Package ‘StarBioTrek’

May 30, 2024

<table>
<thead>
<tr>
<th>Type</th>
<th>Package</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>StarBioTrek</td>
</tr>
<tr>
<td>Version</td>
<td>1.30.0</td>
</tr>
<tr>
<td>Date</td>
<td>04-16-2019</td>
</tr>
<tr>
<td>Author</td>
<td>Claudia Cava, Isabella Castiglioni</td>
</tr>
<tr>
<td>Maintainer</td>
<td>Claudia Cava <a href="mailto:claudia.cava@ibfm.cnr.it">claudia.cava@ibfm.cnr.it</a></td>
</tr>
<tr>
<td>Depends</td>
<td>R (&gt;= 3.3)</td>
</tr>
<tr>
<td>Imports</td>
<td>SpidermiR, graphite, AnnotationDbi, e1071, ROCr, MLmetrics, grDevices, igraph, reshape2, ggplot2</td>
</tr>
<tr>
<td>Description</td>
<td>This tool StarBioTrek presents some methodologies to measure pathway activity and cross-talk among pathways integrating also the information of network data.</td>
</tr>
<tr>
<td>License</td>
<td>GPL (&gt;= 3)</td>
</tr>
<tr>
<td>biocViews</td>
<td>GeneRegulation, Network, Pathways, KEGG</td>
</tr>
<tr>
<td>Suggests</td>
<td>BiocStyle, knitr, rmarkdown, testthat, devtools, roxygen2, qgraph, png, grid</td>
</tr>
<tr>
<td>VignetteBuilder</td>
<td>knitr</td>
</tr>
<tr>
<td>LazyData</td>
<td>true</td>
</tr>
<tr>
<td>URL</td>
<td><a href="https://github.com/claudiacava/StarBioTrek">https://github.com/claudiacava/StarBioTrek</a></td>
</tr>
<tr>
<td>BugReports</td>
<td><a href="https://github.com/claudiacava/StarBioTrek/issues">https://github.com/claudiacava/StarBioTrek/issues</a></td>
</tr>
<tr>
<td>RoxygenNote</td>
<td>6.1.1</td>
</tr>
<tr>
<td>PackageStatus</td>
<td>Deprecated</td>
</tr>
<tr>
<td>git_url</td>
<td><a href="https://git.bioconductor.org/packages/StarBioTrek">https://git.bioconductor.org/packages/StarBioTrek</a></td>
</tr>
<tr>
<td>git_branch</td>
<td>RELEASE_3_19</td>
</tr>
<tr>
<td>git_last_commit</td>
<td>ab618a7</td>
</tr>
<tr>
<td>git_last_commit_date</td>
<td>2024-04-30</td>
</tr>
<tr>
<td>Repository</td>
<td>Bioconductor 3.19</td>
</tr>
<tr>
<td>Date/Publication</td>
<td>2024-05-29</td>
</tr>
</tbody>
</table>
average

For TCGA data get human pathway data and creates a matrix with the average of genes for each pathway.

Description

average creates a matrix with a summarized value for each pathway

Usage

average(pathwayexpsubset)
circleplot

Arguments

pathwayexpsubset

list of pathway data

Value

a matrix value for each pathway

Examples

list_path_gene<-GE_matrix(DataMatrix=Data_CANCER_normUQ_fil,genes.by.pathway=pathway[1:50])
score_mean<-average(pathwayexpsubset=list_path_gene)

Preparation for circle plot

circleplot

Description

circleplot function takes as input data derived by the function plotcrosstalk and plot a circle plot.

Usage

circleplot(preplot, scoregene)

Arguments

preplot a list as obtained from the function plotcrosstalk
scoregene a score for each gene with values included between -10 e +10

Value

a list with correlation matrix and gene set for each gene

Examples

formatplot<-plotcrosstalk(pathway_plot=pathway[1:6],gs_expre=tumo)
score<-runif(length(formatplot[[2]]), min=-10, max=+10)
circleplot(preplot=formatplot, scoregene=score)
Description

GetPathNet creates a list of genes inside the pathways.

