Package 'RVA'

October 12, 2022

Title RNAseq Visualization Automation

Version 0.0.5

Description Automate downstream visualization & pathway analysis in RNAseq analysis. 'RVA' is a collection of functions that efficiently visualize RNAseq differential expression analysis result from summary statistics tables. It also utilize the Fisher's exact test to evaluate gene set or pathway enrichment in a convenient and efficient manner.

Maintainer Xingpeng Li <xingpeng.li@pfizer.com>

URL https://github.com/THERMOSTATS/RVA

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Suggests knitr, rmarkdown

VignetteBuilder knitr

biocViews

Imports GSVAdata (>= 1.22.0), clusterProfiler (>= 3.15.1), data.table (>= 1.12.8), edgeR (>= 3.28.1), org.Hs.eg.db (>= 3.10.0), ComplexHeatmap (>= 2.2.0), GSEABase (>= 1.48.0), circlize (>= 0.4.10), dplyr (>= 1.0.0), ggplot2 (>= 3.3.2), ggpubr (>= 0.4.0), grid (>= 3.6.1), gridExtra (>= 2.3), haven (>= 2.3.1), msigdbr (>= 7.1.1), plotly (>= 4.9.2.1), purrr (>= 0.3.4), rWikiPathways (>= 1.6.1), stringr (>= 1.4.0), tibble, tidyr (>= 1.1.0), XML, rlang

Depends R (>= 2.10)

NeedsCompilation no

Author Xingpeng Li [aut, cre] (<https://orcid.org/0000-0002-1331-1225>)

Repository CRAN

Date/Publication 2021-11-01 21:40:02 UTC

R topics documented:

`c2BroadSets` *This is data to be included in package*

Description

This is data to be included in package

Usage

```
c2BroadSets
```

Format

GeneSetCollection

Genesetcollection GeneSetCollection from BroadCollection

`cal.pathway.scores` *calculate pathway scores*

Description

Calculate pathway scores

Usage

```
cal.pathway.scores(
  data,
  pathway.db,
  gene.id.type,
  FCflag,
  FDRflag,
  FC.cutoff,
  FDR.cutoff,
  OUT.Directional = NULL,
  IS.list = FALSE,
  customized.pathways,
  ...
)
```

Arguments

data	A summary statistics table (data.frame) or data.list generated by DE analysis software like limma or DEseq2
pathway.db	pathway database used
gene.id.type	gene.id.type
FCflag	The column name (character) of fold change information, assuming the FC is log2 transformed. Default = "logFC".
FDRflag	The column name (character) of adjusted p value or FDR. Default = "adj.P.Val".
FC.cutoff	The fold change cutoff (numeric) selected to subset summary statistics table. Default = 1.5.
FDR.cutoff	The FDR cutoff selected (numeric) to subset summary statistics table. Default = 0.05.
OUT.Directional	logical, whether output directional or non-directional pathway analysis result, default: NULL.
IS.list	logical, whether the input is a list, default: NULL
customized.pathways	the customized pathways in the format of two column dataframe to be used in analysis
...	pass over parameters

Value

Returns a dataframe.

References

Xingpeng Li & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

---

calc.cfb *Calculate CFB*

Description

This function calculates the change from baseline.

Usage

```
calc.cfb(data, annot, baseline.flag, baseline.val)
```

Arguments

`data`	Dataframe with subject id, annotation flag, gene id and cpm value (from count tables) columns.
`annot`	A long-format dataframe with any pertinent treatment data about the samples. The only required column is one titled the `sample.id` value with values matching the column names of sample IDs in `data`. Additional columns can contain information such as treament compounds, dates of sample collection, or dosage quantities.
`baseline.flag`	A character vector of column names. These columns in `annot` contain the values to compare across.
`baseline.val`	A character vector of values. This vector must be the same length as `baseline.flag`, and the value at each index must represent a value from the column given by the corresponding index in `baseline.flag`.

`count_table` *This is data to be included in package*

Description

This is data to be included in package

Usage

```
count_table
```

Format

An example count table where row names are gene ID, each column is a sample

counttable count table ...

`dlPathwaysDB` *DL Pathways DB*

Description

Download gene database for enrichment.

Usage

```
dlPathwaysDB(pathway.db, customized.pathways = NULL, ...)
```

Arguments

pathway.db	The databse to be used for encrichment analysis. Can be one of the following, "rWikiPathways", "KEGG", "REACTOME", "Hallmark","rWikiPathways_aug_2020"
customized.pathways	the user provided pathway added for analysis.
...	pass over parameters

Value

Returns a dataframe.

References

Xingpeng Li & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

---

get.cpm.colors *Get CPM Colors*

Description

This function creates the color gradient for the cpm data.

Usage

```
get.cpm.colors(data)
```

Arguments

data	The CPM dataset.

---

get.cutoff.df *Create ggplot object for number of differntially expressed genes with different FDR and fold change cutoff.*

Description

This function processes dataframe from plot_cutoff_single function and produces a ggplot object which depicts the number of differntially expressed genes with different FDR and fold change cutoff.

Usage

```
get.cutoff.df(datin, pvalues, FCs, FCflag = "logFC", FDRflag = "adj.P.Val")
```

Arguments

datin	Dataframe from plot_cutoff_single.
pvalues	A set of p-values for FDR cutoff to be checked.
FCs	A set of fold change cutoff to be checked.
FCflag	The column name of the log2FC in the summary statistics table.
FDRflag	The column name of the False Discovery Rate (FDR) in the summary statistics table.

