Package 'PanomiR'

April 16, 2025

Title Detection of miRNAs that regulate interacting groups of pathways

Version 1.12.0

Description PanomiR is a package to detect miRNAs that target groups of pathways from gene expression data. This package provides functionality for generating pathway activity profiles, determining differentially activated pathways between user-specified conditions, determining clusters of pathways via the PCxN package, and generating miRNAs targeting clusters of pathways. These function can be used separately or sequentially to analyze RNA-Seq data.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.1.2

Suggests testthat (>= 3.0.0), BiocStyle, knitr, rmarkdown

Config/testthat/edition 3

biocViews GeneExpression, GeneSetEnrichment, GeneTarget, miRNA, Pathways

Imports clusterProfiler, dplyr, forcats, GSEABase, igraph, limma, metap, org.Hs.eg.db, parallel, preprocessCore, RColorBrewer, rlang, tibble, withr, utils

Depends R (>= 4.2.0)

URL https://github.com/pouryany/PanomiR

BugReports https://github.com/pouryany/PanomiR/issues

VignetteBuilder knitr

git_url https://git.bioconductor.org/packages/PanomiR

git_branch RELEASE_3_21

git_last_commit 8fb70fa

git_last_commit_date 2025-04-15

Repository Bioconductor 3.21

Date/Publication 2025-04-15

Author Pourya Naderi [aut, cre],
Yue Yang (Alan) Teo [aut],
Ilya Sytchev [aut],
Winston Hide [aut]

Maintainer Pourya Naderi <pouryany@gmail.com>

Contents

aggInvCoverFn . . . 3
aggInvFn . . . 3
aggLogCoverFn . . . 4
aggLogFn . . . 4
alignToUniverse . . . 5
clusterPlot . . . 5
differentialPathwayAnalysis . . . 6
enrichAllPairs . . . 7
getDesignMatrix . . . 8
getDiffExpTable . . . 9
getResidual . . . 9
gscExample . . . 10
jackKnifeBase . . . 10
linColumnFinder . . . 11
mappingPathwaysClusters . . . 12
methodProbBase . . . 13
miniTestsPanomiR . . . 14
miRNAPathwayEnrichment . . . 15
msigdb_c2 . . . 16
pathwayGeneTab . . . 16
pathwaySummary . . . 17
path_gene_table . . . 18
pCutCoverFn . . . 19
pCutFn . . . 19
pcxnToNet . . . 20
prioritizeMicroRNA . . . 21
reportEnrichment . . . 22
samplingDataBase . . . 23
sumlogCoverFn . . . 24
sumlogFn . . . 24
sumzCoverFn . . . 25
sumzFn . . . 25
tableFromGSC . . . 26
targetScan_03 . . . 26

Index **28**

aggInvCoverFn — *Internal function for modification of prioritization.*

Description

Internal function for modification of prioritization.

Usage

```
aggInvCoverFn(selector, coverName)
```

Arguments

selector	a prioritzation table
coverName	a new column name

Value

an updated scoring of miRNAs in a cluster of pathways

aggInvFn — *The function calculate targeting score of miRNA w.r.t to a cluster of pathways via inverse normal method*

Description

The function calculate targeting score of miRNA w.r.t to a cluster of pathways via inverse normal method

Usage

```
aggInvFn(enriches, pathways, isSelector = TRUE, thresh = NULL)
```

Arguments

enriches	a table of miRNA pathway enrichments. Universe
pathways	queried pathways. e.g. cluster pathways
isSelector	internal argument
thresh	internal argument

Value

a scoring of miRNAs in a cluster of pathways

aggLogCoverFn *Internal function for modification of prioritization.*

Description

Internal function for modification of prioritization.

Usage

```
aggLogCoverFn(selector, coverName)
```

Arguments

selector	a prioritzation table
coverName	a new column name

Value

an updated scoring of miRNAs in a cluster of pathways

aggLogFn *The function calculate targeting score of miRNA w.r.t to a cluster of pathways via log aggregation method.*

Description

The function calculate targeting score of miRNA w.r.t to a cluster of pathways via log aggregation method.

