Package 'methyl.O'

March 30, 2021

Title Annotate, Score and Visualize Differentially Methylated Regions

Version 1.0.0

Depends R (>= 3.5.0)

biocViews

Imports EnsDb.Hsapiens.v75,
EnsDb.Hsapiens.v86,
TxDb.Hsapiens.UCSC.hg19.knownGene (>= 3.2.2),
GenomicRanges (>= 1.38.0),
biomaRt (>= 2.42.1),
S4Vectors (>= 0.24.4),
Gviz (>= 1.30.0),
ensembldb (>= 2.10.2),
DT (>= 0.17),
shiny (>= 1.6.0),
shinycssloaders (>= 1.0.0),
shinythemes (>= 1.2.0),
shinyWidgets (>= 0.5.7),
shinyLP (>= 1.1.2),
shinyBS (>= 0.61),
shinyalert (>= 2.0.0),
colourpicker (>= 1.1.0),
ggplot2 (>= 3.3.3),
wesanderson (>= 0.3.6),
enrichR

Manteiner Gianluca Mattei <gianluca.mattei@unifi.it>

Author Gianluca Mattei

Description Methods annotateDMRs() to retrieve annotations for DMRs and scoreAnnotatedDMRs() to assign a score to the annotated DMRs. Method annotateEhancers() is used to associate enhancers to genes. Method plotDMRs() used to visualize the annotated regions.

License LGPL (>= 2.1)

URL https://github.com/GianlucaMattei/methyl.O

BugReports https://github.com/GianlucaMattei/methyl.O/issues

Repository CRAN

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

Suggests knitr, rmarkdown, BiocStyle

VignetteBuilder knitr

R topics documented:

`annotatedDMRs2Enrichr` *Query different databases to find enriched proceses.*

Description

Query gene symbols from resulting annotations list in order to find enriched proceses.

Usage

```
annotatedDMRs2Enrichr(
  annotatedDMRs,
  active.features = c("promoters", "heads"),
  stat.filter = "P.value",
  stat.thr = 0.01,
  db = NULL
)
```

Arguments

`annotatedDMRs`	anotated DMRs list resulting from annotateDMRs() or scoreAnnotatedDMRs()
`active.features`	annotation level from the gene symbols are taken. Default = c('promoters', 'heads')

stat.filter	character indicating which type of statistics use for filtering results. Accepted values: 'P.value', 'Adjusted.P.value' or 'Overlap'. Default 'P.value'.
stat.thr	numeric value indicating the threshold to use for selcted statistical test. Default 0.01.
db	vector of characters indicating DBs to query. Default c("ClinVar_2019", "OMIM_Disease", "Elsevier_Pathway_Collection", "MSigDB_Hallmark_2020", "MSigDB_Oncogenic_Signatures", "GO_Biological_Process_2018", "Human_Phenotype_Ontology", "KEGG_2016", "NCI-Nature_2016", "Panther_2016", "Reactome_2016", "WikiPathways_2019_Human")

Value

data.frame with enriched processes

`annotatedDMRs2Exprs` *Compute correlation between expression and methylation levels.*

Description

Compute correlation between expression and methylation levels.

Usage

```
annotatedDMRs2Exprs(
  annotatedDMRs,
  expressionProfile,
  active.features = c("promoters", "heads"),
  col.genes = 0,
  col.stat = 6,
  stat.thr = 0.05,
  col.logFC = 2,
  logfc.thr = 0,
  convert.genes = FALSE,
  convert.from,
  beta.thr = 0.3,
  overlap.param.thr = 100,
  param.type = "overlap.length",
  line.col = "lightgrey",
  lmfit.col1 = "red",
  lmfit.col2 = "green",
  pal = "RdGy",
  plot.type = "splitted",
  show.text = FALSE,
  cor.type = "pearson",
  filter.by.genes = NULL,
  return.table = FALSE
)
```