---

get.cutoff.ggplot *Create ggplot object for number of differntially expressed genes with different FDR and fold change cutoff.*

Description

This function processes dataframe from plot_cutoff_single function and produces a ggplot object which depicts the number of differntially expressed genes with different FDR and fold change cutoff.

Usage

```
get.cutoff.ggplot(df, FCflag, FDRflag)
```

Arguments

df	Dataframe from plot_cutoff_single.
FCflag	The column name of the log2FC in the summary statistics table.
FDRflag	The column name of the False Discovery Rate (FDR) in the summary statistics table.

---

make.cutoff.plotly *Create plotly object for number of DE genes at different cutoff combinations*

Description

This function processes summary statistics table generated by differential expression analysis like `limma` or `DESeq2` to produce an interactibe visual object which depicts the number of differntially expressed genes with different FDR and fold change cutoff.

Usage

```
make.cutoff.plotly(df)
```

Arguments

df	Summary statistics table from limma or DEseq2, where each row is a gene.

`multiPlot` *Multi Plot*

Description

Multi plot is for directional and non-directional plots

Usage

```
multiPlot(allID, backup.d.sig, nd.res, ...)
```

Arguments

`allID`	A vector of all pathway ID's from directional and non directional enriched datasets.
`backup.d.sig`	A dataframe type of object with directional pathways data prior to any cutoff's being applied
`nd.res`	A dataframe type of object with non directional pathways data prior to any cutoff's being applied
`...`	pass on variables

Details

Multi plot is for directional and non-directional plots, when one of the plots doesn't contain observations.

Value

Returns ggplot.

References

Xingpeng Li & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`nullreturn` *Null Return*

Description

The function takes in a boolean value and a numeric value, which it uses to decide what to output.

Usage

```
nullreturn(IS.list, type = 1)
```

Arguments

IS.list	Indicator of whether the data frame being input is list or not.
type	If type = 1(default) return directional null plot. If type = 2 return non directional null plot.

Details

nullreturn is a function that returns NULL for single df inputs that don't hold true for threshold values. It returns an empty dataframe for list inputs which don't satisfy the cutoff's

Value

The function returns either returns a data frame or the value NULL.

References

Xingpeng Li & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

---

plot_cutoff *Check number of DE genes at different cutoff combinations*

Description

This function processes summary statistics table generated by differential expression analysis like `limma` or `DESeq2` to evaluate the number of differntially expressed genes with different FDR and fold change cutoff.

Usage

```
plot_cutoff(
  data = data,
  comp.names = NULL,
  FCflag = "logFC",
  FDRflag = "adj.P.Val",
  FCmin = 1.2,
  FCmax = 2,
  FCstep = 0.1,
  p.min = 0,
  p.max = 0.2,
  p.step = 0.01,
  plot.save.to = NULL,
  gen.3d.plot = TRUE,
  gen.plot = TRUE
)
```

Arguments

data	Summary statistics table or a list of summary statistics tables from limma or DEseq2, where each row is a gene.
comp.names	A character vector that contains the comparison names which correspond to the same order as data.
FCflag	The column name of the log2FC in the summary statistics table. Default = "logFC".
FDRflag	The column name of the False Discovery Rate (FDR) in the summary statistics table. Default = "adj.P.Val".
FCmin	The minimum starting fold change cutoff to be checked, so the minimum fold change cutoff to be evaluated will be FCmin + FCstep, FCmin default = 1.
FCmax	The maximum fold change cutoff to be checked, default = 2.
FCstep	The step from the minimum to maximum fold change cutoff, one step increase at a time, default = 0.01.
p.min	The minimum starting FDR cutoff to be checked, so the minimum fold change cutoff to be evaluated will be p.min + p.step, p.min default = 0.
p.max	The maximum FDR cutoff to be checked, default = 0.2.
p.step	The step from the minimum to maximum fold change cutoff, one step increase at a time, default = 0.005.
plot.save.to	The address where to save the plot from simplified cutoff combination with FDR of 0.01, 0.05, 0.1, and 0.2.
gen.3d.plot	Whether generate a 3d plotly object to visualize the result, only applys to single dataframe input, default = F.
gen.plot	Whether generate a plot to visualize the result, default = T.

Details

The function takes the summary statistics and returns a list which contains 3 objects: a table which describes the number of DE genes with different cutoff combinations of FDR and fold change, a ggplot object which depicts a simplified version of cutoff selection combination, and a plotly 3d visulization object which depicts a high resolution of cutoff combinations. The default range of the fold change is from 1 to 2, and p value is from 0 to 0.2, with the step of 0.01 for FC and 0.005 for FDR.

Value

If the input data is a data list, then a multi-facet ggplot plot object which contains each of the summary statistics table will be returned; otherwise, if the input data is a data frame, then the function will return a list which contains 3 elements:

df.sub	A dataframe, which contains the number of genes(3rd column) with FDR (1st column), Fold Change (2nd column)
plot3d	A plotly object to show the 3d illustration of all possible cutoff selectiosn and the number of DE genes in the 3d surface
gp	A ggplot object to show the simplified cutoff combination result

References

Xingpeng Li & Olya Besedina, RVA - RNAseq Visualization Automation tool.