Usage

ConvertedIDgenes(path_ALL)

Arguments

path_ALL variable. The user can select the variable as obtained by GetData function

Value

a list of pathways

Examples

pathway<-ConvertedIDgenes(path_ALL=path[1:3])

Description

pathway data list

Format

A dataframe with gene expression profiles
**dsscorecrtlk**

*For TCGA data get human pathway data and creates a measure of discriminating score among pathways*

---

**Description**

dsscorecrtlk creates a matrix with discriminating score for pathways

**Usage**

dsscorecrtlk(dataFilt, pathway_exp)

**Arguments**

dataFilt TCGA matrix
pathway_exp a list of pathway data

**Value**
a matrix value for each pathway

**Examples**

cross_talk_st_dv<-dsscorecrtlk(dataFilt=tumo[,1:2],pathway_exp=pathway[1:5])

---

**eucdistcrtlk**

*For TCGA data get human pathway data and creates a measure of cross-talk among pathways*

---

**Description**
eucdistcrtlk creates a matrix with euclidean distance for pairwise pathways

**Usage**
eucdistcrtlk(dataFilt, pathway_exp)

**Arguments**
dataFilt TCGA matrix
pathway_exp list of pathway data

**Value**
a matrix value for each pathway
Examples

```r
score_euc_dist_t<-eucdistcrtlk(dataFilt=tumo[,1:2],pathway_exp=pathway[1:5])
```

---

**GetData**

*Get general information inside pathways.*

**Description**

GetData creates a list with genes inside the pathways.

**Usage**

```r
GetData(species, pathwaydb)
```

**Arguments**

- `species`: variable. The user can select the species of interest from `SELECT_path_species(path_spec)`
- `pathwaydb`: variable. The user can select the pathway database of interest from `SELECT_path_graphite(path_spec)`

**Value**

- a list of pathways

**Examples**

```r
# Not run:
species="hsapiens"
pathwaydb="pharmgkb"
path<-GetData(species,pathwaydb)
# End(Not run)
```

---

**getNETdata**

*Get network data from GeneMania.*

**Description**

genetdata creates a data frame with network data. Network category can be filtered among: physical interactions, co-localization, genetic interactions and shared protein domain.

**Usage**

genetdata(network, organismID = NULL)
**GetPathData**

*Arguments*

- **network** variable. The user can use the following parameters based on the network types to be used. PHint for Physical_interactions, COloc for Co-localization, GENint for Genetic_interactions and SHpd for Shared_protein_domains
- **organismID** organism==NULL default value is homo sapiens.

*Value*

list with gene-gene (or protein-protein interactions)

*Examples*

```r
## Not run:
organismID="Saccharomyces_cerevisiae"
netw<-getNETdata(network="SHpd",organismID)
## End(Not run)
```

---

**GetPathData**

*Get genes inside pathways.*

*Description*

GetPathData creates a list of genes inside the pathways.

*Usage*

```r
GetPathData(path_ALL)
```

*Arguments*

- **path_ALL** variable. The user can select the variable as obtained by GetData function

*Value*

- a list of pathways

*Examples*

```r
pathway_ALL_GENE<-GetPathData(path_ALL=path[1:3])
```
GetPathNet

Get interacting genes inside pathways.

Description

GetPathNet creates a list of genes inside the pathways.

Usage

GetPathNet(path_ALL)

Arguments

path_ALL variable. The user can select the variable as obtained by GetData function

Value

a list of pathways

Examples

pathway_net<-GetPathNet(path_ALL=path[1:3])

GE_matrix

Get human KEGG pathway data and a gene expression matrix in order to obtain a list with the gene expression for only pathways given in input.

Description

GE_matrix creates a list of gene expression for pathways given by the user.

Usage

GE_matrix(DataMatrix, genes.by.pathway)

Arguments

DataMatrix gene expression matrix (eg.TCGA data)

genes.by.pathway a list of pathway data as provided by GetData and ConvertedID_genes

Value

a list for each pathway (gene expression level belong to that pathway)
GE_matrix_mean

Examples

```
list_path_gene<-GE_matrix(DataMatrix=tumo[,1:2],genes.by.pathway=pathway[1:5])
```

GE_matrix_mean

*Get human KEGG pathway data and a gene expression matrix in order to obtain a matrix with the mean gene expression for only pathways given in input.*

Description

GE_matrix creates a matrix of mean gene expression levels for pathways given by the user.