Arguments

annotatedDMRs	anotated DMRs list resultingfrom annotateDMRs() or scoreAnnotatedDMRs()
expressionProfile	data.frame, expression profile.
active.features	character vectors, containing features to correlate. Must be from names of resulting list from annotateDMRs. Additional feature names can be first exons (exons1) or first intron (intron1). To use more than one feature use c(). Default = c("promoters", "heads")
col.genes	numeric, the column of expressionProfile data.frame with gene Ids. If NULL geneIDs will be taken from rownames() of expressionProfile. Default = 0.
col.stat	numeric, the column of expressionProfile data.frame with the statistics to use. Default = 6.
stat.thr	numeric, threshold for statistical significance. Default = 0.05
col.logFC	numeric, the column of expressionProfile data.frame with log. fold change. Default = 2
logfc.thr	numeric, threshold value for log. fold change. Default = 0.
convert.genes	boolean, used to indicate if gene ids have to be translated in official gene symbols. Default = FALSE
convert.from	character, used annotation for gene in expressionProfile to be converted to symbols gene IDs. Accepted: c("ENTREZID" ,"EXONID" ,"GENEBIOTYPE" ,"GENEID" ,"GENENAME" ,"PROTDOMID" ,"PROTEINDOMAINID" ,"PROTEINDOMAINSOURCE" ,"PROTEINID" ,""SEQSTRAND" ,"SYMBOL" ,"TXBIOTYPE" ,"TXID" ,"TXNAME" ,"UNIPROTID")
beta.thr	numeric, beta difference threshold value. Default = 0.3.
overlap.param.thr	nuemric, threshold value for selected parameter to filter methylations overlapping the selected features. Default = 100
param.type	character, threshold parameter to filter methylations overlapping the selected features. Accetped c("dmr.length", "overlap.length", "overlap.percentage"). Default = "overlap.length".
line.col	character, color of lines at x=0, y=0. Default = "lightgray"
lmfit.col1	character, color of linear model line 1 or for simple plot. Default = "red"
lmfit.col2	character, color of linear model line 2. Default = "green"
pal	character, color palette. hcl.pals() to show available. Default = "RdGy"
plot.type	character, compute or not different linear models for upregulated and downregulated genes. Accepted: "simple" or "splitted". Default = "splitted".
show.text	logical, indicating if print gene names in the final plot. Accepted values: TRUE or FALSE. Default = FALSE.
cor.type	character, correlation method. Available "pearson", "kendall" or "spearman" correlation, Default = "pearson"
filter.by.genes	character vectors, gene symbols used for filtering output
return.table	logical, TRUE return a data.frame instead a plot. Default = FALSE

Value

plot of correlation or (if return.table = TRUE) a data.frame of genes expression associated to beta methylation values.

`annotatedEnh2Exprs` *Compute expression methylation correlation*

Description

Compute and plot correlation between expression and methylation

Usage

```
annotatedEnh2Exprs(
  annotatedEnhancers,
  expressionProfile,
  hg = "hg19",
  enhancer.db = "FANTOM5",
  col.genes = 0,
  col.stat = 6,
  stat.thr = 0.05,
  col.logFC = 2,
  logfc.thr = 0.5,
  convert.genes = FALSE,
  convert.from,
  beta.thr = 0.3,
  overlap.param.thr = 40,
  param.type = "overlap.percentage",
  line.col = "lightgrey",
  lmfit.col1 = "red",
  lmfit.col2 = "green",
  pal = "RdGy",
  plot.type = "splitted",
  show.text = FALSE,
  cor.type = "pearson",
  return.table = FALSE
)
```

Arguments

`annotatedEnhancers`	data.frame. It corresponds to resulting list from annotateEnhancers()
`expressionProfile`	expression profile data.frame
`hg`	character, "hg19", "hg38". Version of the enhancer database. Default = "hg19"
`enhancer.db`	character, which database to use between 'FANTOM5' or '4DGenome'. Default = "FANTOM5"
`col.genes`	numeric, the column of expressionProfile data.frame with gene Ids. If NULL geneIDs will be taken from rownames() of expressionProfile. Default = 0.