Examples

```
plot_cutoff(Sample_summary_statistics_table)

plot_cutoff(data = list(Sample_summary_statistics_table, Sample_summary_statistics_table1),
           comp.names = c("A", "B"))
```

---

plot_cutoff_single *Create plotly object for number of DE genes at different cutoff combinations*

Description

This function processes summary statistics table generated by differential expression analysis like `limma` or `DESeq2` and produces a table which contains gene counts for each of the pvalue and FC combination

Usage

```
plot_cutoff_single(datin, FCflag, FDRflag, FCs, pvalues)
```

Arguments

`datin`	Summary statistics table from limma or DEseq2, where each row is a gene.
`FCflag`	The column name of the log2FC in the summary statistics table.
`FDRflag`	The column name of the False Discovery Rate (FDR) in the summary statistics table.
`FCs`	A set of fold change cutoff to be checked.
`pvalues`	A set of p-values for FDR cutoff to be checked.

plot_gene *Plot gene expression*

Description

This is the function to process the gene count table to show gene expression variations over time or across groups.

Usage

```
plot_gene(
  data = ~dat,
  anno = ~meta,
  gene.names = c("AAAS", "A2ML1", "AADACL3"),
  ct.table.id.type = "ENSEMBL",
  gene.id.type = "SYMBOL",
  treatment = "Treatment",
  sample.id = "sample_id",
  time = "day",
  log.option = TRUE,
  plot.save.to = NULL,
  input.type = "count"
)
```

Arguments

data	Count table in the format of dataframe with gene id as row.names.
anno	Annotation table that provides design information.
gene.names	Genes to be visualized, in the format of character vector.
ct.table.id.type	The gene id format in data should be one of: ACCNUM, ALIAS, ENSEMBL, ENSEMBLPROT, ENSEMBLTRANS, ENTREZID, ENZYME, EVIDENCE, EVIDENCEALL, GENENAME, GO, GOALL, IPI, MAP, OMIM, ONTOLOGY, ONTOLOGYALL, PATH, PFAM, PMID, PROSITE, REFSEQ, SYMBOL, UCSCKG, UNIGENE, UNIPROT.
gene.id.type	The gene id format of gene.names, should be one of: ACCNUM, ALIAS, ENSEMBL, ENSEMBLPROT, ENSEMBLTRANS, ENTREZID, ENZYME, EVIDENCE, EVIDENCEALL, GENENAME, GO, GOALL, IPI, MAP, OMIM, ONTOLOGY, ONTOLOGYALL, PATH, PFAM, PMID, PROSITE, REFSEQ, SYMBOL, UCSCKG, UNIGENE, UNIPROT.
treatment	The column name to specify treatment groups.
sample.id	The column name to specify sample IDs.
time	The column name to specify different time points.
log.option	Logical option, whether to log2 transform the CPM as y-axis. Default = True.

plot.save.to	The address to save the plot from simplified cutoff combination with FDR of 0.01, 0.05, 0.1, and 0.2.
input.type	One of count or cpm indicating what the input data type is. If count, the CPM of the input data will be calculated using edgeR::cpm(). Default = count.

Details

The function takes the gene counts and returns a ggplot that shows gene expression variation over time or group.

Value

The function returns a ggplot object.

References

Xingpeng Li,Tatiana Gelaf Romer & Aliyah Olaniyan, RVA - RNAseq Visualization Automation tool.

Examples

```
plot_gene(data = count_table,
anno = sample_annotation)
```

---

plot_heatmap.cfb *Plot a CFB Heatmap*

Description

An alias for plot_heatmap.expr(annot, cpm, fill = "CFB", ...).

Usage

```
plot_heatmap.cfb(cpm, annot, title = "RVA CFB Heatmap", ...)
```

Arguments

cpm	cpm data
annot	A long-format dataframe with any pertinent treatment data about the samples. The only required column is one titled the sample.id value with values matching the column names of sample IDs in data. Additional columns can contain information such as treament compounds, dates of sample collection, or dosage quantities.
title	A title for the heatmap. Default = "RVA Heatmap".
...	pass over parameters

plot_heatmap.cpm *Plot a CPM Heatmap*

Description

An alias for `plot_heatmap.expr(annot, cpm, fill = "CPM", ...)`.

Usage

```
plot_heatmap.cpm(cpm, annot, title = "RVA CPM Heatmap", ...)
```

Arguments

cpm	cpm data
annot	A long-format dataframe with any pertinent treatment data about the samples. The only required column is one titled the `sample.id` value with values matching the column names of sample IDs in `data`. Additional columns can contain information such as treament compounds, dates of sample collection, or dosage quantities.
title	A title for the heatmap. Default = "RVA Heatmap".
...	pass over parameters

plot_heatmap.expr *Plot Heatmap From Raw CPM*

Description

Create a heatmap with either CFB or CPM averaged across individual samples.

