Package 'daVis'

November 7, 2026

Type Package

Title Visualization Of Differential Expression Analysis

Version 0.99.4

Date 2026-06-15

Description This package contains utility functions to visualize the output from differential expression analysis. The input can be either a model or a list of top tables or a combination of the two. The model can be output from limma, edgeR or DESeq2.

Imports edgeR, ggh4x, ggplot2, grDevices, limma, plyr, stats, utils, DESeq2, rlang, legendry, UpSetR

Suggests AnnotationDbi, Biobase, ggrepel, ggtext, knitr, methods, org.Mm.eg.db, pander, plotly, rmarkdown, S4Vectors, scales, testthat, tools, BiocStyle

URL https://github.com/openanalytics/daVis

BugReports https://github.com/openanalytics/daVis/issues

License GPL-3

biocViews Visualization, DataRepresentation, DifferentialExpression, Software, Microarray, RNASeq

VignetteBuilder knitr

Encoding UTF-8

Config/roxygen2/version 8.0.0

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

git_branch devel

git_last_commit 5f2212f

git_last_commit_date 2026-06-15

Repository Bioconductor 3.24

Date/Publication 2026-11-06

Author Katarzyna Gorczak [aut, cre], Laure Cougnaud [aut]

Maintainer Katarzyna Gorczak <katarzyna.gorczak@openanalytics.eu>

Contents

addHoverText . . . 4
arrangeTopTables . . . 4
calcCorrelation . . . 5
callBarplot . . . 6
callHeatmap . . . 7
callLogRatioPlot . . . 8
callMAplot . . . 9
callScatterPlot . . . 11
callVolcanoPlot . . . 12
callWaterfallPlot . . . 15
cbindFill . . . 16
checkCoef . . . 17
checkColumns . . . 17
checkInput . . . 18
colorBlindPalette . . . 18
concatenateVars . . . 19
createDataBarplot . . . 19
createDataHeatmap . . . 20
createDataLogRatioPlot . . . 21
createDataMAplot . . . 22
createDataScatterPlot . . . 23
createDataUpsetPlot . . . 25
createDataVolcanoPlot . . . 26
createDataWaterfallPlot . . . 27
createExampleData . . . 29
createExpressionSet . . . 29
createPairData . . . 30
createSampleAnnotation . . . 30
createTopTableGenesOfInterest . . . 31
daHeatmapLogFC . . . 32
daLogRatioPlot . . . 34
daMAplot . . . 36
daScatterPlot . . . 39
daSignificantGenesBarplot . . . 42
daUpset . . . 43
daVis-common-args . . . 45
daVis-common-doc . . . 45
daVolcanoPlot . . . 46
daWaterfallPlot . . . 49
downloadData . . . 51
extractColsOfInterest . . . 52
extractFeatures . . . 53
extractPairs . . . 53
extractQueryList . . . 54
extractTopTables . . . 55
extractTTcoef . . . 56
facet . . . 57
filterCommonFeatures . . . 57
filterGenesOfInterest . . . 58
filterLogFC . . . 58

formatAesMA . . . 59
formatAesVP . . . 60
formatAesWP . . . 61
formatManualScale . . . 62
formatTTcoef . . . 62
formatVariableLength . . . 63
formatVariableSpace . . . 64
getCoefColor . . . 64
getCoefLabel . . . 65
getFeatureColor . . . 65
getFeatures . . . 66
getInputIdCoef . . . 66
getModelCoefs . . . 67
getNumberOfRegulatedGenes . . . 67
getNumberOfSignificantGenes . . . 68
getSEModel . . . 68
getTopFeatures . . . 69
getTopGenes . . . 70
getTopTableFromModel . . . 70
ggPlotTheme . . . 71
isModel . . . 72
isTopTable . . . 73
labelCorr . . . 73
labelGenesOfInterest . . . 74
labelTextLRP . . . 75
labelTopGenes . . . 75
mainH . . . 76
mainLRP . . . 77
mainMA . . . 77
mainSP . . . 78
mainVP . . . 79
mainWP . . . 80
makeElementsUnique . . . 81
modelExampleData . . . 81
orderComparison . . . 82
orderFeatures . . . 82
processFeatures . . . 83
reshapeTable . . . 84
runDESeq2 . . . 84
runEdgeR . . . 85
runLimma . . . 85
runTopTable . . . 86
setCategoricalElement . . . 86
setFixElement . . . 87
setGradientScale . . . 87
setManualScale . . . 88
subsetFeatures . . . 88
topFeaturesDeseq . . . 89
topFeaturesEdger . . . 89
topFeaturesLimma . . . 90

Index **91**

addHoverText *Add column with hover text to top table*

Description

Add column with hover text to top table

Usage

```
addHoverText(input, columns)
```

Arguments

input	top table
columns	columns to extract from top table for hover info

Value

top table with additional 'hoverText' column

Author(s)

Katarzyna Gorczak

arrangeTopTables *Arrange top tables*

Description

Arrange top tables

Usage

```
arrangeTopTables(
  input,
  featuresIdVar,
  logFCrange = NULL,
  commonFeatures = FALSE,
  fdr = NULL,
  dir = NULL,
  output = c("table", "list")
)
```

Arguments

input	list of top tables or long table
featuresIdVar	column with feature identifier (must be unique)
logFCrange	numeric vector with two values (low- and high-threshold for log-foldchange)
commonFeatures	logical whether to keep the same set of features per coefficients
fdr	fdr threshold
dir	direction for the logFC ("pos" or "neg"; NULL by default). The features with "pos" diraction will be extracted with logFC above 0. The features with "neg" diraction will be extracted with logFC below 0.
output	whether to output list of top tables or long table

Value

arranged list or long data.frame with top tables for coefficients of interest

Author(s)

Katarzyna Gorczak

---

calcCorrelation *Calculate correlation*

Description

Calculate correlation

Usage

```
calcCorrelation(input)
```

Arguments

input	long-format table

Value

text with information from `cor.test`

Author(s)

Katarzyna Gorczak

`callBarplot` *Create ggplot object with barplot*

Description

Create ggplot object with barplot

Usage

```
callBarplot(
  tbl,
  color,
  annotCex,
  addPercentage,
  title,
  titleCex,
  xlab,
  ylab,
  axesTitleCex,
  axesCex,
  legendPosition,
  legendTitleCex,
  legendCex
)
```

Arguments

`tbl`	combined summaries for all coefficients (i.e.: number of significantly up- and down-regulated genes)
`color`	color palette, must contain two colors.
`annotCex`	cex for the text displayed in the plot (indicating number of genes).
`addPercentage`	logical whether to add percentage
`title`	plot title, NULL by default.
`titleCex`	cex for the plot title.
`xlab`	x-axis title, NULL by default.
`ylab`	y-axis title, NULL by default.
`axesTitleCex`	cex for the axis title.
`axesCex`	cex for the axis text.
`legendPosition`	legend position ("right", "bottom", "none"). If 'none', no legend is shown.
`legendTitleCex`	cex for the legend title.
`legendCex`	cex for the legend text.

Value

ggplot object

Author(s)

Katarzyna Gorczak

`callHeatmap` *Create ggplot object with heatmap*

Description

Create ggplot object with heatmap

Usage

```
callHeatmap(
  topTableOutput,
  xlab,
  ylab,
  title,
  color,
  axesCex,
  featuresColor,
  axesTitleCex,
  titleCex,
  legendTitleCex,
  legendCex,
  typePlot,
  coefColor,
  colorNA
)
```

Arguments

`topTableOutput`	combined topTables for all coefficients
`xlab`	x-axis title, NULL by default.
`ylab`	y-axis title, NULL by default.
`title`	plot title, NULL by default.
`color`	palette for gradient to fill the heatmap.
`axesCex`	cex for the axis text.
`featuresColor`	character vector specifying colors to use for the feature label text. Either length 1 or the same length as `features`, allowing particular features to be highlighted.
`axesTitleCex`	cex for the axis title.
`titleCex`	cex for the plot title.
`legendTitleCex`	cex for the legend title.
`legendCex`	cex for the legend text.
`typePlot`	plot type, can be one of "static" or "interactive".
`coefColor`	color palette for coef labels.
`colorNA`	color for missing values.