Usage

```
aggLogFn(enriches, pathways, isSelector, thresh = 0)
```

Arguments

enriches	a table of miRNA pathway enrichments. Universe
pathways	queried pathways. e.g. cluster pathways
isSelector	internal argument
thresh	internal argument

Value

a scoring of miRNAs in a cluster of pathways

`alignToUniverse` *function to align a list of sets and a reference universe*

Description

function to align a list of sets and a reference universe

Usage

```
alignToUniverse(pathwaySets, universe)
```

Arguments

`pathwaySets`	a list of sets
`universe`	all set elements must be a subset of universe

Value

a list of sets, aligned to universe

`clusterPlot` *Plots clusters of pathways with associated directionality.*

Description

Plots clusters of pathways with associated directionality.

Usage

```
clusterPlot(
  subNet,
  subplot = FALSE,
  topClusters = 2,
  prefix = "",
  outDir = ".",
  plotSave = TRUE
)
```

Arguments

`subNet`	pathways network (edge list of pathways)
`subplot`	if TRUE, store individual clusters plots and connected plots in Figures directory of plots
`topClusters`	plot figures for top x clusters
`prefix`	add prefix to plots
`outDir`	output directory
`plotSave`	saves the plot if set true. Otherwise display

Value

a set of plots for DE-PCXN and subclusters

Examples

```
data(miniTestsPanomiR)
clusterPlot(miniTestsPanomiR$miniPathClusts$DE_PCXN, plotSave = FALSE)
```

`differentialPathwayAnalysis` *Differential Expression Analysis For Pathways*

Description

Performs differential expression analysis for pathways using LIMMA package with gene counts

Usage

```
differentialPathwayAnalysis(
  geneCounts,
  pathways,
  covariates,
  condition,
  adjustCovars = NULL,
  covariateCorrection = FALSE,
  quantileNorm = FALSE,
  outDir = ".",
  saveOutName = NULL,
  id = "ENSEMBL",
  deGenes = NULL,
  minPathSize = 10,
  method = "x2",
  trim = 0.025,
  geneCountsLog = TRUE,
  contrastConds = NA
)
```

Arguments

`geneCounts`	Gene counts, rows refer to genes and columns to samples.
`pathways`	Pathways table, containing pathway names and genes with id specified.
`covariates`	Covariates/metadata file; rows matches the columns of geneCounts.
`condition`	Condition to be examined (tumor vs normal etc); must exist in covariates column.
`adjustCovars`	Adjustment covariates like batch; if NULL, no adjustments performed.

covariateCorrection
: If TRUE, performs covariates detection and correction; requires **adjustCovars**; (limma).

quantileNorm
: If TRUE, performs quantile normalization on pathway summary statistics; from *preprocess* package.

outDir
: Output directory.

saveOutName
: If not NULL, saves output as RDS using save name, if NULL, does not save output.

id
: ID matching genes to pathways; rownames of geneCounts.

deGenes
: If not NULL, add t-scores to pathways summary statistics; filter by genes t-scores.

minPathSize
: Minimum pathway size.

method
: Define method to use for pathway summary statistics; specifications in documentations.

trim
: Filter pathways with mean less than trim threshold in pathway summary statistics.

geneCountsLog
: If TRUE, log(geneCounts).

contrastConds
: Provide a contrast expression to be used in Limma comparison. This is necessary if you have more than two levels in the condition covariate.

Value

List containing differentially expressed pathways as DEP and pathway summary statistics as pathwaySummaryStats.

Examples

```
data("path_gene_table")
data("miniTestsPanomiR")

differentialPathwayAnalysis(geneCounts = miniTestsPanomiR$mini_LIHC_Exp,
pathways =  path_gene_table,
covariates = miniTestsPanomiR$mini_LIHC_Cov,
condition = 'shortLetterCode')
```

---

enrichAllPairs *Pairwise enrichment analysis between two given lists of sets*

Description

Pairwise enrichment analysis between two given lists of sets

Usage

```
enrichAllPairs(mirSets, pathwaySets, pathsRef, numCores)
```

Arguments

mirSets	a list of targets of miRNAs
pathwaySets	a list of pathways
pathsRef	universe of genes.
numCores	number of cores to calculate the results.