Usage

```
plot_heatmap.expr(
  data = ~count,
  annot = ~meta,
  sample.id = "sample_id",
  annot.flags = c("day", "Treatment", "tissue"),
  ct.table.id.type = "ENSEMBL",
  gene.id.type = "SYMBOL",
  gene.names = NULL,
  gene.count = 10,
  title = "RVA Heatmap",
  fill = "CFB",
  baseline.flag = "day",
  baseline.val = "0",
  plot.save.to = NULL,
  input.type = "count"
)
```

Arguments

data	A wide-format dataframe with geneid rownames, sample column names, and fill data matching input.type.
annot	A long-format dataframe with any pertinent treatment data about the samples. The only required column is one titled the sample.id value with values matching the column names of sample IDs in data. Additional columns can contain information such as treament compounds, dates of sample collection, or dosage quantities.
sample.id	The column name to specify sample ID.
annot.flags	A vector of column names corresponding to column names in annot which will be used to define the x-axis for the heatmap. Default = c("day", "dose").
ct.table.id.type	The gene id format in data should be one of: ACCNUM, ALIAS, ENSEMBL, ENSEMBLPROT, ENSEMBLTRANS, ENTREZID, ENZYME, EVIDENCE, EVIDENCEALL, GENENAME, GO, GOALL, IPI, MAP, OMIM, ONTOLOGY, ONTOLOGYALL, PATH, PFAM, PMID, PROSITE, REFSEQ, SYMBOL, UCSCKG, UNIGENE, UNIPROT.
gene.id.type	The gene id format of gene.names, should be one of: ACCNUM, ALIAS, ENSEMBL, ENSEMBLPROT, ENSEMBLTRANS, ENTREZID, ENZYME, EVIDENCE, EVIDENCEALL, GENENAME, GO, GOALL, IPI, MAP, OMIM, ONTOLOGY, ONTOLOGYALL, PATH, PFAM, PMID, PROSITE, REFSEQ, SYMBOL, UCSCKG, UNIGENE, UNIPROT.
gene.names	A character vector or list of ensembl IDs for which to display gene information. If NULL, all genes will be included. Default = NULL.
gene.count	The number of genes to include, where genes are selected based on ranking by values in fill. Default = 10.
title	A title for the heatmap. Default = "RVA Heatmap".
fill	One of c("CPM", "CFB") to fill the heatmap cells with. Default = "CFB".
baseline.flag	A character vector of column names. If fill = "CFB", these columns in annot contain the values to compare across. Ignored if fill = "CPM". Default = "timepoint".
baseline.val	A character vector of values. This vector must be the same length as baseline.flag, and the value at each index must represent a value from the column given by the corresponding index in baseline.flag. The samples corresponding to these values will be used as a baseline when calculating CFB. Ignored if fill = "CPM". Default = "Week 0".
plot.save.to	The address to save the heatmap plot.
input.type	One of count or cpm indicating what the input data type is. If count, the CPM of the input data will be calculated using edgeR::cpm(). Default = count.

Details

The function takes raw CPM data and returns both a list containing a data frame with values based on the fill parameter and a heatmap plot.

Value

The function returns a list with 2 items:

df.sub	"A data frame of change from baselines values (fill = CFB in this example) for each gene id that is divided by a combination of treatment group and time point
gp	A Heatmap object from ComplexHeatmap which can be plotted

References

Xingpeng Li,Tatiana Gelaf Romer & Aliyah Olaniyan, RVA - RNAseq Visualization Automation tool.

Examples

```
plot <- plot_heatmap.expr(data = count_table[,1:20],annot = sample_annotation[1:20,])
```

---

plot_pathway *Pathway analysis and visualization*

Description

This is the function to do pathway enrichment analysis (and visualization) with rWikipathways (also KEGG, REACTOME & Hallmark) from a summary statistics table generated by differential expression analysis like limma or DESeq2.

Usage

```
plot_pathway(
  data = ~df,
  comp.names = NULL,
  gene.id.type = "ENSEMBL",
  FC.cutoff = 1.2,
  FDR.cutoff = 0.05,
  FCflag = "logFC",
  FDRflag = "adj.P.Val",
  Fisher.cutoff = 0.1,
  Fisher.up.cutoff = 0.1,
  Fisher.down.cutoff = 0.1,
  plot.save.to = NULL,
  pathway.db = "rWikiPathways",
  customized.pathways = NULL,
  ...
)
```

Arguments

data	A summary statistics table (data.frame) or data.list generated by DE analysis software like limma or DEseq2, where rownames are gene id.
comp.names	A character vector containing the comparison names corresponding to the same order of the data.list. Default = NULL.
gene.id.type	The gene id format in data should be one of: ACCNUM, ALIAS, ENSEMBL, ENSEMBLPROT, ENSEMBLTRANS, ENTREZID, ENZYME, EVIDENCE, EVIDENCEALL, GENENAME, GO, GOALL, IPI, MAP, OMIM, ONTOLOGY, ONTOLOGYALL, PATH, PFAM, PMID, PROSITE, REFSEQ, SYMBOL, UCSCKG, UNIGENE, UNIPROT.
FC.cutoff	The fold change cutoff (numeric) selected to subset summary statistics table. Default = 1.5.
FDR.cutoff	The FDR cutoff selected (numeric) to subset summary statistics table. Default = 0.05.
FCflag	The column name (character) of fold change information, assuming the FC is log2 transformed. Default = "logFC".
FDRflag	The column name (character) of adjusted p value or FDR. Default = "adj.P.Val".
Fisher.cutoff	The FDR cutoff selected (numeric) for the pathway enrichment analysis' Fisher's exact test with all determined Differentially Expressed (DE) genes by FC.cutoff and FDR.cutoff.
Fisher.up.cutoff	The FDR cutoff selected (numeric) for the pathway enrichment analysis' Fisher's exact test with the upregulated gene set.
Fisher.down.cutoff	The FDR cutoff selected (numeric) for the pathway enrichment analysis' Fisher's exact test with the downregulated gene set.
plot.save.to	The address to save the plot from simplified cutoff combination with FDR of 0.01, 0.05, 0.1, and 0.2.
pathway.db	The databse to be used for encrichment analysis. Can be one of the following, "rWikiPathways", "KEGG", "REACTOME", "Hallmark","rWikiPathways_aug_2020".
customized.pathways	the customized pathways in the format of two column dataframe (column name as "gs_name" and "entrez_gene") to be used in analysis.
...	pass on variables