Value

ggplot object

Author(s)

Katarzyna Gorczak

`callLogRatioPlot` *Create ggplot object with log ratio plot*

Description

Create ggplot object with log ratio plot

Usage

```
callLogRatioPlot(
  topTableOutput,
  xlab,
  xexpand,
  axesCex,
  axesTitleCex,
  title,
  titleCex,
  facetNCol,
  facetCex,
  facetColor,
  color,
  textVar = NULL,
  textVarCex,
  featuresColor,
  typePlot,
  multiplePlot
)
```

Arguments

`topTableOutput`	combined topTables for all coefficients
`xlab`	x-axis title, or `NULL` to remove. By default, set to: 'logFC' or 'logFC (+- SE)' if error bar(s) are available.
`xexpand`	expansion factor for the x-axis (see `expansion`. If `text` is specified, the x-axis is expanded by 20% on each side.
`axesCex`	relative size for the label of the axes.
`axesTitleCex`	relative size for the title of the axes.
`title`	plot title, NULL by default.
`titleCex`	relative size for the title of the plot.
`facetNCol`	number of columns in facets, by default the function `n2mfrow` is used to.
`facetCex`	cex for the facets if multiple plot are used.
`facetColor`	color for the text of the facets
`color`	string with colors for the bars, one per coefficient.
`textVar`	(optional) String with name of a column to display as text.

textVarCex	cex for the text next to bars
featuresColor	character vector specifying colors to use for the feature label text. Either length 1 or the same length as features, allowing particular features to be highlighted.
typePlot	plot type, can be one of "static" or "interactive".
multiplePlot	whether to use facet_wrap on coefficients

Value

ggplot object

Author(s)

Laure Cougnaud, Kirsten Van Hoorde, Katarzyna Gorczak

---

callMAplot *Create ggplot object with MA plot*

Description

Create ggplot object with MA plot

Usage

```
callMAplot(
  topTableOutput,
  topTableOutputTopGenes,
  includeTableGenesOfInterest,
  title,
  topTableOutputGenesOfInterest,
  xlab,
  ylab,
  axesCex,
  axesTitleCex,
  titleCex,
  facetCex,
  facetColor,
  facetNCol,
  genesToHighlightCex,
  topGenesCex,
  color,
  direction,
  alpha,
  sizeVar,
  size,
  legendPosition,
  legendTitleCex,
  legendCex,
  typePlot,
  multiplePlot,
  ...
)
```

Arguments

topTableOutput	combined topTables for all coefficients
topTableOutputTopGenes	data.frame with top genes
includeTableGenesOfInterest	whether to label genesToHighlight
title	plot title, NULL by default.
topTableOutputGenesOfInterest	data.frame with genesToHighlight
xlab	x-axis title, NULL by default.
ylab	y-axis title, NULL by default.
axesCex	cex for the axis text.
axesTitleCex	cex for the axis title.
titleCex	cex for the plot title.
facetCex	cex for the facets if multiple plot are used.
facetColor	color for the text of the facets
facetNCol	number of columns in facets, by default the function n2mfrow is used to.
genesToHighlightCex	cex for genesToHighlight
topGenesCex	cex for topGenes labels
color	colors for points indicating direction (should be three colors: significant up- and down-regulated and non-significant)
direction	logical whether to color significant up- and down-regulated genes
alpha	transparency level for the points, 0.4 by default.
sizeVar	column name used for the size, empty by default.
size	character or factor with specified size(s) (cex) for the points, replicated if needed. This is used only if sizeVar is empty. By default: '2.5' if sizeVar is not specified and default ggplot size(s) otherwise
legendPosition	legend position ("right", "bottom", "none"). If 'none', no legend is shown.
legendTitleCex	cex for the legend title.
legendCex	cex for the legend text.
typePlot	plot type can be one of "static" or "interactive".
multiplePlot	whether to use facet_wrap on coefficients
...	Extra parameters passed to geom_text_repel to customize the position of the gene labels.

Value

ggplot object

Author(s)

Katarzyna Gorczak

`callScatterPlot` *Create ggplot object with scatter plot*

Description

Create ggplot object with scatter plot

Usage

```
callScatterPlot(
  topTableOutput,
  topTableOutputTopGenes,
  includeTableGenesOfInterest,
  coef,
  topTableOutputGenesOfInterest,
  facetNCol,
  xlab,
  ylab,
  title,
  fdr,
  color,
  axesCex,
  axesTitleCex,
  titleCex,
  legendPosition,
  legendTitleCex,
  legendCex,
  facetCex,
  facetColor,
  topGenesCex,
  genesToHighlightCex,
  alpha,
  pointSize,
  typePlot,
  multiplePlot,
  correlation,
  correlationCex,
  ...
)
```

Arguments

`topTableOutput`	combined topTables for all coefficients
`topTableOutputTopGenes`	data.frame with top genes
`includeTableGenesOfInterest`	whether to label `genesToHighlight`
`coef`	character, coefficient names.
`topTableOutputGenesOfInterest`	data.frame with `genesToHighlight`

facetNCol	number of columns in facets, by default the function n2mfrow is used to.
xlab	x-axis title, NULL by default.
ylab	y-axis title, NULL by default.
title	plot title, NULL by default.
fdr	threshold considered for significance, NULL by default.
color	color palette to distinguish significance groups. Four colors must be specified.
axesCex	cex for the axis text.
axesTitleCex	cex for the axis title.
titleCex	cex for the plot title.
legendPosition	legend position ("right", "bottom", "none"). If 'none', no legend is shown.
legendTitleCex	cex for the legend title.
legendCex	cex for the legend text.
facetCex	cex for the facets if multiple plot are used.
facetColor	color for the text of the facets
topGenesCex	cex for topGenes labels
genesToHighlightCex	cex for genesToHighlight
alpha	transparency level for the points, 0.4 by default.
pointSize	point size, 2 by default.
typePlot	plot type, can be one of "static" or "interactive"
multiplePlot	logical whether to use facet on coefficients
correlation	logical whether to calculate the correlation and add text to the plot.
correlationCex	cex for the text with correlation value
...	Extra parameters passed to geom_text_repel to customize the position of the gene labels.

Value

ggplot object

Author(s)

Katarzyna Gorczak

---

callVolcanoPlot *Create ggplot object with volcano plot*

Description

Create ggplot object with volcano plot

Usage

```
callVolcanoPlot(
  topTableOutput,
  topTableOutputTopGenes,
  includeTableGenesOfInterest,
  topTableOutputGenesOfInterest,
  xlab,
  ylab,
  title,
  fdr,
  colorVar,
  color,
  shapeVar,
  shape,
  alphaVar,
  alpha,
  alphaRange,
  sizeVar,
  size,
  sizeRange,
  axesCex,
  axesTitleCex,
  titleCex,
  legendPosition,
  legendTitleCex,
  legendCex,
  facetCex,
  facetColor,
  facetNCol,
  topGenesCex,
  genesToHighlightCex,
  typePlot,
  multiplePlot,
  ...
)
```

Arguments

topTableOutput	combined topTables for all coefficients
topTableOutputTopGenes	data.frame with top genes
includeTableGenesOfInterest	whether to label genesToHighlight
topTableOutputGenesOfInterest	data.frame with genesToHighlight
xlab	x-axis title, NULL by default.
ylab	y-axis title, NULL by default.
title	plot title, NULL by default.
fdr	threshold considered for significance, NULL by default.
colorVar	(optional) string with column name containing variable used for coloring. If not specified, coloring is based on the adjusted p-values.

color	color palette, only used if colorVar is specified.
shapeVar	(optional) string with column name containing variable used for shaping. If not specified, only points are used.
shape	shape palette, only used if shapeVar is specified.
alphaVar	column name used for the transparency, empty by default.
alpha	character or factor with specified transparency(s) for the points, replicated if needed. By default: '1' if alphaVar is not specified.
alphaRange	transparency (alpha) range used in the plot, possible only if the alphaVar is 'numeric' or 'integer'.
sizeVar	column name used for the size, empty by default.
size	character or factor with specified size(s) (cex) for the points, replicated if needed. This is used only if sizeVar is empty. By default: '2.5' if sizeVar is not specified and default ggplot size(s) otherwise
sizeRange	size (cex) range used in the plot, possible only if the sizeVar is 'numeric' or 'integer'
axesCex	cex for the axis text.
axesTitleCex	cex for the axis title.
titleCex	cex for the plot title.
legendPosition	legend position ("right", "bottom", "none"). If 'none', no legend is shown.
legendTitleCex	cex for the legend title.
legendCex	cex for the legend text.
facetCex	cex for the facets if multiple plot are used.
facetColor	color for the text of the facets
facetNCol	number of columns in facets, by default the function n2mfrow is used to.
topGenesCex	cex for topGenes labels
genesToHighlightCex	cex for genesToHighlight
typePlot	plot type can be one of "static" or "interactive".
multiplePlot	whether to use facet_wrap on coefficients
...	Extra parameters passed to geom_text_repel to customize the position of the gene labels.