col.stat	numeric, the column of expressionProfile data.frame with the statistics to use. Default = 6.
stat.thr	threshold for statistical significance. Default = 0.05
col.logFC	numeric, the column of expressionProfile data.frame with log. fold change. Default = 2
logfc.thr	numeric, threshold value for log. fold change. Default = 0.
convert.genes	logical, used to indicate if gene ids have to be translated in official gene symbols. Default = FALSE
convert.from	character, used annotation for gene in expressionProfile to be converted to symbols gene IDs. Accepted: c("ENTREZID" ,"EXONID" ,"GENEBIOTYPE" ,"GENEID" ,"GENENAME" ,"PROTDOMID" ,"PROTEINDOMAINID" ,"PROTEINDOMAINSOURCE" ,"PROTEINID" ,""SEQSTRAND" ,"SYMBOL" ,"TXBIOTYPE" ,"TXID" ,"TXNAME" ,"UNIPROTID")
beta.thr	numeric, beta difference threshold value. Default = 0.3.
overlap.param.thr	numeric, threshold value for selected parameter to filter methylations overlapping the selected features. Default = 100
param.type	character, threshold parameter to filter methylations overlapping the selected features. Accetped c("dmr.length", "overlap.length", "overlap.percentage"). Default = "overlap.length".
line.col	character, color of lines at x=0, y=0. Defalt = "lightgray"
lmfit.col1	character, color of linear model line 1 or for simple plot. Defalt = "red"
lmfit.col2	character, color of linear model line 2. Defalt = "green"
pal	character, color palette. hcl.pals() to show available. Default = "RdGy"
plot.type	character, compute or not different linear models for upregulated and downregulated genes. Accepted: "simple" or "splitted". Default = "splitted".
show.text	logical, indicating if print gene names in the final plot. Accepted values: TRUE or FALSE. Default = FALSE.
cor.type	character, correlation method. Available "pearson", "kendall" or "spearman" correlation, Default = "pearson"
return.table	logical, TRUE return a data.frame instead a plot. Default = FALSE

Value

plot of correlation or (if return.table = TRUE) a data.frame of genes expression associated to beta methylation values of enhancers.

---

annotateDMRs *Annotates the the differentially methylated regions.*

Description

Maps DMRs on genes returning a list for each features.

Usage

```
annotateDMRs(
  DMRsRanges,
  prom.length = 1500,
  head.length = 1500,
  longest.trx = TRUE,
  annotation = "ensembl",
  hg = "hg19",
  annotation.fast = TRUE,
  thr.beta = 0.3,
  thr.cgis = 0.4,
  col.betadiff = 4,
  col.beta1 = NULL,
  col.beta2 = NULL
)
```

Arguments

DMRsRanges	data.frame, the DMRs ranges, it must have the following columns: chr, start, end, beta diff. Other columns will be stored in the resulting output under the column other.
prom.length	numeric, length of promoters. Default = 1500
head.length	numeric, length of the first part of the gene, named head, starting from the TSS. If longer than the gene, the entire txs will be considered as head. Default = 1500
longest.trx	logical, option to use the longest transcript to represent the gene
annotation	character, database to use for transcripts mapping. Available "ensembl" or "ucsc". Default ="ensembl"
hg	character, Available "hg19", "hg38". Genome assembly version. Default = "hg19"
annotation.fast	logical, compute 1:1 mapping or 1:many - many:1 - many:many mapping. Default = TRUE
thr.beta	numeric, beta difference threshold to consider methylations. Default = 0.3
thr.cgis	numeric, length, in percentage, of methylated CGIs in order to be considered altered. Default = 0.4
col.betadiff	nuemric, column position for beta diff. in input table. Default = 4
col.beta1	numeric, column position for first sample beta values in input table
col.beta2	numeric, column position for second sample beta values in input table

Value

list, features overlapped by annotated DMRs

`annotateEnhancers` *Query database to find enhancer.*

Description

Query database of annotations results list to enhancer database in order to associate differentially methylated segments to genes.

Usage

```
annotateEnhancers(
  DMRsRanges,
  hg = "hg19",
  thr.beta = 0.3,
  overlap.param.thr = 40,
  param.type = "overlap.percentage",
  score.modifier = 0.5,
  col.betadiff = 4
)
```

Arguments

`DMRsRanges`	the DMRs ranges, it must have the following columns: chr, start, end, beta diff. Other columns will be stored in the resulting output under the column other.
`hg`	character, Available "hg19", "hg38". Genome assembly version. Default = "hg19"
`thr.beta`	numeric, beta difference threshold to consider methylations. Default = 0.3
`overlap.param.thr`	numeric, threshold value for selected parameter to filter methylations overlapping the selected features. Default = 100
`param.type`	character, threshold parameter to filter methylations overlapping the selected features. Accetped c("dmr.length", "overlap.length", "overlap.percentage"). Default = "overlap.length".
`score.modifier`	numeric, value between 0-1. It specifies how the final score is computed by assigning different weights to the methylation charactersistics of enhacners or to genes already involved in pathologies. By increasing this value to 1, resulting scores will be focused on discovering segments affecting gene expression. A value equal to 0 will focus the results on enahcners involving genes associated to pathologies, not considering the effect of methylation. Default = 0.5
`col.betadiff`	numeric, column position for beta diff. in input table. Default = 4