Value

enrichment analysis results

---

getDesignMatrix *Obtain Design Matrix*

Description

Modified from covariates pipeline of Menachem Former. Imported from https://github.com/th1vairam/CovariateAnalysis

Usage

```
getDesignMatrix(covariatesDataFrame, intercept = TRUE, reLevels = list())
```

Arguments

covariatesDataFrame	Dataframe of covariates.
intercept	intercept in the linear model.
reLevels	TBA.

Value

List containing a design matrix.

Examples

```
data(iris)
getDesignMatrix(iris)
```

getDiffExpTable *function to get a DE table*

Description

function to get a DE table

Usage

```
getDiffExpTable(expMat, designMat, contrastsName)
```

Arguments

expMat	an expression matrix
designMat	a design Matrix
contrastsName	the contrast to perform

Value

a table of differential expression

getResidual *function to get residuals with respect to a set of covariates*

Description

function to get residuals with respect to a set of covariates

Usage

```
getResidual(covariates, adjustCovars, pathSumStats)
```

Arguments

covariates	a covariate dataframe.
adjustCovars	covariates to adjust for
pathSumStats	an expression matrix

Value

a matrix of adjusted expression

gscExample — *Example genesets from MSigDB*

Description

Example genesets from MSigDB

Usage

```
data(gscExample)
```

Format

A GeneSet Collection object containing two genesets.

Source

http://www.gsea-msigdb.org/gsea/index.jsp

Examples

```
data(gscExample)
```

jackKnifeBase — *Outputs a table with col x (miRNA), probability of observing k (depending on methodology) against a random distribution with jack-knifing of the pathway cluster (removing a pathway at a time)*

Description

Outputs a table with col x (miRNA), probability of observing k (depending on methodology) against a random distribution with jack-knifing of the pathway cluster (removing a pathway at a time)

Usage

```
jackKnifeBase(
  selector,
  pathways,
  enrichNull,
  fn,
  jackKnifeData,
  m,
  numCores = 1
)
```

Arguments

selector	Table with x(miRNA) in pathway cluster and observed k (depending on methodology).
pathways	Pathways in pathway cluster.
enrichNull	Enrichment dataset with x (miRNA), y (pathway) and pval (probability of observing x in pathway cluster).
fn	Methodology function.
jackKnifeData	Random distribution data with jack-knifing (i.e. one less pathway)
m	method name
numCores	number of cores

Value

Outputs a new selector table with col x, pval_jk

---

linColumnFinder *Function imported from https://github.com/th1vairam/CovariateAnalysis Modified from http://stackoverflow.com/questions/13088770/ Function to find linearly dependednt columns of a matrix*

Description

Function imported from https://github.com/th1vairam/CovariateAnalysis Modified from http://stackoverflow.com/questions/1
Function to find linearly dependednt columns of a matrix

Usage

```
linColumnFinder(mat)
```

Arguments

mat	an input design matrix.

Value

a list of independent columns

Examples

```
data("iris")
designMat <- getDesignMatrix(iris)
linColumnFinder(designMat$design)
```

`mappingPathwaysClusters` *Outputs a table with pathways and their respective clusters*

Description

Outputs a table with pathways and their respective clusters

Usage

```
mappingPathwaysClusters(
  pcxn,
  dePathways,
  clusteringFunction = NULL,
  edgeFDR = 0.05,
  correlationCutOff = 0.316,
  pathwayFDR = 0.05,
  topPathways = 200,
  plotOut = TRUE,
  subplot = TRUE,
  topClusters = 2,
  prefix = "",
  outDir = ".",
  saveNameCSV = NULL,
  weighted = FALSE
)
```

Arguments

`pcxn`	pathways network (edge list of pathways)
`dePathways`	differential expressed pathways, obtained from *DifferentialPathwayAnalysis*
`clusteringFunction`	clustering algorithm
`edgeFDR`	FDR threshold for pathway-pathway adjusted p-values; filter edges with adjusted p-values less than given threshold
`correlationCutOff`	cut-off threshold for pathway-pathway correlation; filter pathways with correlation less than given threshold
`pathwayFDR`	FDR threshold for DE pathways adjusted p-values; filter pathways with adjusted p-values less than given threshold
`topPathways`	use only top x paths; if NULL, use all paths
`plotOut`	if TRUE, store graph plot in Figures directory of plots
`subplot`	if TRUE, store individual clusters plots and connected plots in Figures directory of plots

topClusters	plot figures for top x clusters
prefix	add prefix to plots
outDir	output directory
saveNameCSV	if not NULL, saves output as csv using save name
weighted	True if you wish to include correlation weights in clustering

Value

a list where the first item is a table with each row containing a pathway and its respective cluster. The second item is an igraph object.