Value

ggplot object

Author(s)

Laure Cougnaud, Kirsten Van Hoorde, Katarzyna Gorczak

`callWaterfallPlot` *Create ggplot object with waterfall plot*

Description

Create ggplot object with waterfall plot

Usage

```
callWaterfallPlot(
  topTableOutput,
  featuresColor,
  xlab,
  ylab,
  axesCex,
  axesTitleCex,
  title,
  titleCex,
  facetNCol,
  fillVar,
  fill,
  colorVar,
  color,
  alphaVar,
  alpha,
  alphaRange,
  legendPosition,
  legendCex,
  legendTitleCex,
  facetCex,
  facetColor,
  typePlot,
  multiplePlot
)
```

Arguments

`topTableOutput`	combined top tables for all coefficients
`featuresColor`	character vector specifying colors to use for the feature label text. Either length 1 or the same length as `features`, allowing particular features to be highlighted.
`xlab`	x-axis title, 'logFC' by default.
`ylab`	y-axis title, NULL by default.
`axesCex`	cex for the axis text.
`axesTitleCex`	cex for the axis title.
`title`	plot title, NULL by default.
`titleCex`	cex for the plot title.
`facetNCol`	number of columns in facets, by default the function `n2mfrow` is used to.
`fillVar`	name of variable (in 'input') used for filling, empty by default

fill	character or factor with specified color(s) for the boxplot inside, replicated if needed. By default: 'skyblue2' if fillVar is not specified and default ggplot palette otherwise.
colorVar	name of variable (in 'input') used for coloring, empty by default
color	character or factor with specified color(s) for the bar border, replicated if needed. By default: 'skyblue2' if colorVar is not specified and default ggplot palette otherwise.
alphaVar	column name used for the transparency, empty by default.
alpha	character or factor with specified transparency(s) for the bars, replicated if needed. By default: '1' if alphaVar is not specified.
alphaRange	transparency (alpha) range used in the plot, possible only if the alphaVar is 'numeric' or 'integer'.
legendPosition	legend position ("right", "bottom", "none"). If 'none', no legend is shown.
legendCex	cex for the legend text.
legendTitleCex	cex for the legend title.
facetCex	cex for the facets if multiple plot are used.
facetColor	color for the text of the facets
typePlot	plot type, can be one of "static" or "interactive".
multiplePlot	logical whether to use facet_wrap on coefficients

Value

ggplot object

Author(s)

Katarzyna Gorczak

`cbindFill` *Combine data.frames by columns allowing for different number of rows.*

Description

Combine data.frames by columns allowing for different number of rows.

Usage

```
cbindFill(list, featuresIdVar, sort = FALSE)
```

Arguments

list	list with data frames which should be combined by columns
featuresIdVar	columns with feature ids by which data frames should be combined
sort	logical, whether the rows should be sorted

Value

data.frame

Author(s)

Katarzyna Gorczak

---

checkCoef *Check coefficients*

Description

Check coefficients

Usage

```
checkCoef(input, coef = NULL)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character vector with coefficients

Value

no returned value, error if any of the coefficients not present

---

checkColumns *Check input fields*

Description

Check input fields

Usage

```
checkColumns(input, featuresIdVar, coef, cols, error = TRUE)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
featuresIdVar	column name with unique feature identifiers
coef	coefficient name
cols	columns with feature annotation
error	Logical, if TRUE (by default) an error is returned if required columns are not present

Value

(invisibly) columns not available in the input and error if required columns not present (if `error` is TRUE)

Author(s)

Katarzyna Gorczak

---

checkInput *Check input*

Description

Check input

Usage

```
checkInput(input, error = TRUE)
```

Arguments

`input`	model or a list with top tables named with coefficients. For model: object of class `MArrayLM` (linear model, see `eBayes`), `DGELRT` (see `edgeR`) or `DESeqResults` (see `DESeq2`) are supported. See more details on the top table format in section: Top table format.
`error`	logical whether to return error message

Value

(invisible) logical indicating if `input` passes the check, and error if input is not of the desired class (if `error` is TRUE).

---

colorBlindPalette *Color-blind palette*

Description

Color-blind palette

Usage

```
colorBlindPalette(n, grey = TRUE)
```

Arguments

`n`	number of colors to return
`grey`	whether '#999999' color should be included

Value

n colors

concatenateVars *Concatenate feature variables*

Description

Concatenate feature variables

Usage

```
concatenateVars(tbl, vars, nChar = NULL)
```

Arguments

tbl	long-form top table for all coefficients
vars	column names to concatenate
nChar	maximum number of characters to truncate the feature labels to

Value

character vector with concatenated with ' | ' columns

Author(s)

Katarzyna Gorczak

createDataBarplot *Create table for barplot*

Description

Create table for barplot

Usage

```
createDataBarplot(
  input,
  coef,
  coefLabel = coef,
  fdr,
  logFCrange = NULL,
  addPercentage = FALSE
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character vector with coefficients
coefLabel	character vector with coefficient labels or list with such vector (in case of multiple labels)
fdr	threshold considered for significance, NULL by default.
logFCrange	numeric, two values (upper and lower bounds for logFC).
addPercentage	logical whether to add percentage

Value

data.frame

Author(s)

Katarzyna Gorczak

---

createDataHeatmap *Create table for heatmap*

Description

Create table for heatmap

Usage

```
createDataHeatmap(
  input,
  coef,
  coefLabel = NULL,
  features = NULL,
  featuresIdVar,
  featuresVar,
  featuresMaxNChar,
  typePlot
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character, coefficient names.
coefLabel	character vector of labels or a function to transform the existing labels.
features	IDs of features to show. If NULL (by default), show top 20 features.

featuresIdVar	column name with feature ids.
featuresVar	column name with feature ids to label, the same as featuresIdVar by default.
featuresMaxNChar	maximum number of characters to truncate the feature labels to.
typePlot	plot type, can be one of "static" or "interactive".

Value

data.frame

Author(s)

Laure Cougnaud, Katarzyna Gorczak

---

createDataLogRatioPlot *Create table for log ratio plot*

Description

Create table for log ratio plot

Usage

```
createDataLogRatioPlot(
  input,
  coef,
  coefLabel = NULL,
  text = NULL,
  features = NULL,
  featuresIdVar,
  featuresVar,
  featuresMaxNChar,
  order,
  featuresOrder,
  errorBars,
  typePlot
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character vector with coefficients
coefLabel	character vector with coefficient labels or list with such vector (in case of multiple labels)

text	(optional) String with name of a column, or function to extract a text from the columns of a top table, to display as text. See the available columns in the top table in section: Top table format.
features	character vector with features to subset input. If NULL, top table for the top n features (for the first coefficient) is extracted.
featuresIdVar	column name with unique feature ids (empty by default)
featuresVar	column name with feature ids to label.
featuresMaxNChar	maximum number of characters to truncate the feature labels to.
order	logical whether to order features
featuresOrder	if not NULL, features in the graph are re-ordered, either based on similarity ('similarity') or significance ('significance') of the statistics. See section 'Feature ordering'.
errorBars	logical whether to add standard errors. Allowed only for input of class "MArrayLM" (or list of those).
typePlot	plot type, can be one of "static" or "interactive".

Value

data.frame

`createDataMAplot` *Create table for MA plot*

Description

Create table for MA plot

Usage

```
createDataMAplot(
  input,
  coef,
  coefLabel = NULL,
  featuresIdVar,
  featuresVar,
  topGenes,
  topGenesVar,
  genesToHighlight,
  genesToHighlightVar,
  logFCrange,
  direction,
  sizeVar,
  fdr,
  typePlot
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character vector with coefficients
coefLabel	character vector with coefficient labels or list with such vector (in case of multiple labels)
featuresIdVar	column name with unique feature ids (empty by default)
featuresVar	column name with feature ids to label.
topGenes	numeric, number of top genes with highest logFC or p-value to highlight in the plot for each considered coefficient, 0 by default.
topGenesVar	column name with feature identifier to label topGenes
genesToHighlight	string with identifiers of the genes to highlight, NULL by default. The gene identifiers should correspond to the variable specified in featuresIdVar or row names of input
genesToHighlightVar	column name for the labels of genesToHighlight. Same as featuresIdVar by default.
logFCrange	numeric, two values (upper and lower bounds for logFC).
direction	logical whether to color significant up- and down-regulated genes
sizeVar	column name used for the size, empty by default.
fdr	threshold considered for direction (up- or down- significant features), 0.05 by default.
typePlot	plot type can be one of "static" or "interactive".

Value

data.frame

Author(s)

Katarzyna Gorczak

---

createDataScatterPlot *Create table for the scatter plot*

Description

Create table for the scatter plot

Usage

```
createDataScatterPlot(
  input,
  coef,
  coefLabel = NULL,
  featuresIdVar,
  featuresVar,
  fdr,
  topGenes,
  topGenesVar,
  genesToHighlight,
  genesToHighlightVar,
  genesToHighlightThresholdPValue,
  genesToHighlightThresholdLogFC,
  logFCrange,
  typePlot
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character vector with coefficients
coefLabel	character vector with coefficient labels or list with such vector (in case of multiple labels)
featuresIdVar	column name with unique feature ids (empty by default)
featuresVar	column name with feature ids to label, the same as 'featuresIdVar' by default.
fdr	threshold considered for significance, NULL by default.
topGenes	numeric, number of top genes with highest logFC or p-value to highlight in the plot for each considered coefficient, 0 by default.
topGenesVar	column name with feature identifier to label topGenes
genesToHighlight	string with identifiers of the genes to highlight, NULL by default. The gene identifiers should correspond to the variable specified in genesToHighlightVar, and be contained among the column names of the output of the topTable function from limma.
genesToHighlightVar	column name with the genes to highlight. Same as featuresIdVar by default.
genesToHighlightThresholdPValue	numeric, if specified (1 by default) keep among the genes to highlight, the genes which have a raw p-value lower (strict) than this threshold for at least one of the coefficient considered.
genesToHighlightThresholdLogFC	numeric, if specified (NULL by default) keep among the genes to highlight, the genes which have an absolute log FC higher (strict) than this threshold for at least one of the coefficient considered.
logFCrange	numeric, two values (upper and lower bounds for logFC).
typePlot	plot type, can be one of "static" or "interactive"

Value

data.frame

Author(s)

Katarzyna Gorczak

`createDataUpsetPlot` *Create data for upset plot*

Description

Create data for upset plot

Usage