Details

The function takes the summary statistics table and use user selected parameter based on check.cutoff to do pathway enrichment analysis

Value

The function returns a list of 5 objects:

1	result table from directional pathway enrichment analysis

2	result table from non-directional pathway enrichment analysis
3	plot from directional pathway enrichment analysis
4	plot from non-directional pathway enrichment analysis
5	plot combining both directional and non-directional plot

References

Xingpeng Li & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

Examples

```
result <- plot_pathway(data = Sample_summary_statistics_table,
gene.id.type = "ENSEMBL",
FC.cutoff = 1.5,
p.cutoff = 0.05,
pathway.db = "rWikiPathways_aug_2020"
)
```

---

plot_qq *Plot qqplot*

Description

This function generates a QQ-plot object with confidence interval from summary statistics table generated by differential expression analysis like `limma` or `DESeq2`.

Usage

```
plot_qq(
  data = data,
  comp.names = NULL,
  p.value.flag = "P.Value",
  ci = 0.95,
  plot.save.to = NULL
)
```

Arguments

`data`	Summary statistics table or a list that contains multiple summary statistics tables from limma or DEseq2, where each row is a gene.
`comp.names`	A character vector that contains the comparison names which correspond to the same order as `data`. No default.
`p.value.flag`	The column name of `P-VALUE` (NOT FDR, NO multiplicity adjusted p-value) in the summary statistics table. Default = "P.Value".
`ci`	Confidence interval. Default = 0.95
`plot.save.to`	The file name and the address where to save the qq-plot "~/address_to_folder/qqplot.png". Default = NULL.

Details

The function produces the qqplot to evaluate the result from differential expression analysis. The output is a ggplot object.

Value

The function return a ggplot object of qqplot

References

Xingpeng Li & Tatiana Gelaf Romer & Olya Besedina, RVA - RNAseq Visualization Automation tool.

Examples

```
plot_qq(data = Sample_summary_statistics_table)
plot_qq(data = list(Sample_summary_statistics_table, Sample_summary_statistics_table1),
        comp.names = c("A","B"))
```

`plot_volcano` *Plot volcanoplot*

Description

This function processes the summary statistics table generated by differential expression analysis like `limma` or `DESeq2` to show on the volcano plot with the highlight gene set option (like disease related genes from Disease vs Healthy comparison).

Usage

```
plot_volcano(
  data = data,
  comp.names = NULL,
  geneset = NULL,
  geneset.FCflag = "logFC",
  highlight.1 = NULL,
  highlight.2 = NULL,
  upcolor = "#FF0000",
  downcolor = "#0000FF",
  plot.save.to = NULL,
  xlim = c(-4, 4),
  ylim = c(0, 12),
  FCflag = "logFC",
  FDRflag = "adj.P.Val",
  highlight.FC.cutoff = 1.5,
  highlight.FDR.cutoff = 0.05,
```

```
  title = "Volcano plot",
  xlab = "log2 Fold Change",
  ylab = "log10(FDR)"
)
```

Arguments

data	Summary statistics table or a list contain multiple summary statistics tables from limma or DEseq2, where each row is a gene.
comp.names	A character vector that contains the comparison names which correspond to the same order as data. Required if data is list. No default.
geneset	Summary statistic table that contains the genes which needed to be highlighted, the gene name format (in row names) needs to be consistent with the main summary statistics table). For example, this summary statistics table could be the output summary statistics table from the Disease vs Healthy comparison (Only contains the subsetted significant genes to be highlighted).
geneset.FCflag	The column name of fold change in geneset, Default = "logFC".
highlight.1	Genes to be highlighted, in the format of a vector consists of gene names. The gene name format needs to be consistent to the main summary statistics table.
highlight.2	Genes to be highlighted, in the format of a vector consists of gene names. The gene name format needs to be consistent to the main summary statistics table.
upcolor	The color of the gene names in highlight.1 or the positive fold change gene in geneset, default = "#FDE725FF" (viridis color palette).
downcolor	The color of the gene names in highlight.2 or the negative fold change gene in geneset, default = "#440154FF" (viridis color palette).
plot.save.to	The file name and address where to save the volcano plot, e.g. "~/address_to_folder/volcano_plot.png".
xlim	Range of x axis. Default = c(-3,3).
ylim	Range of x axis. Default = c(0,6).
FCflag	Column name of log2FC in the summary statistics table. Default = "logFC".
FDRflag	Column name of FDR in the summary statistics table. Default = "adj.P.Val".
highlight.FC.cutoff	Fold change cutoff line want to be shown on the plot. Default = 1.5.
highlight.FDR.cutoff	FDR cutoff shades want to be shown on the plot. Default = 0.05.
title	The plot title. Default "Volcano plot".
xlab	The label for x-axis. Default "log2 Fold Change".
ylab	The label for y-axis. Default "log10(FDR)".

Details

The function takes the summary statistics table and returns a ggplot, with the option to highlight genes, e.g. disease signature genes, the genes which are up-regulated and down-regulated in diseased subjects.