Examples

```
data("miniTestsPanomiR")

mappingPathwaysClusters(pcxn = miniTestsPanomiR$miniPCXN,
                        dePathways = miniTestsPanomiR$miniDEP,
                        topPathways = 200,
                        outDir=".",
                        plot = FALSE,
                        subplot = FALSE,
                        prefix='',
                        clusteringFunction = "cluster_louvain",
                        correlationCutOff = 0.1)
```

---

methodProbBase *Outputs a table with col x, miRNA, probability of observing k against a random distribution of the cover of methodology*

Description

Outputs a table with col x, miRNA, probability of observing k against a random distribution of the cover of methodology

Usage

```
methodProbBase(samplingData, selector, m, nPaths = 100, coverFn = NULL)
```

Arguments

samplingData	Random distribution data.
selector	Table with x(miRNA) in pathway cluster and observed k (depending on methodology).
m	Method name.
nPaths	Number of pathways used to generate the samplingData at each iteration. Default is set at 100.
coverFn	Cover of methodology function.

Value

Outputs a new selector table with col x, pval and cover.

`miniTestsPanomiR` *Readouts and datasets for minimal reproducible examples of the PanomiR.*

Description

The item miniEnrich is a reduced representation of the TargetScan For full table use miRNAPathwayEnrichment function in the package along with msigdb_c2 and targetScan_03 datasets.

Usage

```
data(miniTestsPanomiR)
```

Format

A list of 5:

mini_LIHC_Exp a reduced expression dataset from TCGA LIHC data

mini_LIHC_Cov a reduced covariates dataset from TCGA LIHC data

miniEnrich a reduced table of miRNA-pathway enrichment, TargetScan.

miniDEP Differentially activated pathways from reduced TCGA LIHC

miniPCXN reduced representation of PCXN network

miniPathClusts miniDEP mapped to miniPCXN

Details

These datasets include reduced representation of TCGA LIHC data for reproducing the pipeline. doi: 10.1016/j.cell.2017.05.046

A reduced representation of PCxN is provided. For full dataset and method please refer to pcxn.org or https://doi.org/10.1371/journal.pcbi.1006042

Examples

```
data(miniTestsPanomiR)
```

`miRNAPathwayEnrichment` *Enrichment Probability Of miRNAs*

Description

Outputs enrichment probability of miRNAs based on pathway clusters.

Usage

```
miRNAPathwayEnrichment(
  mirSets,
  pathwaySets,
  geneSelection = NULL,
  mirSelection = NULL,
  fromID = "ENSEMBL",
  toID = "ENTREZID",
  minPathSize = 9,
  numCores = 1,
  outDir = ".",
  saveOutName = NULL
)
```

Arguments

`mirSets`	Table of miRNAs and a list of their interactions with genes in ENTREZ ID.
`pathwaySets`	Table of pathways and a list of their interactions with genes in ENTREZ ID.
`geneSelection`	Table of genes with dtype; if not NULL, select only genes from a given table.
`mirSelection`	Table of miRNA names; if not NULL, select only miRNAs from given table.
`fromID`	ID of genes in geneSelection.
`toID`	ID of genes used in pcxn and pathways set.
`minPathSize`	Filter out pathways with sets less than given value.
`numCores`	Number of CPU cores to use, must be at least one.
`outDir`	Output directory.
`saveOutName`	If not NULL, saves output as RDS using save name.

Value

Table of enrichment, each row contains mirna-pathway and its enrichment p-values.

Examples

```
data(msigdb_c2)
data(targetScan_03)
miRNAPathwayEnrichment(targetScan_03[1:20],msigdb_c2[1:20])
```

msigdb_c2 — *Canonical pathways from Molecular Signatures Database, MsigDb V6.2*

Description

Canonical pathways from Molecular Signatures Database, MsigDb V6.2

Usage

```
data(msigdb_c2)
```

Format

A list of 1143 pathways

Source

http://www.gsea-msigdb.org/gsea/index.jsp

Examples

```
data(msigdb_c2)
```

pathwayGeneTab — *Pathway-Gene Associations*

Description

Generates a table of pathways and genes associations.