```
createDataUpsetPlot(
  input,
  coef,
  coefLabel = NULL,
  featuresIdVar = character(),
  fdr,
  dir
)
```

Arguments

`input`	model or a list with top tables named with coefficients. For model: object of class `MArrayLM` (linear model, see `eBayes`), `DGELRT` (see `edgeR`) or `DESeqResults` (see `DESeq2`) are supported. See more details on the top table format in section: Top table format.
`coef`	character vector with coefficients
`coefLabel`	character vector with coefficient labels or list with such vector (in case of multiple labels)
`featuresIdVar`	column name with unique feature ids (empty by default)
`fdr`	threshold considered for significance, 0.05 by default.
`dir`	direction for feature regulation ('up' to select up-regulated features; 'down' to select down-regulated features)

Value

data.frame compatible with UpSetR

Author(s)

Kirsten Van Hoorde, Katarzyna Gorczak

createDataVolcanoPlot *Create table for the volcano plot*

Description

Create table for the volcano plot

Usage

```
createDataVolcanoPlot(
  input,
  coef,
  coefLabel = NULL,
  featuresIdVar,
  featuresVar,
  colorVar,
  shapeVar,
  alphaVar,
  sizeVar,
  fdr,
  additionalThresholdsAdjPValue,
  topGenes,
  topGenesVar,
  genesToHighlight,
  genesToHighlightVar,
  genesToHighlightThresholdPValue,
  genesToHighlightThresholdLogFC,
  logFCrange,
  typePlot
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character vector with coefficients
coefLabel	character vector with coefficient labels or list with such vector (in case of multiple labels)
featuresIdVar	column name with unique feature ids (empty by default)
featuresVar	column name with feature ids to label; used for extra labels if typePlot is 'interactive'.
colorVar	(optional) string with column name containing variable used for coloring. If not specified, coloring is based on the adjusted p-values.
shapeVar	(optional) string with column name containing variable used for shaping. If not specified, only points are used.
alphaVar	column name used for the transparency, empty by default.
sizeVar	column name used for the size, empty by default.

fdr	threshold considered for significance, NULL by default.
additionalThresholdsAdjPValue	numeric, additional adjusted p-values thresholds to use for the coloring of the points, and indicated in the legend.
topGenes	numeric, number of top genes with highest logFC or p-value to highlight in the plot for each considered coefficient, 0 by default.
topGenesVar	column name with feature identifier to label topGenes
genesToHighlight	string with identifiers of the genes to highlight, NULL by default. The gene identifiers should correspond to the variable specified in genesToHighlightVar, and be contained among the column names of the output of the topTable function from limma.
genesToHighlightVar	column name with the genes to highlight. Same as featuresIdVar by default.
genesToHighlightThresholdPValue	numeric, if specified (1 by default) keep among the genes to highlight, the genes which have a raw p-value lower (strict) than this threshold for at least one of the coefficient considered.
genesToHighlightThresholdLogFC	numeric, if specified (NULL by default) keep among the genes to highlight, the genes which have an absolute log FC higher (strict) than this threshold for at least one of the coefficient considered.
logFCrange	numeric, two values (upper and lower bounds for logFC).
typePlot	plot type can be one of "static" or "interactive".

Value

data.frame

Author(s)

Laure Cougnaud, Kirsten Van Hoorde, Katarzyna Gorczak, Michela Pasetto

---

`createDataWaterfallPlot` *Create data for waterfall plot*

Description

Create data for waterfall plot

Usage

```
createDataWaterfallPlot(
  input,
  coef,
  coefLabel = NULL,
  features = NULL,
  featuresIdVar,
```

```
  featuresVar,
  featuresMaxNChar,
  fillVar,
  colorVar,
  alphaVar,
  fdr,
  logFCrange,
  typePlot
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character vector with coefficients
coefLabel	character vector with coefficient labels or list with such vector (in case of multiple labels)
features	character vector with features to subset input. If NULL, top table for the top n features (for the first coefficient) is extracted.
featuresIdVar	column name with unique feature ids (empty by default)
featuresVar	column name with feature ids to label, the same as 'featuresIdVar' by default.
featuresMaxNChar	numeric, maximum number of characters to truncate the feature labels to.
fillVar	name of variable (in 'input') used for filling, empty by default
colorVar	name of variable (in 'input') used for coloring, empty by default
alphaVar	column name used for the transparency, empty by default.
fdr	threshold considered for significance, NULL by default.
logFCrange	numeric, two values (upper and lower bounds for logFC).
typePlot	plot type, can be one of "static" or "interactive".

Value

data.frame

Author(s)

Katarzyna Gorczak

createExampleData *Create example limma-, edgeR-object or a list of top tables.*

Description

Create example limma-, edgeR-object or a list of top tables.

Usage

```
createExampleData(
  path = ".",
  output = c("limma", "topTable", "edgeR", "deseq2"),
  quiet = TRUE
)
```

Arguments

path	absolute path where data should be downloaded to
output	which object should be created. Possible options: limma, edgeR, topTableList, DESeq2
quiet	if TRUE, suppress status messages (if any), and the progress bar (see download.file function)

Value

list with specific objects (output from limma, edgeR, topTable and/or DESeq2)

Examples

```
tmpDir <- tempfile(); dir.create(tmpDir)
getData <- createExampleData(path = tmpDir, output = "limma")
```

createExpressionSet *Create ExpressionSet object from example data*

Description

Create ExpressionSet object from example data

Usage

```
createExpressionSet(path)
```

Arguments

path	absolute path where data should be downloaded to

Value

ExpressionSet object

Author(s)

Katarzyna Gorczak

`createPairData` *Create pairwise data*

Description

Create pairwise data

Usage

```
createPairData(tbl, featuresIdVar, fdr, typePlot, columns)
```

Arguments

`tbl`	list of top tables
`featuresIdVar`	column name with feature ids, empty by default. If specified and input is a model, featuresIdVar should be a column name in 'genes' slot.
`fdr`	threshold considered for significance, NULL by default.
`typePlot`	plot type, can be one of "static" or "interactive"
`columns`	columns of interest from top table

Value

data.frame with reference and compared contrasts

Author(s)

Katarzyna Gorczak

`createSampleAnnotation` *Create sample annotation from example data*

Description

Create sample annotation from example data

Usage

```
createSampleAnnotation(path)
```

Arguments

path	absolute path where data should be downloaded to

Value

data.frame with sample annotation

Author(s)

Katarzyna Gorczak

---

createTopTableGenesOfInterest

Create table for genes of interest

Description

Create table for genes of interest

Usage

```
createTopTableGenesOfInterest(
  input,
  featuresIdVar = character(),
  genesToHighlight,
  genesToHighlightVar,
  genesToHighlightThresholdPValue = 1,
  genesToHighlightThresholdLogFC = NULL
)
```

Arguments

input	long-format table with top tables for coefficients of interest
featuresIdVar	column name with feature ids, 'ENTREZID' by default.
genesToHighlight	string with identifiers of the genes to highlight. The gene identifiers should correspond to the variable specified in genesToHighlightVar.
genesToHighlightVar	column of the data to which the genes to highlight should be map
genesToHighlightThresholdPValue	numeric, if specified (1 by default) keep among the genes to highlight, the genes which have a raw p-value lower (strict) than this threshold for at least one of the coefficient considered
genesToHighlightThresholdLogFC	numeric, if specified (NULL by default) keep among the genes to highlight, the genes which have an absolute log FC higher (strict) than this threshold for at least one of the coefficient considered

Value

long-format table

Author(s)

Laure Cougnaud, Kirsten Van Hoorde

`daHeatmapLogFC` *Heatmap of log fold-changes*

Description

This is a function to create a heatmap that represents logFC values for several coefficients.

Usage