Value

The function return a volcano plot as a ggplot object.

References

Xingpeng Li & Tatiana Gelaf Romer & Olya Besedina, RVA - RNAseq Visualization Automation tool.

Examples

```
plot_volcano(data = Sample_summary_statistics_table,
             geneset = Sample_disease_gene_set)

plot_volcano(data = list(Sample_summary_statistics_table, Sample_summary_statistics_table1),
            comp.names = c("A", "B"),
            geneset = Sample_disease_gene_set)
```

---

prettyGraphs *Pretty Graphs*

Description

Special cases where list input and at least one treatment has signal but others don't.

Usage

```
prettyGraphs(vizdf, ...)
```

Arguments

vizdf	A dataframes of enriched pathways.
...	pass on variables

Details

Pretty Graphs is a function specifically meant to be in cases where one of the input treatments meet cutoff, but one or more of the other treatments don't meet the cutoff values. This is important so that ggplot doesn't throw any errors.

Value

Returns a dataframe.

References

Xingpeng Li & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

produce.cutoff.message *Create a message about fold change and pvalues used to produce the plot.*

Description

This function processes summary statistics table generated by differential expression analysis like limma or DESeq2 and produces a message about pvalues and fold change used.

Usage

```
produce.cutoff.message(
  data,
  FCmin,
  FCmax,
  FCstep,
  FDRflag,
  p.min,
  p.max,
  p.step
)
```

Arguments

data	Summary statistics table from limma or DEseq2, where each row is a gene.
FCmin	The minimum starting fold change cutoff to be checked, so the minimum fold change cutoff to be evaluated will be FCmin + FCstep, FCmin default = 1.
FCmax	The maximum fold change cutoff to be checked, default = 2.
FCstep	The step from the minimum to maximum fold change cutoff, one step increase at a time, default = 0.01.
FDRflag	The column name of the False Discovery Rate (FDR) in the summary statistics table.
p.min	The minimum starting FDR cutoff to be checked, so the minimum fold change cutoff to be evaluated will be p.min + p.step, p.min default = 0.
p.max	The maximum FDR cutoff to be checked, default = 0.2.
p.step	The step from the minimum to maximum fold change cutoff, one step increase at a time, default = 0.005.

produce.cutoff.warning *Create a warning about pvalue or FDR minimum value*

Description

This function processes summary statistics table generated by differential expression analysis like limma or DESeq2 and produces a warning about pvalue or FDR minimum value

Usage

```
produce.cutoff.warning(data, FDRflag)
```

Arguments

data	Summary statistics table from limma or DEseq2, where each row is a gene.
FDRflag	The column name of the False Discovery Rate (FDR) in the summary statistics table.

reformat.ensembl *Reformat Ensembl GeneIDs*

Description

This is the function to exclude the version number from the input ensembl type gene ids.

This is the function to exclude the version number from the input ensembl type gene ids.

Usage

```
reformat.ensembl(logcpm, ct.table.id.type)

reformat.ensembl(logcpm, ct.table.id.type)
```

Arguments

logcpm	The input count table transformed into log counts per million.
ct.table.id.type	The gene id format in logcpm should be one of: ACCNUM, ALIAS, ENSEMBL, ENSEMBLPROT, ENSEMBLTRANS, ENTREZID, ENZYME, EVIDENCE, EVIDENCEALL, GENENAME, GO, GOALL, IPI, MAP, OMIM, ONTOLOGY, ONTOLOGYALL, PATH, PFAM, PMID, PROSITE, REFSEQ, SYMBOL, UCSCKG, UNIGENE, UNIPROT.

sample_annotation *This is data to be included in package*

Description

This is data to be included in package

Usage

```
sample_annotation
```

Format

Sample annotation document

sample_id sample name

tissue tissue for comparison

subject_id subject id

day time points ...

sample_count_cpm *This is data to be included in package*

Description

This is data to be included in package

Usage

```
sample_count_cpm
```

Format

An example cpm table where row names are gene ID, each column is a sample

counttable count cpm table ...

Sample_disease_gene_set

This is data to be included in package

Description

This is data to be included in package

Usage

```
Sample_disease_gene_set
```

Format

An example disease gene set from summary statistics table as dataframe, row names are gene ID the summary statistics can be calculated from disease vs healthy, which is this example.

logFC log2 fold change from comparison

AveExpr Average expression for this gene

P.Value p value

adj.P.Val adjusted p value or FDR ...

Sample_summary_statistics_table

This is data to be included in package

Description

This is data to be included in package

Usage

```
Sample_summary_statistics_table
```

Format

An example summary statistics table as dataframe, row names are gene ID

logFC log2 fold change from comparison

AveExpr Average expression for this gene

P.Value p value

adj.P.Val adjusted p value or FDR ...

Sample_summary_statistics_table1

This is data to be included in package

Description

This is data to be included in package

Usage

```
Sample_summary_statistics_table1
```

Format

Second example summary statistics table as dataframe, row names are gene ID

logFC log2 fold change from comparison

AveExpr Average expression for this gene

P.Value p value

adj.P.Val adjusted p value or FDR ...

secondCutoffErr *Second Cutoff Error*

Description

The function takes in a list of dataframe, comp names and a specified type, to output a dataframe styled for ggplot.