Usage

```
pathwayGeneTab(
  pathAdress = NA,
  pathwayList = NA,
  fromType = "ENTREZID",
  toType = "ENSEMBL",
  outDir = NA
)
```

Arguments

pathAdress	Address to an RDS file containing list of pathways where each element is a list of genes similar to GMT format.
pathwayList	If you wish to use a list of pathways instead of a file use this argument instead. The list must contain no NA values.
fromType	gene annotation type used in your input data.
toType	gene annotation type to be produced in the output.
outDir	Address to save an RDS for a table of pathway-gene association

Value

pathExpTab Table of pathway-gene association.

Examples

```
pathway1 <- c("125", "3099", "126")
pathway2 <- c("5232", "5230", "5162")
pathList <- list("Path1" = pathway1, "Path2" = pathway2)
res <- pathwayGeneTab(pathwayList = pathList)

data(msigdb_c2)
pathwayGeneTab(pathwayList = msigdb_c2[1:2])
```

---

pathwaySummary *Pathway Summary Statistics*

Description

Generates a table of pathway activity profiles per sample

Usage

```
pathwaySummary(
  exprsMat,
  pathwayRef,
  id = "ENSEMBL",
  zNormalize = FALSE,
  method = FALSE,
  deGenes = NULL,
  trim = 0,
  tScores = NULL
)
```

Arguments

exprsMat	Gene expression matrix with row names as genes and samples as columns.
pathwayRef	Table of pathway-gene associations. Created from pathwayGeneTab function.
id	Gene annotation type in the row name of gene expression data.
zNormalize	Normalization of pathway summary score.
method	Choice of how to summarize gene ranks into pathway statistics.
deGenes	List of differentially expressed genes along with t-scores. Only necessary if working on Top 50% summary method.
trim	Percentage of top and bottom ranked genes to be excluded from pathway summary statistics.
tScores	Argument for-top-50-percent-genes method.

Value

pathExp Table of pathway activity profiles per sample.

Examples

```
pathTab <- tibble::tribble(
 ~Pathway, ~ENTREZID,  ~ENSEMBL,
 "Path1",   "125",      "ENSG00000196616",
 "Path1",   "3099",     "ENSG00000159399",
 "Path2",   "5230",     "ENSG00000102144",
 "Path2",   "5162",     "ENSG00000168291"
 )
exprsMat <- matrix(2 * (seq_len(12)), 4, 3)
rownames(exprsMat) <- pathTab$ENSEMBL
colnames(exprsMat) <- LETTERS[seq_len(3)]
pathwaySummary(exprsMat, pathTab, method = "x2")
```

---

path_gene_table *A table of gene-pathway association. based on the pathways of MSigDB.*

Description

A table of gene-pathway association. based on the pathways of MSigDB.

Usage

```
data(path_gene_table)
```

Format

A matrix with 3 columns and 76926 rows:

Pathway An MSigDB annotated pathway

ENTREZID The ENTREZID of a gene belonging to the pathway

ENSEMBL The ENSEMBL of a gene belonging to the pathway

Examples

```
data(path_gene_table)
```

---

pCutCoverFn *Internal function for modification of prioritization.*

Description

Internal function for modification of prioritization.

Usage

```
pCutCoverFn(selector, coverName)
```

Arguments

selector	a prioritzation table
coverName	a new column name

Value

an updated scoring of miRNAs in a cluster of pathways

---

pCutFn *Score miRNAs In a Cluster Of Pathways*

Description

The function to count the number of enriched pathways for each miRNA.

Usage

```
pCutFn(enriches, pathways, isSelector, thresh = 0.05)
```

Arguments

enriches	Table of miRNA pathway enrichments.
pathways	Queried pathways, e.g. cluster pathways.
isSelector	Internal argument.
thresh	Threshold from p-value cut-off.

Value

P-value based scoring of miRNAs in a cluster of pathways.