```
daHeatmapLogFC(
  input,
  coef = NULL,
  coefLabel = NULL,
  features = NULL,
  featuresIdVar = character(),
  featuresVar = featuresIdVar,
  featuresColor = "black",
  featuresMaxNChar = 50,
  xlab = NULL,
  ylab = NULL,
  axesCex = 1,
  axesTitleCex = 1,
  title = NULL,
  titleCex = 1,
  legendTitleCex = 1,
  legendCex = 0.8,
  coefColor = "black",
  color = c("#0072B2", "white", "#D55E00"),
  colorNA = "grey",
  typePlot = c("static", "interactive")
)
```

Arguments

`input`	model or a list with top tables named with coefficients. For model: object of class `MArrayLM` (linear model, see `eBayes`), `DGELRT` (see `edgeR`) or `DESeqResults` (see `DESeq2`) are supported. See more details on the top table format in section: Top table format.
`coef`	character, coefficient names.
`coefLabel`	character vector of labels or a function to transform the existing labels.
`features`	IDs of features to show. If NULL (by default), show top 20 features.
`featuresIdVar`	column name with feature ids.
`featuresVar`	column name with feature ids to label, the same as `featuresIdVar` by default.
`featuresColor`	character vector specifying colors to use for the feature label text. Either length 1 or the same length as `features`, allowing particular features to be highlighted.

featuresMaxNChar	maximum number of characters to truncate the feature labels to.
xlab	x-axis title, NULL by default.
ylab	y-axis title, NULL by default.
axesCex	cex for the axis text.
axesTitleCex	cex for the axis title.
title	plot title, NULL by default.
titleCex	cex for the plot title.
legendTitleCex	cex for the legend title.
legendCex	cex for the legend text.
coefColor	color palette for coef labels.
color	palette for gradient to fill the heatmap.
colorNA	color for missing values.
typePlot	plot type, can be one of "static" or "interactive".

Value

ggplot object

Author(s)

Kirsten Van Hoorde, Laure Cougnaud and Katarzyna Gorczak

Examples

```
exampleData <- createExampleData(path = ".", output = c("limma", "topTable"))
model <- exampleData$limma
topTableList <- exampleData$topTable
coefs <- c("B.LvsP", "L.LvsP", "B.PvsV", "L.PvsV")

# Simple heatmap
daHeatmapLogFC(input = model, coef = coefs)

# Specify feature annotation
daHeatmapLogFC(input = model, coef = coefs, featuresIdVar = "ENTREZID",
featuresVar = c("SYMBOL", "GENENAME"), featuresMaxNChar = 35)

# Color coefficient labels
daHeatmapLogFC(input = model, coef = coefs,
coefLabel = c("A", "B", "C", "D"),
coefColor = c("blue", "red", "blue", "red"))

# Specify different set of features
coefs <- c("B.LvsP", "L.LvsP", "B.PvsV", "L.PvsV", "A")
daHeatmapLogFC(
input = list(model, A = topTableList[["B.LvsP"]][c(seq_len(6), 9, 10), ]),
coef = coefs)

# see vignette for other examples
```

daLogRatioPlot *Plot log ratios*

Description

This is a function to create barplot with log-foldchanges.

Usage

```
daLogRatioPlot(
  input,
  coef = NULL,
  coefLabel = NULL,
  features = NULL,
  featuresIdVar = character(),
  featuresVar = featuresIdVar,
  featuresOrder = NULL,
  featuresMaxNChar = 50,
  featuresColor = "black",
  color = character(),
  text = NULL,
  textCex = 4,
  xlab,
  xexpand = if (!is.null(text)) {
     ggplot2::expansion(mult = 0.2)
 },
  axesCex = 1,
  axesTitleCex = 1,
  title = NULL,
  titleCex = 1,
  facetCex = 1,
  facetColor = "black",
  facetNCol = NULL,
  errorBars = TRUE,
  typePlot = c("static", "interactive")
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character, coefficient names.
coefLabel	character vector of labels or function to transform the existing labels, or list with such labels. If a list is specified, nested facets are used. In that case, the coefficients are ordered based on the groups defined by the coefficient labels.
features	feature ids or numeric. If NULL (by default), show top 20 features according to the first coef. If numeric and input is a list, the tables are subsetted according to the row names of the table for the first coef.

featuresIdVar	column name with feature ids.
featuresVar	column name with feature ids to label.
featuresOrder	if not NULL, features in the graph are re-ordered, either based on similarity ('similarity') or significance ('significance') of the statistics. See section 'Feature ordering'.
featuresMaxNChar	maximum number of characters to truncate the feature labels to.
featuresColor	character vector specifying colors to use for the feature label text. Either length 1 or the same length as features, allowing particular features to be highlighted.
color	string with colors for the bars, one per coefficient.
text	(optional) String with name of a column, or function to extract a text from the columns of a top table, to display as text. See the available columns in the top table in section: Top table format.
textCex	cex for the text next to bars
xlab	x-axis title, or NULL to remove. By default, set to: 'logFC' or 'logFC (+- SE)' if error bar(s) are available.
xexpand	expansion factor for the x-axis (see expansion. If text is specified, the x-axis is expanded by 20% on each side.
axesCex	relative size for the label of the axes.
axesTitleCex	relative size for the title of the axes.
title	plot title, NULL by default.
titleCex	relative size for the title of the plot.
facetCex	cex for the facets if multiple plot are used.
facetColor	color for the text of the facets
facetNCol	number of columns in facets, by default the function n2mfrow is used to.
errorBars	logical whether to add error bars to the plot (+/- one standard error). If input (or each element within a input list) is a: • model: only supported for MArrayLM • top table: a 'se' column should be available
typePlot	plot type, can be one of "static" or "interactive".

Value

ggplot object

Feature ordering

The features are ordered based on:

- 'similarity': a hierarchical clustering of the (Euclidean) distances between the statistics. The statistics are for:
  - limma: the t-statistic of each coefficient
  - edgeR: the (overall) F-statistic is considered
- 'significance': decreasing average (-log10) p-values across coefficients. The p-values are for:
  - limma: the significance of each coefficient
  - edgeR: the (overall) p-value of the model

Author(s)

Laure Cougnaud, Katarzyna Gorczak, Heather Turner, Aditya Bhagwat, Kirsten Van Hoorde

Examples

```
exampleData <- createExampleData(path = ".", output = c("limma", "topTable"))
model <- exampleData$limma
topTableList <- exampleData$topTable
coefs <- c("B.LvsP", "L.LvsP", "B.PvsV", "L.PvsV")

# Simple log-ratio plot
daLogRatioPlot(input = model, coef = coefs, facetNCol = 4)

# Specify features and annotation
features <- daVis:::getTopFeatures(input = model, coef = "B.LvsP",
featuresIdVar = "ENTREZID", n = 20)
daLogRatioPlot(input = model, coef = coefs,
features = features, featuresIdVar = "ENTREZID",
featuresVar = c("SYMBOL", "GENENAME"), featuresMaxNChar = 35, facetNCol = 4)

# Specify different set of features
coefs <- c("B.LvsP", "L.LvsP", "B.PvsV", "L.PvsV", "A")
set.seed(123)
features <- sample(features, 20)
daLogRatioPlot(
input = list(model, A = topTableList[["B.LvsP"]][c(seq_len(6), 9, 10), ]),
featuresIdVar = "ENTREZID", features = features, coef = coefs,
facetNCol = 5, errorBars = TRUE)

# Sort features as specified
coefs <- c("B.LvsP", "L.LvsP", "B.PvsV", "L.PvsV")
daLogRatioPlot(input = model, featuresIdVar = "ENTREZID",
features = features, coef = coefs, facetNCol = 4, errorBars = TRUE)

# Sort features based on similarity
daLogRatioPlot(input = model, featuresIdVar = "ENTREZID",
features = features, coef = coefs, facetNCol = 4, errorBars = TRUE,
featuresOrder = "similarity")

# see vignette for other examples
```

`daMAplot` *MA plot*

Description

This is a function to create a MA plot. When several coefficients are used, multiple plots side by side are returned. MAplot visualizes mean expression versus logFC for each gene.

Usage

```
daMAplot(
  input,
```

```
  coef = NULL,
  coefLabel = NULL,
  featuresIdVar = character(),
  featuresVar = featuresIdVar,
  logFCrange = NULL,
  xlab = "log2 mean expression",
  ylab = "logFC",
  axesCex = 1,
  axesTitleCex = 1,
  title = NULL,
  titleCex = 1,
  legendPosition = c("right", "bottom", "none"),
  legendTitleCex = 1,
  legendCex = 0.8,
  facetCex = 1.1,
  facetColor = "black",
  facetNCol = grDevices::n2mfrow(length(coef))[2],
  topGenes = 0,
  topGenesVar = featuresIdVar,
  topGenesCex = 2.5,
  returnTopGenes = FALSE,
  genesToHighlight = NULL,
  genesToHighlightVar = featuresIdVar,
  genesToHighlightCex = 2.5,
  direction = FALSE,
  fdr = 0.05,
  color = if (direction) character() else "grey",
  alpha = 0.5,
  sizeVar = character(),
  size = if (length(sizeVar) > 0) numeric() else 2,
  typePlot = c("static", "interactive"),
  ...
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character, coefficient names.
coefLabel	character vector of labels or a function to transform existing labels.
featuresIdVar	column name with feature ids.
featuresVar	column name with feature ids to label.
logFCrange	numeric, two values (upper and lower bounds for logFC).
xlab	x-axis title, NULL by default.
ylab	y-axis title, NULL by default.
axesCex	cex for the axis text.
axesTitleCex	cex for the axis title.
title	plot title, NULL by default.

titleCex	cex for the plot title.
legendPosition	legend position ("right", "bottom", "none"). If 'none', no legend is shown.
legendTitleCex	cex for the legend title.
legendCex	cex for the legend text.
facetCex	cex for the facets if multiple plot are used.
facetColor	color for the text of the facets
facetNCol	number of columns in facets, by default the function n2mfrow is used to.
topGenes	numeric, number of top genes with highest logFC or p-value to highlight in the plot for each considered coefficient, 0 by default.
topGenesVar	column name with feature identifier to label topGenes
topGenesCex	cex for topGenes labels
returnTopGenes	logical, if TRUE (FALSE by default), return a list with the top genes highlighted in the plot.
genesToHighlight	string with identifiers of the genes to highlight, NULL by default. The gene identifiers should correspond to the variable specified in featuresIdVar or row names of input
genesToHighlightVar	column name for the labels of genesToHighlight. Same as featuresIdVar by default.
genesToHighlightCex	cex for genesToHighlight
direction	logical whether to color significant up- and down-regulated genes
fdr	threshold considered for direction (up- or down- significant features), 0.05 by default.
color	colors for points indicating direction (should be three colors: significant up- and down-regulated and non-significant)
alpha	transparency level for the points, 0.4 by default.
sizeVar	column name used for the size, empty by default.
size	character or factor with specified size(s) (cex) for the points, replicated if needed. This is used only if sizeVar is empty. By default: '2.5' if sizeVar is not specified and default ggplot size(s) otherwise
typePlot	plot type can be one of "static" or "interactive".
...	Extra parameters passed to geom_text_repel to customize the position of the gene labels.

Value

ggplot object or a list with ggplot object and top genes highlighted in the MA plot (top 10 genes with highest significance and/or highest logFC) featuresVar with names featuresIdVar (if returnTopGenes is set to TRUE)

Author(s)

Katarzyna Gorczak

Examples