Value

a vector of presence or a data.frame

`associateTFs2Exprs` *Associate target genes to TFs and retrieve their expression*

Description

For each TF find and associate the target genes, within the annotation results, and retrieve expression

Usage

```
associateTFs2Exprs(
  annotatedDMRs,
  expressionProfile,
  active.features = c("promoters", "heads"),
  col.genes = 0,
  col.stat = 6,
  stat.thr = 0.05,
  col.logFC = 2,
  logfc.thr = 0,
  convert.genes = FALSE,
  convert.from,
  beta.thr = 0.3,
  overlap.param.thr = 30,
  param.type = "overlap.percentage"
)
```

Arguments

`annotatedDMRs`	anotated DMRs list resultingfrom annotateDMRs() or scoreAnnotatedDMRs()
`expressionProfile`	expression data.frame
`active.features`	character vectors containing features to correlate. Must be from names of resulting list from annotateDMRs. Additional feature names can be first exons (exons1) or first intron (intron1). To use more than one feature use c(). Default = c("promoters", "heads")
`col.genes`	numeric, the column of expressionProfile data.frame with gene Ids. If NULL geneIDs will be taken from rownames() of expressionProfile. Default = 0.
`col.stat`	numeric, the column of expressionProfile data.frame with the statistics to use. Default = 6.
`stat.thr`	numeric, threshold for statistical significance. Default = 0.05
`col.logFC`	numeric, the column of expressionProfile data.frame with log. fold change. Default = 2
`logfc.thr`	numeric, threshold value for log. fold change. Default = 0.
`convert.genes`	logical, used to indicate if gene ids have to be translated in official gene symbols. Default = FALSE
`convert.from`	character, used annotation for gene in expressionProfile to be converted to symbols gene IDs. Accepted: c("ENTREZID" ,"EXONID" ,"GENEBIOTYPE" ,"GENEID" ,"GENENAME" ,"PROTDOMID" ,"PROTEINDOMAINID" ,"PROTEINDOMAINSOURCE" ,"PROTEINID" ,""SEQSTRAND" ,"SYMBOL" ,"TXBIOTYPE" ,"TXID" ,"TXNAME" ,"UNIPROTID")

beta.thr	numeric, beta difference threshold value. Default = 0.3.
overlap.param.thr	numeric, threshold value for selected parameter to filter methylations overlapping the selected features. Default = 100
param.type	character, threshold parameter to filter methylations overlapping the selected features. Accetped c("dmr.length", "overlap.length", "overlap.percentage"). Default = "overlap.length".

Value

data.frame with TF methylation levels, target.genes expression

---

`formatDMRsInput` *Convert input table to proper format.*

Description

Convert input table to proper format. The firts three column of the input table must have chr, start, end coordinates.

Usage

```
formatDMRsInput(
  tableIn,
  thr.beta,
  col.betadiff = 4,
  col.beta1 = NULL,
  col.beta2 = NULL
)
```

Arguments

thr.beta	numeric, beta difference threshold to consider methylations. Default = 0.3
col.betadiff	numeric, column position for beta diff. in input table. Default = 4
col.beta1	numeric, column position for first sample beta values in input table
col.beta2	numeric, column position for second sample beta values in input table
DMRsRanges	the DMRs ranges, it must have the following columns: chr, start, end, beta diff. Other columns will be stored in the resulting output under the column other.

Value

data.frame of DMR ranges

genesToNCG *Find genes annotated in NCG.*

Description

Assess the presence of genes in results from annotateDMRs() in NCG database.