---

pcxnToNet *Creates a network out of pcxn table*

Description

Creates a network out of pcxn table

Usage

```
pcxnToNet(pcxn, edgeFDR, correlationCutOff, weighted)
```

Arguments

pcxn	pathways network edge list of pathways
edgeFDR	FDR threshold for pathway-pathway adjusted p-values; filter edges with adjusted p-values less than given threshold
correlationCutOff	cut-off threshold for pathway-pathway correlation; filter pathways with correlation less than given threshold
weighted	True if you wish to include correlation weights in clustering

Value

enrichment analysis results

prioritizeMicroRNA *Prioritize miRNA*

Description

Outputs a table of miRNA ordered with respective p-values derived from method for prioritization

Usage

```
prioritizeMicroRNA(
  enriches0,
  pathClust,
  method = "AggInv",
  methodThresh = NULL,
  enrichmentFDR = 0.25,
  topClust = 2,
  sampRate = 1000,
  outDir = ".",
  dataDir = ".",
  saveSampling = TRUE,
  runJackKnife = TRUE,
  saveJackKnife = FALSE,
  numCores = 1,
  saveCSV = TRUE,
  prefix = "",
  autoSeed = TRUE
)
```

Arguments

enriches0	miRNA-pathway enrichment dataset obtained from miRNAPathwayEnrichment.
pathClust	Pathway clusters, obtained from MappingPathwaysClusters.
method	Vector of methods pCut, AggInv, AggLog, sumz, sumlog.
methodThresh	Vector of methods threshold for each method in method, if NULL use default thresh values in method.
enrichmentFDR	FDR cut-off calculating miRNA-pathway hits in the input cluster based on significant enrichment readouts.
topClust	Top x clusters to perform miRNA prioritization on.
sampRate	Sampling rate for CLT.
outDir	Output directory.
dataDir	Data directory.
saveSampling	If TRUE, saves sampling data as RDS for each cluster in topClust in dataDir.
runJackKnife	If TRUE, jacknifing will be performed.

saveJackKnife	If TRUE, saves jack-knifed sampling data as RDS for each cluster in topClust in dataDir.
numCores	Number of CPU cores to use, must be at least one.
saveCSV	If TRUE, saves CSV file for each cluster in topClust in outDir.
prefix	Prefix for all saved data.
autoSeed	random permutations are generated based on predetermined seeds. TRUE will give identical results in different runs.

Value

Table of miRNA and p-values, each row contains a miRNA and its associated p-values from the methods.

Examples

```
data("miniTestsPanomiR")

prioritizeMicroRNA(enriches0 = miniTestsPanomiR$miniEnrich,
   pathClust = miniTestsPanomiR$miniPathClusts$Clustering,
   topClust = 1,
   sampRate = 50,
   method = c("aggInv"),
   saveSampling = FALSE,
   runJackKnife = FALSE,
   numCores = 1,
   saveCSV = FALSE)
```

---

reportEnrichment *Publication-ready miRNA-Pathway Enrichment table*

Description

This function summarizes the outputs

Usage

```
reportEnrichment(enrichmentTable)
```

Arguments

enrichmentTable	Outputs from [miRNAPathwayEnrichment()] function

Value

A summarized miRNA-Pathway enrichment table

Examples

```
data(msigdb_c2)
data(targetScan_03)
eTab <- miRNAPathwayEnrichment(targetScan_03[1:20],msigdb_c2[1:20])

repTab <- reportEnrichment(eTab)
```

`samplingDataBase` *Outputs a table of sampling data(rows are miRNA and cols are samples)*

Description

Outputs a table of sampling data(rows are miRNA and cols are samples)

Usage

```
samplingDataBase(
  enrichNull,
  selector,
  sampRate,
  fn,
  nPaths,
  samplingDataFile,
  jackKnife = FALSE,
  saveSampling,
  numCores = 1,
  autoSeed = TRUE
)
```

Arguments

`enrichNull`	Enrichment dataset with x (miRNA), y (pathway) and pval (probability of observing x in pathway cluster).
`selector`	Table with x(miRNA) in pathway cluster.
`sampRate`	Sampling rate.
`fn`	Methodology function.
`nPaths`	Number of pathways in pathway cluster.
`samplingDataFile`	If file exists, load. Else, perform random sampling
`jackKnife`	If TRUE, conduct sampling with one less pathway, used for jack knifing
`saveSampling`	If TRUE, data is saved.
`numCores`	number of cores used
`autoSeed`	random permutations are generated based on predetermined seeds. TRUE will give identical results in different runs.