Usage

```
secondCutoffErr(df, comp.names, TypeQ = 1)
```

Arguments

df	A list of dataframes.
comp.names	a character vector contain the comparison names corresponding to the same order to the dat.list. default = NULL.
TypeQ	If type = 1(default) return directional null plot. If type = 2 return non directional null plot.

Details

secondCutoffErr is a function specifically meant to be used for list inputs. It is used for cases where after applying filter to the data, one of the comparison ID gets left out, this adversely effects the ggplot

Value

Returns a dataframe.

References

Xingpeng Li & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`transform.geneid` *Transform GeneIDs*

Description

This is the function to transform the input gene id type to another gene id type.

This is the function to transform the input gene id type to another gene id type.

Usage

```
## S3 method for class 'geneid'
transform(gene.names, from = ~gene.id.type, to = ~ct.table.id.type)

## S3 method for class 'geneid'
transform(gene.names, from = ~gene.id.type, to = ~ct.table.id.type)
```

Arguments

`gene.names`	Genes,in the format of character vector, to be transformed.
`from`	The gene id format of `gene.names`, should be one of: ACCNUM, ALIAS, ENSEMBL, ENSEMBLPROT, ENSEMBLTRANS, ENTREZID, ENZYME, EVIDENCE, EVIDENCEALL, GENENAME, GO, GOALL, IPI, MAP, OMIM, ONTOLOGY, ONTOLOGYALL, PATH, PFAM, PMID, PROSITE, REFSEQ, SYMBOL, UCSCKG, UNIGENE, UNIPROT.
`to`	The new gene id format should be one of: ACCNUM, ALIAS, ENSEMBL, ENSEMBLPROT, ENSEMBLTRANS, ENTREZID, ENZYME, EVIDENCE, EVIDENCEALL, GENENAME, GO, GOALL, IPI, MAP, OMIM, ONTOLOGY, ONTOLOGYALL, PATH, PFAM, PMID, PROSITE, REFSEQ, SYMBOL, UCSCKG, UNIGENE, UNIPROT.

`validate.annot` *Validate Annotation Table*

Description

Ensure that an annotation has all of the required columns.

Usage

```
validate.annot(
  data,
  annot,
  annot.flags,
  sample.id,
  fill = "CPM",
  baseline.flag = NULL,
  baseline.val = NULL
)
```

Arguments

`data`	The input count data.
`annot`	The annotation dataframe.
`annot.flags`	The vector of annotation flags passed by the user.
`sample.id`	Sample id label to check if in annot.
`fill`	The fill value indicated by the user,"count" or "CPM".
`baseline.flag`	The baseline.flag passed by the user.
`baseline.val`	The baseline value passed by the user.

Details

The function will check the following:

- The `annot.flags` values are columns in `annot`
- If `fill` = "cfb": validate the `baseline.flag` and `baseline.val` parameters.
- `sample.id` is a column in `annot`.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

validate.baseline *Validate Baseline Values*

Description

Ensures that user-input `baseline.val` and `baseline.flag` parameters are valid with respect to the `annot` dataframe.

Usage

```
validate.baseline(annot, baseline.val, baseline.flag)
```

Arguments

`annot`	The annotation dataframe.
`baseline.val`	The baseline value passed by the user.
`baseline.flag`	The baseline.flag passed by the user.

Details

Specifically, validates that `baseline.flag` value(s) are columns in `annot`, and that `baseline.val` value(s) occur at least once in their respective `baseline.flag` columns.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

validate.col.types *Check Summary Statistics Required Column Types*

Description

`FCflag` and `FDRflag` must be numeric.

Usage

```
validate.col.types(datin, name = 1, flags)
```

Arguments

`datin`	the summary statistics file.
`name`	summary statistics file position indicator
`flags`	`FCflag` or `FDRflag` to be checked

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`validate.comp.names` *Validate Comp Names*

Description

This function ensures that when a list of data frames are used as input the the number of comp names are the same as the number of data frames.

Usage

```
validate.comp.names(comp.names, data)
```

Arguments

`comp.names`	a character vector contain the comparison names corresponding to the same order to the `dat.list`. default = NULL.
`data`	summary statistics table (data.frame) from limma or DEseq2, where rownames are gene id.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`validate.data` *Validate Data Input*

Description

Ensures that the data input has the required formatting.

Usage

```
validate.data(data)
```

Arguments

`data`	The wide-format dataframe with input data.

Details

Specifically, checks if `data` has rownaems and that all other columns can be coerced to numeric.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`validate.data.annot` *Validate Data in the Context of Annotation*

Description

Ensures that the annotation file matches the data file with respect to sample IDs. Throws warnings if there are discrepencies.

Usage

```
validate.data.annot(data, annot, sample.id)
```

Arguments

`data`	input data
`annot`	annotation file
`sample.id`	sample id in the input

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`validate.FC` *Validate Foldchange*

Description

This function ensures the fold change minimum, maximum, and step are valid.

Usage

```
validate.FC(FCmin, FCmax, FCstep)
```

Arguments

`FCmin`	The minimum starting fold change cutoff to be checked, so the minimum fold change cutoff to be evaluated will be FCmin + FCstep, FCmin default = 1.
`FCmax`	The maximum fold change cutoff to be checked, default = 2.
`FCstep`	The step from the minimum to maximum fold change cutoff, one step increase at a time, default = 0.01.