Usage

```
genesToNCG(annotatedDMRs, ncg, return.table = FALSE)
```

Arguments

annotatedDMRs	anotated DMRs list resultingfrom annotateDMRs() or scoreAnnotatedDMRs()
ncg	the NCG gene vectors
return.table	logical, option TRUE return a table instead a vector of presences. Default = TRUE

Value

data.frame or vector of presences

plotDMRs *Converts annotated DMRs in a plot*

Description

This function allows to track in a plot the beta value of a methylated segment mapped on a transcript of the human genome

Usage

```
plotDMRs(
  annotatedDMRs,
  symbol,
  annotation = "ensembl",
  hg = "hg19",
  beta1.name = NULL,
  beta2.name = NULL,
  beta.colors = c("red", "navy"),
  blackandwhite = FALSE,
  show.all.transcripts = FALSE,
  prom.width = 1500,
  path = NULL,
  coord.zoom = NULL,
  smartzoom = TRUE,
  height.pdf = 9,
  width.pdf = 16
)
```

Arguments

annotatedDMRs	anotated DMRs list resultingfrom annotateDMRs() or scoreAnnotatedDMRs()
symbol	character, gene symbol to plot.
annotation	character, "ensembl" or "ucsc". Annotation used to track the plot. Default = "esembl".
hg	character, "hg19" or "hg38". Genome Assembly version. Default = "hg19".
beta1.name	character, if unsued beta difference is plotted. character string identifying beta value of first sample in "other" column in results from annotateDMRs() or indentifying colname in input table used in annotateDMRs()
beta2.name	character, it identifies beta value of second sample in "other" column in results from annotateDMRs() or it identifies colname in input table used in annotateDMRs()
beta.colors	character vectors, colors of tracks for the first and the second bvalue, respectivetely. Default is c("red", "navy"). If beta diff is plotted, only the first element of vector is considered.
blackandwhite	logical, it allows to get all the plot in greyscale. Default = FALSE.
show.all.transcripts	logical, if TRUE all transcripts of genes are tracked, if FALSE only the longest transcript is tracked. Default = FALSE.
prom.width	integer, promoter lenght. Default = 1500.
path	logical, path where the plot is saved in a pdf file. If NULL the plot is not saved. Default = NULL.
coord.zoom	numeric vectors, coordinates of zoom region. If NULL the plot is not zoomed. Default = NULL.
smartzoom	logical, automatic zoom on the methylated region. Default = TRUE.
height.pdf	integer, hight pdf file. Default = 9.
width.pdf	integer, width pdf file. Default = 16.

Value

Plot of the beta value(s) mapped on transcript(s).

---

plotDMRs2Enrichr *Plot EnrichR Results*

Description

Visualize enrichment results and the contributes of hyper-metyhlated and hypo-methylated genes.

Usage

```
plotDMRs2Enrichr(
  enrichr.results,
  annotatedDMRs,
  stat = "P.value",
  n = 25,
```

```
  plot.type = "barplot",
  pal.col = "Dynamic",
  col.hyper = "#ff0000",
  col.hypo = "#00b3ff",
  thrs = c(0.01, 0.05),
  thrs.cols = c("green", "yellow")
)
```

Arguments

enrichr.results	data.frame resulting list from annotatedDMTs2Enrichr().
annotatedDMRs	anotated DMRs list resultingfrom annotateDMRs() or scoreAnnotatedDMRs()
stat	character, statistics to visualize. Accepted "P.value", "Adjusted.P.value" or "Overlap". Default = 'P.value'.
n	numeric, value indicating the number of enrichment to plot, starting from the most enriched, to visualize. Default = 25.
plot.type	character, compute or not different linear models for upregulated and downregulated genes. Accepted: "simple" or "splitted". Default = "splitted".
pal.col	character, the palette color for plot.type = 'lollipop'. Must be one from hcl.pals(). Default = 'Dynamic'.
col.hyper	character, the color representing hyper-methylated genes. Default = 'Grey70'.
col.hypo	character, the color representing hyper-methylated genes. Default = 'Grey30'.
thrs	numeric vectors, statistical thresholds to plot. Default = c(0.01, 0.05)
thrs.cols	characters vector, the colors to use for thresholds. Default = c('green', "yellow").

Value

plot of enrichR results

---

plotMethylationOverview

Plot a simple distribution of methylations.