Value

Outputs of sampling data.

`sumlogCoverFn` *Internal function for modification of prioritization.*

Description

Internal function for modification of prioritization.

Usage

```
sumlogCoverFn(selector, coverName)
```

Arguments

`selector`	a prioritzation table
`coverName`	a new column name

Value

an updated scoring of miRNAs in a cluster of pathways

`sumlogFn` *The function calculate targeting score of miRNA w.r.t to a cluster of pathways via sumlog aggregation method.*

Description

The function calculate targeting score of miRNA w.r.t to a cluster of pathways via sumlog aggregation method.

Usage

```
sumlogFn(enriches, pathways, isSelector, thresh = NULL)
```

Arguments

`enriches`	a table of miRNA pathway enrichments. Universe
`pathways`	queried pathways. e.g. cluster pathways
`isSelector`	internal argument
`thresh`	internal argument

Value

a scoring of miRNAs in a cluster of pathways

sumzCoverFn — *Internal function for modification of prioritization.*

Description

Internal function for modification of prioritization.

Usage

```
sumzCoverFn(selector, coverName)
```

Arguments

selector	a prioritzation table
coverName	a new column name

Value

an updated scoring of miRNAs in a cluster of pathways

sumzFn — *The function calculate targeting score of miRNA w.r.t to a cluster of pathways via sumz aggregation method.*

Description

The function calculate targeting score of miRNA w.r.t to a cluster of pathways via sumz aggregation method.

Usage

```
sumzFn(enriches, pathways, isSelector, thresh = NULL)
```

Arguments

enriches	a table of miRNA pathway enrichments. Universe
pathways	queried pathways. e.g. cluster pathways
isSelector	internal argument
thresh	internal argument

Value

a scoring of miRNAs in a cluster of pathways

tableFromGSC *Pathway-Gene Associations from GeneSet collections*

Description

This function enables to utilize MSigDB packages and GSEABase objects to incorporate customized genesets into PanomiR.

Usage

```
tableFromGSC(gsCollection, fromType = "ENTREZID", toType = "ENSEMBL")
```

Arguments

gsCollection	An GSEABase gene set collection object
fromType	gene annotation type used in your input data
toType	gene annotation type to be produced in the output

Value

A table of pathway-gene associations

Examples

```
data(gscExample)
tableFromGSC(gscExample)
```

targetScan_03 *A processed list of miRNA target gene sets from the TargetScan dataset. Each list item is a list of genes targeted by the respective miRNA family*

Description

The interactions are filtered to only human interactions.

Usage

```
data(targetScan_03)
```

Format

A list of 439 items

Details

The interactions are filtered to have a Cumulative weighted context++ score of < -0.3

Source

http://www.targetscan.org/vert_72/

Examples

```
data(targetScan_03)
```

Index

∗ **datasets**
  gscExample, 10
  miniTestsPanomiR, 14
  msigdb_c2, 16
  path_gene_table, 18
  targetScan_03, 26
∗ **internal**
  aggInvCoverFn, 3
  aggInvFn, 3
  aggLogCoverFn, 4
  aggLogFn, 4
  pCutCoverFn, 19
  pCutFn, 19
  sumlogCoverFn, 24
  sumlogFn, 24
  sumzCoverFn, 25
  sumzFn, 25

aggInvCoverFn, 3
aggInvFn, 3
aggLogCoverFn, 4
aggLogFn, 4
alignToUniverse, 5

clusterPlot, 5

differentialPathwayAnalysis, 6

enrichAllPairs, 7

getDesignMatrix, 8
getDiffExpTable, 9
getResidual, 9
gscExample, 10

jackKnifeBase, 10

linColumnFinder, 11

mappingPathwaysClusters, 12
methodProbBase, 13
miniTestsPanomiR, 14
miRNAPathwayEnrichment, 15
msigdb_c2, 16

path_gene_table, 18
pathwayGeneTab, 16, *18*
pathwaySummary, 17
pCutCoverFn, 19
pCutFn, 19
pcxnToNet, 20
prioritizeMicroRNA, 21

reportEnrichment, 22

samplingDataBase, 23
sumlogCoverFn, 24
sumlogFn, 24
sumzCoverFn, 25
sumzFn, 25

tableFromGSC, 26
targetScan_03, 26