```
exampleData <- createExampleData(path = ".", output = "limma")
model <- exampleData$limma
coefs <- c("B.LvsP", "L.LvsP")

# Simple MA plot
daMAplot(input = model, coef = coefs[1])

# Color by significant direction
daMAplot(input = model, coef = coefs, coefLabel = c("A", "B"),
direction = TRUE, color = c("steelblue", "firebrick", "grey"), facetNCol = 2)

# see vignette for other examples
```

---

daScatterPlot *Scatter plot*

Description

This is a function to create a scatter plot. When several coefficients are used, multiple plots side by side are returned.

Usage

```
daScatterPlot(
  input,
  coef = NULL,
  coefLabel = coef,
  featuresIdVar = character(),
  featuresVar = featuresIdVar,
  fdr = 0.05,
  logFCrange = NULL,
  xlab = NULL,
  ylab = NULL,
  axesCex = 1,
  axesTitleCex = 1,
  title = NULL,
  titleCex = 1,
  legendPosition = c("right", "bottom", "none"),
  legendTitleCex = 1,
  legendCex = 0.8,
  facetCex = 1,
  facetColor = "black",
  facetNCol = grDevices::n2mfrow(length(coef))[2],
  color = c("gray90", "darkgoldenrod1", "dodgerblue", "darkgreen"),
  topGenes = 0,
  topGenesVar = featuresIdVar,
  topGenesCex = 2.5,
  returnTopGenes = FALSE,
  genesToHighlight = NULL,
```

```
  genesToHighlightVar = featuresIdVar,
  genesToHighlightCex = 2.5,
  genesToHighlightThresholdPValue = 1,
  genesToHighlightThresholdLogFC = NULL,
  alpha = 0.4,
  pointSize = 2,
  correlation = FALSE,
  correlationCex = 3,
  typePlot = c("static", "interactive"),
  ...
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character, coefficient names.
coefLabel	character vector of labels or a function to transform existing labels.
featuresIdVar	column name with feature ids, empty by default. If specified and input is a model, featuresIdVar should be a column name in 'genes' slot.
featuresVar	column name with feature ids to label, the same as 'featuresIdVar' by default.
fdr	threshold considered for significance, NULL by default.
logFCrange	numeric, two values (upper and lower bounds for logFC).
xlab	x-axis title, NULL by default.
ylab	y-axis title, NULL by default.
axesCex	cex for the axis text.
axesTitleCex	cex for the axis title.
title	plot title, NULL by default.
titleCex	cex for the plot title.
legendPosition	legend position ("right", "bottom", "none"). If 'none', no legend is shown.
legendTitleCex	cex for the legend title.
legendCex	cex for the legend text.
facetCex	cex for the facets if multiple plot are used.
facetColor	color for the text of the facets
facetNCol	number of columns in facets, by default the function n2mfrow is used to.
color	color palette to distinguish significance groups. Four colors must be specified.
topGenes	numeric, number of top genes with highest logFC or p-value to highlight in the plot for each considered coefficient, 0 by default.
topGenesVar	column name with feature identifier to label topGenes
topGenesCex	cex for topGenes labels
returnTopGenes	logical, if TRUE (FALSE by default), return a list with the top genes highlighted in the plot.

genesToHighlight
: string with identifiers of the genes to highlight, NULL by default. The gene identifiers should correspond to the variable specified in genesToHighlightVar, and be contained among the column names of the output of the topTable function from limma.

genesToHighlightVar
: column name with the genes to highlight. Same as featuresIdVar by default.

genesToHighlightCex
: cex for genesToHighlight

genesToHighlightThresholdPValue
: numeric, if specified (1 by default) keep among the genes to highlight, the genes which have a raw p-value lower (strict) than this threshold for at least one of the coefficient considered.

genesToHighlightThresholdLogFC
: numeric, if specified (NULL by default) keep among the genes to highlight, the genes which have an absolute log FC higher (strict) than this threshold for at least one of the coefficient considered.

alpha
: transparency level for the points, 0.4 by default.

pointSize
: point size, 2 by default.

correlation
: logical whether to calculate the correlation and add text to the plot.

correlationCex
: cex for the text with correlation value

typePlot
: plot type, can be one of "static" or "interactive"

...
: Extra parameters passed to geom_text_repel to customize the position of the gene labels.

Value

ggplot object or a list with ggplot object and top genes highlighted in the scatter plot (top 10 genes with highest significance and/or highest logFC) featuresVar with names featuresIdVar (if returnTopGenes is set to TRUE)

Author(s)

Katarzyna Gorczak

Examples

```
exampleData <- createExampleData(path = ".", output = "limma")
model <- exampleData$limma
coefs <- c("B.LvsP", "L.LvsP", "B.PvsV", "L.PvsV")

# Simple scatter plot
daScatterPlot(input = model, coef = coefs[c(1,2)])

# More coefficients
daScatterPlot(input = model, coef = coefs, coefLabel = c("A", "B", "C", "D"),
facetNCol = 3)

# LogFC range
daScatterPlot(input = model, coef = coefs, coefLabel = c("A", "B", "C", "D"),
facetNCol = 3, logFCrange = c(-2, 2))

# see vignette for other examples
```

daSignificantGenesBarplot

Barplot with the number of significant genes

Description

This is a function to create a barplot indicating the number of significant genes in each coefficient. The number of up- and down-regulated genes is shown.

Usage

```
daSignificantGenesBarplot(
  input,
  coef,
  coefLabel = coef,
  fdr = 0.05,
  logFCrange = NULL,
  xlab = NULL,
  ylab = NULL,
  axesCex = 1,
  axesTitleCex = 1.1,
  title = NULL,
  titleCex = 1.1,
  legendPosition = c("right", "bottom", "none"),
  legendTitleCex = 1,
  legendCex = 0.8,
  color = c("#32a6d3", "#e52323"),
  annotCex = 3.5,
  addPercentage = FALSE
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character, coefficient names.
coefLabel	character vector of labels or a function to transform the existing labels.
fdr	threshold considered for significance, NULL by default.
logFCrange	numeric, two values (upper and lower bounds for logFC).
xlab	x-axis title, NULL by default.
ylab	y-axis title, NULL by default.
axesCex	cex for the axis text.
axesTitleCex	cex for the axis title.
title	plot title, NULL by default.
titleCex	cex for the plot title.
legendPosition	legend position ("right", "bottom", "none"). If 'none', no legend is shown.

legendTitleCex	cex for the legend title.
legendCex	cex for the legend text.
color	color palette, must contain two colors.
annotCex	cex for the text displayed in the plot (indicating number of genes).
addPercentage	logical whether to add percentage

Value

ggplot object

Author(s)

Katarzyna Gorczak

Examples

```
exampleData <- createExampleData(path = ".", output = "limma")
model <- exampleData$limma
coefs <- c("B.LvsP", "L.LvsP", "B.PvsV", "L.PvsV")

# Simple barplot
daSignificantGenesBarplot(input = model, coef = coefs)

# Add percentage of genes
daSignificantGenesBarplot(input = model, coef = coefs,
coefLabel = c("A", "B", "C", "D"), addPercentage = TRUE)

# see vignette for other examples
```

---

daUpset *Upset plot for up- or down-regulated genes*

Description

This is a function to create a customized upset plot for up- or down-regulated genes.

Usage

