite)) {
  kegg <- pathways("hsapiens", "kegg")
  graph <- pathwayGraph(convertIdentifiers(kegg$'Chronic myeloid leukemia', "entrez"))
  getGraphEntryGenes(graph)
}
**getJunctionTreePaths**  
Extract the shortest paths along the junction tree of the graph.

**Description**  
Find the shortest paths in the Junction tree designed with the cliques of the graph.

**Usage**  
```r  
getJunctionTreePaths(graph, root=NULL)  
```

**Arguments**  
- `graph` a `graphNEL` object.
- `root` nodes by which rip ordering is performed (as far as possible) on the variables using the maximum cardinality search algorithm.

**Value**  
list of clique indices representing the shortest paths of the graph.

**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**  
```r  
if (require(graphite)) {  
  kegg <- pathways("hsapiens", "kegg")  
  graph <- pathwayGraph(convertIdentifiers(kegg$'Chronic myeloid leukemia', "entrez"))  
  getJunctionTreePaths(graph)  
}
```

---

**nameCliques**  
Generate clique names from their own elements.

**Description**  
Starting from the sorted elements of each clique of the list, this function generates names fusing in a string the element names.

**Usage**  
```r  
nameCliques(cliques)  
```

**Arguments**  
- `cliques` a list where each element is a clique.
**Description**

Performs variance and mean test using qpipf on the whole pathway.

**Usage**

```r
pathQ(expr, classes, graph, nperm=100, alphaV=0.05, b=100,
permute=TRUE, paired=FALSE, alwaysShrink=FALSE)
```

**Arguments**

- `expr`: an expression matrix or ExpressionSet with colnames for samples and rownames for expression features.
- `classes`: vector of 1,2 indicating the classes of the samples (columns).
- `graph`: a graphNEL object.
- `nperm`: number of permutations. Default = 100.
- `alphaV`: pvalue significance threshold for variance test to be used during mean test. Default = 0.05.
- `b`: number of permutations for mean analysis. Default = 100.
- `permute`: always performs permutations in the concentration matrix test. If FALSE, the test is made using the asymptotic distribution of the log-likelihood ratio. This option should be use only if samples size is >=40 per class.
- `paired`: perform the test for paired sample. It assumes that class labels are ordered so that the first occurrence of class 2 is paired with the first occurrence of class 1 and so on.
- `alwaysShrink`: always perform the shrinkage estimates of variance.

**Value**

a list with alphaVar (pvalue for the variance test) and alphaMean (pvalue for mean test).

**Note**

This function is based on the Gaussian Graphical Models and to use it in a proper way it is necessary that the graph is an Direct Acyclic Graph. Please check any graph in input using `isAcyclic` from `ggm` package.
plotInCytoscape

Plot a pathway graph in Cytoscape highlighting the relevant path.

Description
Renders the topology of a pathway as a Cytoscape graph and marks the genes of the selected path.

Usage
plotInCytoscape(graph, path, color="#6699FF", main="graph", layout="jgraph-spring")

Arguments
- graph: a graphNEL object.
- path: vector summarizing a path (a rows of clipper output matrix).
- color: color code string: genes of the most involved fragment will be colored using color. Default = "#6699FF"
- main: a graph name to be used in Cytoscape. Default = 'graph'
- layout: a 'string' of choice among the values returned by 'getLayoutNames', default = 'jgraph-spring'

Details
Requires the RCytoscape package.

See Also
clipper

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
```r
if (require(graphite) & require(ALL)){
  kegg <- pathways("hsapiens", "kegg")
  graph <- pathwayGraph(convertIdentifiers(kegg$'Chronic myeloid leukemia', "entrez"))
  genes <- nodes(graph)
  data(ALL)
  all <- ALL[1:length(genes),1:20]
  classes <- c(rep(1,10), rep(2,10))
  featureNames(all@assayData)<- genes
  graph <- subGraph(genes, graph)
  pathQ(all, classes, graph, nperm=100, permute=FALSE)
}
```
prunePaths

Examples

```r
## Not run: if (require(graphite)) {
  if (requireNamespace("RCytoscape")){
    kegg <- pathways("hsapiens", "kegg")
    graph <- pathwayGraph(convertIdentifiers(kegg$Chronic myeloid leukemia', "entrez"))
  }
  path <-
    c(3,17,5,9,13.04,2.60,0.209,0.321,"6,7,8,9,10","3,5,6,7,8,9,10,14,17",
      "1029;4193;7157","1019;1021;1026;1029;595")
  plotInCytoscape(graph,path)
}
## End(Not run)
```

prunePaths

Summarize the paths obtained by clipper according to their similarity.

Description

This function allows the user to choose only one representative of those paths that have more than 1-thr similarity. The best scoring path is chosen.

Usage

```r
prunePaths(pathSummary, thr=NULL, clust=NULL, sep=";")
```

Arguments

- `pathSummary` a matrix resulting from clipper function.
- `thr` a dissimilarity threshold. `NULL` means no pruning.
- `clust` filename where path-cluster is saved. `NULL` means no cluster saved.
- `sep` the separator to split genes for similarity computation. Default = ;

Value

a matrix

See Also

clipper

Examples

```r
toyEx <- matrix(c(1,1,5,3,5,2,5,3,8,2,3,2,1,0.3,0.1,2,1,"1;2;3;4;5","1;2;3",
                  "1;2;3;4;5","1;2;3","1;2;3;4;5","1;2;3","1;2;3;4;5","1;2;3"),2,12)
row.names(toyEx) <- c("1;5","1;3")
toyEx
prunePaths(toyEx, thr=0.1)
```
Index

clipper, 2, 16, 17
clipperAllRoots, 4
cliqueMeanTest, 3, 5, 5, 10, 12
cliqueMixedTest, 7
cliquePairedTest, 8
cliqueVarianceTest, 3, 5–7, 9, 9, 12
deleteEdge, 10
easyClip, 11
easyLook, 12
getGraphEntryGenes, 13
getJunctionTreePaths, 3, 5, 12, 14

nameCliques, 14

path\(Q\) (pathwayTest), 15
pathwayTest, 15
plotInCytoscape, 16
prunePaths, 17