Details

Specifically it checks that the FCmax is greater than the FCmin, that at least 1 FCstep can fit within the FCmax and FCmin, that FCmax and FCmin values are non-negative, and that FCstep is positive.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`validate.flag` *Validate Flag Value Is Valid*

Description

Enures that the `value` is one of `Options` and throws an error otherwise.

Usage

```
validate.flag(value, name, Options)
```

Arguments

`value`	The user-input value for the parameter
`name`	The name of the parameter to be displayed in the error
`Options`	A vector of valid values for `value`

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`validate.genes.present` *Validate genes present*

Description

Checks how many of the gene id's in the dataset are there in the geneset.

Usage

```
validate.genes.present(data.genes, geneset)
```

Arguments

data.genes	The gene id's.
geneset	a summary statistic table contain the genes want to be highlighted, the gene name format (in row names) needs to be consistent to the main summary statistics table). For example, this summary statistics table coulb be the output summary statistics table from Disease vs Healthy comparison (Only contain the subsetted significant genes want to be highlighted).

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

---

validate.geneset *Validate Geneset*

Description

This function ensures that the input geneset to check.cutoff is formatted properly and in a usable form.

Usage

```
validate.geneset(data, geneset, highlight.1, highlight.2)
```

Arguments

data	summary statistics table or a list contain multiple summary statistics tables from limma or DEseq2, where each row is a gene.
geneset	a summary statistic table contain the genes want to be highlighted, the gene name format (in row names) needs to be consistent to the main summary statistics table). For example, this summary statistics table coulb be the output summary statistics table from Disease vs Healthy comparison (Only contain the subsetted significant genes want to be highlighted).
highlight.1	genes want to be highlighted, in the format of a vector consists of gene names. The gene name format needs to be consistent to the main summary statistics table.
highlight.2	genes want to be highlighted, in the format of a vector consists of gene names. The gene name format needs to be consistent to the main summary statistics table.

Details

The function ensures that only a dataframe or vectors are supplied, that at least one or the other is supplied, and that their formatting is correct if supplied. It also checks if any of the genes overlap with the genes in the datanames.

Value

A character value indicating if the geneset was passed as a dataframe (`df`) or two vectors (`vec`), if a list is input the number of returned values equal the length of the list

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`validate.numeric` *Validate Numeric Column*

Description

Ensures that a column in a dataframe which must be numeric is numeric and throws an error otherwise.

Usage

```
validate.numeric(datin, col, name = 1)
```

Arguments

`datin`	The data in question.
`col`	The column to validate as numeric.
`name`	the position of dataset

Details

This specifically checks if any of the values in the column can be coerced as numeric.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

validate.pathways.db *Validate Pathways DB*

Description

To ensure selected db name is correct.

Usage

```
validate.pathways.db(pathway.db, customized.pathways)
```

Arguments

pathway.db	The databse to be used for encrichment analysis. Can be one of the following, "rWikiPathways", "KEGG", "REACTOME", "Hallmark","rWikiPathways_aug_2020"
customized.pathways	the customized pathways in the format of two column dataframe (column name as "gs_name" and "entrez_gene") to be used in analysis

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

validate.pval.range *Validate P-value Range*

Description

Error-handling for invalid p-value.

Usage

```
validate.pval.range(pval, name)
```

Arguments

pval	The pvalue
name	The name of the value to include in the error.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

validate.pvalflag *Validate pval flag*

Description

To ensure p value flags are the same accross datasets.

Usage

```
validate.pvalflag(data, value)
```

Arguments

data	A list of summary statistics table (data.frame) from limma or DEseq2, where rownames are gene id.
value	P value flag.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

validate.pvals *Validate Pvalues*

Description

This function ensures the fold change minimum, maximum, and step are valid.

Usage

```
validate.pvals(p.min, p.max, p.step)
```

Arguments

p.min	The minimum starting FDR cutoff to be checked, so the minimum fold change cutoff to be evaluated will be p.min + p.step, p.min default = 0.
p.max	The maximum FDR cutoff to be checked, default = 0.2.
p.step	The step from the minimum to maximum fold change cutoff, one step increase at a time, default = 0.005.

Details

Specifically it checks that the pvalues are between 0-1, and that at least 1 p.step fits within the p.min and p.max bounds and is positive.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`validate.single.table.isnotlist` *Validate Single Table is not list*

Description

Makes sure the summary table being input is of the right class and format.

Usage

```
validate.single.table.isnotlist(data)
```

Arguments

`data`	summary statistics table (data.frame) from limma or DEseq2, where rownames are gene id.

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

`validate.stats` *Validate Summary Statistics File*

Description

Check for required column names and types.

Usage

```
validate.stats(datin, name = 1, ...)
```

Arguments

`datin`	the summary statistics file.
`name`	summary statistics file position indicator
`...`	pass on variables

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

validate.stats.cols *Check Summary Statistics Required Columns*

Description

Required columns are FCflag and FDRflag

Usage

```
validate.stats.cols(datin, name = 1, req.cols)
```

Arguments

datin	the summary statistics file.
name	summary statistics file position indicator
req.cols	required column names of FCflag and FDRflag pass on from validate.stats

References

Xingpeng Li, Tatiana Gelaf Romer & Siddhartha Pachhai RVA - RNAseq Visualization Automation tool.

wpA2020 *This is data to be included in package*

Description

This is data to be included in package

Usage

```
wpA2020
```

Format

Rwikipathway data downloaded version 2020

name pathway name

version version

wpid pathway id

org host name ...

Index