```
daUpset(
  input,
  coef = NULL,
  coefLabel = coef,
  featuresIdVar = character(),
  fdr = 0.05,
  dir = c("up", "down"),
  ylab = "Intersection Size",
  xlab = "Set Size",
  axesCex = c(1.2, 1.7),
  axesTitleCex = c(1.4, 1.9),
  barsCex = 1.5,
  returnAnalysis = FALSE
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character, coefficient names.
coefLabel	character vector of labels or a function to transform existing labels.
featuresIdVar	column name with unique feature ids, empty by default.
fdr	threshold considered for significance, 0.05 by default.
dir	direction for feature regulation ('up' to select up-regulated features; 'down' to select down-regulated features)
ylab	y-axis title
xlab	x-axis title
axesCex	cex for the axis text. If two numbers provided, the first one is used for the x-axis and the second one for y-axis.
axesTitleCex	cex for the axis title. If two numbers provided, the first one is used for the x-axis and the second one for y-axis.
barsCex	cex for the counts above the bars.
returnAnalysis	logical, if TRUE (FALSE by default), return also the output of the analysis (list with all overlapping sets based on featuresIdVar), otherwise only the plot object

Value

uspet plot; if returnAnalysis is TRUE, return a list with overlapping sets and plot

Author(s)

Kirsten Van Hoorde, Katarzyna Gorczak, Michela Pasetto

Examples

```
exampleData <- createExampleData(path = ".", output = "limma")
model <- exampleData$limma
coefs <- c("B.LvsP", "L.LvsP", "B.PvsV", "L.PvsV")

# Significantly up-regulated genes
daUpset(input = model, coef = coefs, fdr = 0.05, dir = "up")

# see vignette for other examples
```

daVis-common-args *Common parameters for the functions of the* daVis *package.*

Description

Common parameters for the functions of the daVis package.

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.

Value

No returned value

daVis-common-doc *Common documentation for the* daVis *package.*

Description

Common documentation for the daVis package.

Value

No returned value

Top table format

The top table extracted from a specified model, or specified by the user contains at least the columns:

- 'logFC': log fold change
- 'AveExpr': average expression
- 'P.Value' and 'adj.PVal': raw and (multiplicity correction) adjusted p-values
  - For edgeR model: this is extracted from the columns: 'PValue' and 'FDR' respectively.
- feature identifier as specified by: featuresIdVar

daVolcanoPlot *Volcano plot*

Description

This is a function to create a volcano plot. When several coefficients are used, multiple plots side by side are returned.

Usage

```
daVolcanoPlot(
  input,
  coef = NULL,
  coefLabel = NULL,
  featuresIdVar = character(),
  featuresVar = featuresIdVar,
  fdr = 0.05,
  logFCrange = NULL,
  xlab = "logFC",
  ylab = "-log10(p-value)",
  axesCex = 1,
  axesTitleCex = 1.1,
  title = NULL,
  titleCex = 1.1,
  legendPosition = c("right", "bottom", "none"),
  legendTitleCex = 1,
  legendCex = 0.8,
  facetCex = 1,
  facetColor = "black",
  facetNCol = grDevices::n2mfrow(length(coef))[2],
  topGenes = 0,
  topGenesVar = featuresIdVar,
  topGenesCex = 2.5,
  returnTopGenes = FALSE,
  genesToHighlight = NULL,
  genesToHighlightVar = featuresIdVar,
  genesToHighlightCex = 2.5,
  genesToHighlightThresholdPValue = 1,
  genesToHighlightThresholdLogFC = NULL,
  colorVar = character(),
  color = if (length(colorVar) > 0) character() else "black",
  shapeVar = character(),
  shape = if (length(shapeVar) > 0) numeric() else 19,
  alphaVar = character(),
  alpha = if (length(alphaVar) > 0) numeric() else 0.4,
  alphaRange = numeric(),
  sizeVar = character(),
  size = if (length(sizeVar) > 0) numeric() else 2,
  sizeRange = numeric(),
  additionalThresholdsAdjPValue = NULL,
  typePlot = c("static", "interactive"),
```

```
  ...
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character, coefficient names.
coefLabel	character vector of labels or a function to transform existing labels.
featuresIdVar	column name with feature ids (should be unique). If not specified, row names are used. featuresIdVar is used to label topGenes unless topGenesVar is specified.
featuresVar	column name with feature ids to label; used for extra labels if typePlot is 'interactive'.
fdr	threshold considered for significance, NULL by default.
logFCrange	numeric, two values (upper and lower bounds for logFC).
xlab	x-axis title, NULL by default.
ylab	y-axis title, NULL by default.
axesCex	cex for the axis text.
axesTitleCex	cex for the axis title.
title	plot title, NULL by default.
titleCex	cex for the plot title.
legendPosition	legend position ("right", "bottom", "none"). If 'none', no legend is shown.
legendTitleCex	cex for the legend title.
legendCex	cex for the legend text.
facetCex	cex for the facets if multiple plot are used.
facetColor	color for the text of the facets
facetNCol	number of columns in facets, by default the function n2mfrow is used to.
topGenes	numeric, number of top genes with highest logFC or p-value to highlight in the plot for each considered coefficient, 0 by default.
topGenesVar	column name with feature identifier to label topGenes
topGenesCex	cex for topGenes labels
returnTopGenes	logical, if TRUE (FALSE by default), return a list with the top genes highlighted in the plot.
genesToHighlight	string with identifiers of the genes to highlight, NULL by default. The gene identifiers should correspond to the variable specified in genesToHighlightVar, and be contained among the column names of the output of the topTable function from limma.
genesToHighlightVar	column name with the genes to highlight. Same as featuresIdVar by default.
genesToHighlightCex	cex for genesToHighlight

genesToHighlightThresholdPValue
: numeric, if specified (1 by default) keep among the genes to highlight, the genes which have a raw p-value lower (strict) than this threshold for at least one of the coefficient considered.

genesToHighlightThresholdLogFC
: numeric, if specified (NULL by default) keep among the genes to highlight, the genes which have an absolute log FC higher (strict) than this threshold for at least one of the coefficient considered.

colorVar
: (optional) string with column name containing variable used for coloring. If not specified, coloring is based on the adjusted p-values.

color
: color palette, only used if colorVar is specified.

shapeVar
: (optional) string with column name containing variable used for shaping. If not specified, only points are used.

shape
: shape palette, only used if shapeVar is specified.

alphaVar
: column name used for the transparency, empty by default.

alpha
: character or factor with specified transparency(s) for the points, replicated if needed. By default: '1' if alphaVar is not specified.

alphaRange
: transparency (alpha) range used in the plot, possible only if the alphaVar is 'numeric' or 'integer'.

sizeVar
: column name used for the size, empty by default.

size
: character or factor with specified size(s) (cex) for the points, replicated if needed. This is used only if sizeVar is empty. By default: '2.5' if sizeVar is not specified and default ggplot size(s) otherwise

sizeRange
: size (cex) range used in the plot, possible only if the sizeVar is 'numeric' or 'integer'

additionalThresholdsAdjPValue
: numeric, additional adjusted p-values thresholds to use for the coloring of the points, and indicated in the legend.

typePlot
: plot type can be one of "static" or "interactive".

...
: Extra parameters passed to geom_text_repel to customize the position of the gene labels.

Value

ggplot object or list with ggplot object and top genes highlighted in the volcano plot (top 10 genes with highest significance and/or highest logFC) topGenesVar with names featuresIdVar (if returnTopGenes is set to TRUE)

Author(s)

Laure Cougnaud, Kirsten Van Hoorde, Katarzyna Gorczak

Examples

```
exampleData <- createExampleData(path = ".", output = "limma")
model <- exampleData$limma

# Simple volcano plot
daVolcanoPlot(input = model, coef = "B.LvsP")
```

```
# Specify logFC range
daVolcanoPlot(input = model, coef = "B.LvsP", logFCrange = c(-2, 2))

# Customized aesthetics
daVolcanoPlot(input = model, coef = c("B.LvsP", "L.LvsP"),
coefLabel = c("A", "B"), facetNCol = 2, colorVar = "adj.P.Val")

# Customized gene annotation
model$genes$group <- rep(c("gr A", "gr B", "gr C"), each = 5323)
daVolcanoPlot(input = model, coef = "B.LvsP", colorVar = "group",
color = setNames(c("orange", "red", "blue"), c("gr A", "gr B", "gr C")))

# Facet by variable(s)
coefs <- c("B.LvsP", "L.LvsP", "B.PvsV", "L.PvsV")
coefsLabel <- list(
sub("(.+)\\.(.+)", "\\2", coefs),
sub("(.+)\\.(.+)", "\\1", coefs)
)
daVolcanoPlot(input = model, coef = coefs, coefLabel = coefsLabel,
facetNCol = 4, colorVar = "adj.P.Val")

# see vignette for other examples
```

---

daWaterfallPlot *Waterfall plot*

Description

This is a function to create a waterfall plot. When several coefficients are used, multiple plots side by side are returned.

Usage

```
daWaterfallPlot(
  input,
  features = NULL,
  coef,
  coefLabel = coef,
  featuresIdVar = character(),
  featuresVar = featuresIdVar,
  featuresColor = "black",
  featuresMaxNChar = 60,
  fdr = NULL,
  logFCrange = NULL,
  xlab = "logFC",
  ylab = NULL,
  axesCex = 1,
  axesTitleCex = 1.1,
  title = NULL,
  titleCex = 1.1,
  legendPosition = c("right", "bottom", "none"),
  legendTitleCex = 1,
```