Usage

```
GE_matrix_mean(DataMatrix, genes.by.pathway)
```

Arguments

- `DataMatrix`: gene expression matrix (e.g., TCGA data)
- `genes.by.pathway`: list of pathway data as provided by getKEGGdata

Value

a matrix for each pathway (mean gene expression level belong to that pathway)

Examples

```
list_path_plot<-GE_matrix_mean(DataMatrix=tumo[,1:2],genes.by.pathway=pathway[1:5])
```

GOChord

*Displays the relationship between genes and terms.*

Description

The GOChord function generates a circularly composited overview of selected/specific genes and their assigned processes or terms. More generally, it joins genes and processes via ribbons in an intersection-like graph.

Usage

```
GOChord(data, title, space, gene.order, gene.size, gene.space, nlfc = 1, lfc.col, lfc.min, lfc.max, ribbon.col, border.size, process.label, limit)
```
**Arguments**

- **data**: The matrix represents the binary relation (1= is related to, 0= is not related to) between a set of genes (rows) and processes (columns); a column for the logFC of the genes is optional.
- **title**: The title (on top) of the plot.
- **space**: The space between the chord segments of the plot.
- **gene.order**: A character vector defining the order of the displayed gene labels.
- **gene.size**: The size of the gene labels.
- **gene.space**: The space between the gene labels and the segment of the logFC.
- **nlfc**: Defines the number of logFC columns (default=1).
- **lfc.col**: The fill color for the logFC specified in the following form: c(color for low values, color for the mid point, color for the high values).
- **lfc.min**: Specifies the minimum value of the logFC scale (default = -3).
- **lfc.max**: Specifies the maximum value of the logFC scale (default = 3).
- **ribbon.col**: The background color of the ribbons.
- **border.size**: Defines the size of the ribbon borders.
- **process.label**: The size of the legend entries.
- **limit**: A vector with two cutoff values (default= c(0,0)).

---

**IPPI**

*Multilayer analysis Cava et al. BMC Genomics 2017*

---

**Description**

IPPI function takes as input pathway and network data in order to select genes with central role in that pathway. Please see Cava et al. 2017 BMC Genomics.

**Usage**

`IPPI(pathax, netwa)`

**Arguments**

- **pathax**: pathway matrix Please see example path for format.
- **netwa**: a dataframe Please see example path for format netw.

**Value**

a list with driver genes for each pathway.

**Examples**

```r
# Not run:
DRIVER_SP<-IPPI(pathax=pathway_matrix[,1:3],netwa=netw_IPPI[1:50000,])
# End(Not run)
```
listpathnet

Get human KEGG pathway data and the output of list_path_net define the common genes.

Description

listpathnet creates a list of interacting genes for each human pathway.

Usage

listpathnet(lista_net, pathway_exp)

Arguments

lista_net output of path_net
pathway_exp pathway data as provided by getKEGGdata

Value

a list of genes for each pathway (interacting genes belong to that pathway)

Examples

lista_network<-pathnet(genes.by.pathway=pathway[1:5],data=netw)
list_path<-listpathnet(lista_net=lista_network,pathway=pathway[1:5])

netw network data

Description

network data

Format

A data frame with rows and variables

netw_IPPI network data for IPPI function

Description

network data for IPPI function

Format

A list
**norm**

*TCGA data with normal samples*

**Description**

TCGA data with normal samples

**Format**

A data frame with rows and variables

---

**path**

*pathway data list*

**Description**

pathway data list

**Format**

A list of dataframe

---

**pathnet**

*Get human KEGG pathway data and creates a network data.*

**Description**

pathnet creates a list of network data for each human pathway. The network data will be generated when interacting genes belong to that pathway.

**Usage**

`pathnet(genes.by.pathway, data)`

**Arguments**

- `genes.by.pathway`
  - a list of pathway data as provided by ConvertedIDgenes
- `data`
  - a list of network data as provided by getNETdata

**Value**

a list of network data for each pathway (interacting genes belong to that pathway)

**Examples**

`lista_net<-pathnet(genes.by.pathway=pathway[1:5], data=netw)`
<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>pathway</strong></td>
<td>pathway data</td>
<td>A data frame with rows and variables</td>
</tr>
<tr>
<td><strong>pathway_matrix</strong></td>
<td>network data</td>
<td>A data frame with rows and variables</td>
</tr>
<tr>
<td><strong>path KEgg</strong></td>
<td>All pathways data from KEGG</td>
<td>A list of pathways with the involved genes</td>
</tr>
</tbody>
</table>
**plotcrosstalk**  
*Preparation for plotting cross-talk*