Description

Plot distribution of beta values on chromosomes, this function has mainly an internal utility and is design for shinyApp version of package

Usage

```
plotMethylationOverview(annotatedDMRs, plot.type, palette)
```

Arguments

annotatedDMRs	anotated DMRs list resultingfrom annotateDMRs() or scoreAnnotatedDMRs()
plot.type	character, compute or not different linear models for upregulated and downregulated genes. Accepted: "simple" or "splitted". Default = "splitted".
palette	character, color palette of plot. It must be one resulting from hcl.pals()

Value

plot of enrichR results

`plotTFs2Exprs` *Plot the beta value of TF and related target genes expression*

Description

Barplot of the beta value of TF and the expression of the target genes

Usage

```
plotTFs2Exprs(
  associatedTFs2Expr,
  symbol,
  col.meth = "#8e0000",
  pals.bars = "Cold"
)
```

Arguments

`associatedTFs2Expr`	data.frame, results from associateTFs2Exprs().
`symbol`	character, feature to correlate. Must be from names of resulting list from annotateDMRs. Additional feature names can be first exons (exons1) or first intron (intron1). To use more than one feature use c().
`col.meth`	character, color for beta value. Defaut = "#8e0000" (red)
`pals.bars`	character, palette for gene expresion. hcl.pals() to show available. Default = "Cold"

Value

plot showing methylation levels of TF and expresion of target genes

---

`queryDatabase` *Query database to find pathologic genes.*

Description

Query elements of resulting annotations list to specified database in order to find pathogenic genes.

Usage

```
queryDatabase(
  DMRsRanges,
  db,
  return.table = TRUE,
  hold.columns,
  is.genomic.ranges = FALSE,
  thr = 0
)
```

Arguments

DMRsRanges	DMRs ranges
db	character, database to query
return.table	logical, optition to return a table instead a vector of presence. Default = TRUE
hold.columns	numeric vectors, column positions to hold when return.table=TRUE
is.genomic.ranges	logical, specifies if database is a GenomicRange objet or a data.frame. Default = FALSE
thr	numeric, threshold for beta difference values. Default = 0

Value

a vector of presence or a data.frame

---

runOnDesktop *Start Graphical User Interface*

Description

Start Graphical User Interface

Usage

```
runOnDesktop()
```

Value

GUI

---

scoreAnnotatedDMRs *Score the annotated methylation segments.*

Description

Assigns a score to annotated methylated segments resulting from annotateDMRs() function

Usage

```
scoreAnnotatedDMRs(
  annotatedDMRs,
  active.features = c("promoters", "heads"),
  score.modifier = 0.5
)
```

Arguments

annotatedDMRs	anotated DMRs list resultingfrom annotateDMRs()
active.features	character vectors, containing features to correlate. Must be from names of resulting list from annotateDMRs. Additional feature names can be first exons (exons1) or first intron (intron1). To use more than one feature use c(). Default = c("promoters", "heads")
score.modifier	numeric, value between 0-1. It specifies how the final score is computed by assigning different weights to the methylation charactersistics of enhacners or to genes already involved in pathologies. By increasing this value to 1, resulting scores will be focused on discovering segments affecting gene expression. A value equal to 0 will focus the results on enahcners involving genes associated to pathologies, not considering the effect of methylation. Default = 0.5

Value

data.frame of annotated DMRs with assigned scores

---

tfs2Enrichr *Query different databases to find enriched proceses.*

Description

Query TF's targeted genes in order to find enriched proceses.

Usage

```
tfs2Enrichr(
  associatedTFs2Expr,
  logfc.thr = 1,
  stat.filter = "P.value",
  stat.thr = 0.01,
  db = NULL
)
```

Arguments

associatedTFs2Expr	data.frame. Corresponding to resulting data.frame from associateTFs2Exprs().
logfc.thr	numeric value indicating logFC threshold. Default = 1.
stat.filter	character indicating which type of statistics use for filtering results. Accepted values: 'P.value' or 'Adjusted.P.value'. Default 'P.value'.
stat.thr	numeric value indicating the threshold to use for selcted statistical test. Default 0.01.
db	vector of characters indicating DBs to query. Default c("ClinVar_2019", "OMIM_Disease", "Elsevier_Pathway_Collection", "MSigDB_Hallmark_2020", "MSigDB_Oncogenic_Signatures", "GO_Biological_Process_2018", "Human_Phenotype_Ontology", "KEGG_2016", "NCI-Nature_2016", "Panther_2016", "Reactome_2016", "WikiPathways_2019_Human")

Value

data.frame with enriched processes

Index