**Description**

plot_crosstalk function takes as input pathway data and prepares the data to visualize (e.g. ggplot2, qqgraph, igraph)

**Usage**

`plotcrosstalk(pathway_plot, gs_expre)`

**Arguments**

- `pathway_plot`  
  pathway
- `gs_expre`  
  a gene expression matrix

**Value**

a list with correlation matrix and gene set for each gene

**Examples**

```r
formatplot<-plotcrosstalk(pathway_plot=pathway[1:6], gs_expre=tumo)
```

---

**score_euc_dist**  
*Score Matrix of pairwise pathway using euclidean distance*

**Description**

Score Matrix of pairwise pathway using euclidean distance

**Format**

A data frame with rows and variables
**SelectedSample**  
*Select the class of TCGA data*

**Description**

select two labels from ID barcode

**Usage**

```r
SelectedSample(Dataset, typesample)
```

**Arguments**

- **Dataset**: gene expression matrix
- **typesample**: the labels of the samples (e.g. tumor, normal)

**Value**

a gene expression matrix of the samples with specified label

**Examples**

```r
tumo<-SelectedSample(Dataset=Data_CANCER_normUQ_fil,typesample="tumour")[,2]
```

---

**select_class**  
*Select the class of TCGA data*

**Description**

select best performance

**Usage**

```r
select_class(performance_matrix, cutoff)
```

**Arguments**

- **performance_matrix**: list of AUC value
- **cutoff**: cut-off for AUC value

**Value**

a gene expression matrix with only pairwise pathway with a particular cut-off
StarBioTrek allows you to Download data of samples from StarBioTrek.

Details

The functions you’re likely to need from StarBioTrek is path_star. Otherwise refer to the vignettes to see how to format the documentation.

stdv

For TCGA data get human pathway data and creates a measure of standard deviations among pathways.

Description

stdv creates a matrix with standard deviation for pathways.

Usage

stdv(gslist)

Arguments

gslist  pathway data

Value

a matrix value for each pathway

Examples

list_path_gene<-GE_matrix(DataMatrix=tumo[,1:2],genes.by.pathway=pathway[1:5])
score_stdev<-stdv(gslist=list_path_gene)
**svm_classification**  

SVM classification for each feature

**Description**

svm class creates a list with AUC, Accuracy, Sensitivity, Specificity values

**Usage**

```
svm_classification(TCGA_matrix, tumour, normal, nfs)
```

**Arguments**

- **TCGA_matrix** gene expression matrix where the first two columns represent the interacting pathways.
- **tumour** barcode samples for a class
- **normal** barcode samples for another class
- **nfs** nfs split data into a training and test set
- **Target** label for the classes

**Value**

a list with AUC value for pairwise pathway

**Examples**

```r
## Not run:
nf <- 60
res_class<-svm_classification(TCGA_matrix=score_euc_dist[,1:30,],nfs=nf,
normal=colnames(norm[,1:10]),tumour=colnames(tumo[,1:10]))
## End(Not run)
```

---

tumo  

**TCGA data with tumour samples**

**Description**

TCGA data with tumour samples

**Format**

A data frame with rows and variables
Index

* internal
  - Data_CANCER_normUQ_fil, 4
  - netw, 11
  - netw_IPPI, 11
  - norm, 12
  - path, 12
  - path_KEGG, 13
  - pathway, 13
  - pathway_matrix, 13
  - score_euc_dista, 14
  - tumo, 17

average, 2

circleplot, 3
ConvertedIDgenes, 4

Data_CANCER_normUQ_fil, 4
dsscorecrtlk, 5
eucdistcrtlk, 5

GE_matrix, 8
GE_matrix_mean, 9
GetData, 6
getNETdata, 6
GetPathData, 7
GetPathNet, 8
GOChord, 9

IPPI, 10

listpathnet, 11

netw, 11
netw_IPPI, 11
norm, 12

path, 12
path_KEGG, 13
pathnet, 12

pathway, 13
pathway_matrix, 13
plotcrosstalk, 14

score_euc_dista, 14
select_class, 15
SelectedSample, 15
StarBioTrek, 16
StarBioTrek-package (StarBioTrek), 16
stdv, 16
svm_classification, 17

tumo, 17