```
  legendCex = 0.8,
  facetCex = 1,
  facetColor = "black",
  facetNCol = grDevices::n2mfrow(length(coef))[2],
  fillVar = character(),
  fill = if (length(fillVar) > 0) character() else "skyblue2",
  colorVar = character(),
  color = if (length(colorVar) > 0) character() else "white",
  alphaVar = character(),
  alpha = if (length(alphaVar) > 0) 1 else numeric(),
  alphaRange = numeric(),
  typePlot = c("static", "interactive")
)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
features	IDs of features to show. If NULL (by default), show top 20 features.
coef	character, coefficient names.
coefLabel	character vector of labels or a function to transform existing labels.
featuresIdVar	column name with feature ids, empty by default. If specified and input is a model, featuresIdVar should be a column name in 'genes' slot.
featuresVar	column name with feature ids to label, the same as 'featuresIdVar' by default.
featuresColor	character vector specifying colors to use for the feature label text. Either length 1 or the same length as features, allowing particular features to be highlighted.
featuresMaxNChar	numeric, maximum number of characters to truncate the feature labels to.
fdr	threshold considered for significance, NULL by default.
logFCrange	numeric, two values (upper and lower bounds for logFC).
xlab	x-axis title, 'logFC' by default.
ylab	y-axis title, NULL by default.
axesCex	cex for the axis text.
axesTitleCex	cex for the axis title.
title	plot title, NULL by default.
titleCex	cex for the plot title.
legendPosition	legend position ("right", "bottom", "none"). If 'none', no legend is shown.
legendTitleCex	cex for the legend title.
legendCex	cex for the legend text.
facetCex	cex for the facets if multiple plot are used.
facetColor	color for the text of the facets
facetNCol	number of columns in facets, by default the function n2mfrow is used to.
fillVar	name of variable (in 'input') used for filling, empty by default

fill	character or factor with specified color(s) for the boxplot inside, replicated if needed. By default: 'skyblue2' if fillVar is not specified and default ggplot palette otherwise.
colorVar	name of variable (in 'input') used for coloring, empty by default
color	character or factor with specified color(s) for the bar border, replicated if needed. By default: 'skyblue2' if colorVar is not specified and default ggplot palette otherwise.
alphaVar	column name used for the transparency, empty by default.
alpha	character or factor with specified transparency(s) for the bars, replicated if needed. By default: '1' if alphaVar is not specified.
alphaRange	transparency (alpha) range used in the plot, possible only if the alphaVar is 'numeric' or 'integer'.
typePlot	plot type, can be one of "static" or "interactive".

Value

ggplot object

Author(s)

Katarzyna Gorczak

Examples

```
exampleData <- createExampleData(path = ".", output = "limma")
model <- exampleData$limma

# Simple waterfall plot
daWaterfallPlot(input = model, coef = "B.LvsP")

# see vignette for other examples
```

---

downloadData *Download example data from https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE6045*

Description

Download example data from https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE60450

Usage

```
downloadData(path, quiet)
```

Arguments

path	absolute path where data should be downloaded to
quiet	if TRUE, suppress status messages (if any), and the progress bar (see `download.file` function)

Value

(invisibly) downloads files at specific location ('path')

Author(s)

Katarzyna Gorczak

`extractColsOfInterest` *Extract columns of interest from top table*

Description

Extract columns of interest from top table

Usage

```
extractColsOfInterest(
  mean,
  topTableCoef,
  columns,
  stat,
  errorBars,
  coefI,
  text
)
```

Arguments

`mean`	logical (FALSE by default) whether to extract column with averaged expression or logCPM ("AveExpr" or "logCPM") depending on the `input`
`topTableCoef`	top table for one coefficient
`columns`	columns to extract
`stat`	logical (FALSE by default) whether to extract column with statistic: 't' or 'F'
`errorBars`	logical whether to add standard errors. Allowed only for `input` of class `"MArrayLM"` (or list of those).
`coefI`	coefficient of interest
`text`	(optional) String with name of a column, or function to extract a text from the columns of a top table, to display as text. See the available columns in the top table in section: Top table format.

Value

character vector

Author(s)

Laure Cougnaud, Katarzyna Gorczak

extractFeatures *Extract feature ids for each overlapping set*

Description

Extract feature ids for each overlapping set

Usage

```
extractFeatures(data)
```

Arguments

data	data.frame with 0s and 1s; the number of columns corresponds to the number of coefs and the number of rows corresponds to the number of unique feature ids each column is filled with 0s or 1s depending on the overlapping set

Value

list with features in overlapping sets

Author(s)

Katarzyna Gorczak

extractPairs *Get top table per pair (ref and comp coef) for scatter plot*

Description

Get top table per pair (ref and comp coef) for scatter plot

Usage

```
extractPairs(
  decideTestsOutput,
  refCoef,
  iCoef,
  multiplePlot,
  compCoef,
  tbl,
  typePlot,
  columns
)
```

Arguments

decideTestsOutput	output of decideTests
refCoef	coefficient on the x-axis
iCoef	coefficients on the y-axis
multiplePlot	whether to facet on coefs to compare
compCoef	coefficients to compare with refCoef
tbl	list of top tables
typePlot	plot type, can be one of "static" or "interactive"
columns	columns of interest from top table

Value

ggplot object

Author(s)

Katarzyna Gorczak

---

extractQueryList *Extract query list*

Description

Extract query list

Usage

```
extractQueryList(dataFrameSG)
```

Arguments

dataFrameSG	data.frame compatible with UpSetR

Value

a list with queries for upset plot

`extractTopTables` *Extract top tables for coefficients of interest*

Description

Extract top tables for coefficients of interest

Usage

```
extractTopTables(
  input,
  coef,
  coefLabel = NULL,
  features = NULL,
  n = 20,
  columns,
  errorBars = FALSE,
  mean = FALSE,
  stat = FALSE,
  featuresIdVar = character(),
  hoverText = FALSE,
  text = NULL,
  output = c("table", "list")
)
```

Arguments

`input`	model or a list with top tables named with coefficients. For model: object of class `MArrayLM` (linear model, see `eBayes`), `DGELRT` (see `edgeR`) or `DESeqResults` (see `DESeq2`) are supported. See more details on the top table format in section: Top table format.
`coef`	character vector with coefficients
`coefLabel`	character vector with coefficient labels or list with such vector (in case of multiple labels)
`features`	character vector with features to subset `input`. If NULL, top table for the top `n` features (for the first coefficient) is extracted.
`n`	top features to extract. If set to Inf, all features are extracted
`columns`	columns to extract
`errorBars`	logical whether to add standard errors. Allowed only for `input` of class `"MArrayLM"` (or list of those).
`mean`	logical (FALSE by default) whether to extract column with averaged expression or logCPM ("AveExpr" or "logCPM") depending on the `input`
`stat`	logical (FALSE by default) whether to extract column with statistic: 't' or 'F'
`featuresIdVar`	column name with unique feature ids (empty by default)
`hoverText`	logical whether to add point label for interactive plot (FALSE by default)
`text`	(optional) String with name of a column, or function to extract a text from the columns of a top table, to display as text. See the available columns in the top table in section: Top table format.
`output`	output can be one of "table" or "list"

Value

data.frame with top tables or list of those (if `output` is 'list') for coefficients of interest

Author(s)

Katarzyna Gorczak, Laure Cougnaud

`extractTTcoef` *Extract top table for a specific coef*

Description

Extract top table for a specific coef

Usage

```
extractTTcoef(input, coefI, errorBars, text)
```

Arguments

`input`	model or a list with top tables named with coefficients. For model: object of class `MArrayLM` (linear model, see `eBayes`), `DGELRT` (see `edgeR`) or `DESeqResults` (see `DESeq2`) are supported. See more details on the top table format in section: Top table format.
`coefI`	coefficient of interest
`errorBars`	logical whether to add standard errors. Allowed only for `input` of class `"MArrayLM"` (or list of those).
`text`	(optional) String with name of a column, or function to extract a text from the columns of a top table, to display as text. See the available columns in the top table in section: Top table format.

Value

data.frame with statistics for a specific coef

Author(s)

Laure Cougnaud, Katarzyna Gorczak

facet *Add facets to ggplot object*

Description

Add facets to ggplot object

Usage

```
facet(g, topTableOutput, facetNCol, scales = "fixed")
```

Arguments

g	ggplot object
topTableOutput	combined topTables for all coefficients
facetNCol	number of columns in facets, by default the function n2mfrow is used
scales	see facet_wrap or facet_nested_wrap

Value

ggplot object

Author(s)

Laure Cougnaud, Katarzyna Gorczak

filterCommonFeatures *Filter top tables based on common features across all tables*

Description

Filter top tables based on common features across all tables

Usage

```
filterCommonFeatures(input, featuresIdVar)
```

Arguments

input	list of top tables
featuresIdVar	column with feature identifier (must be unique)

Value

list of top tables with common features

Author(s)

Katarzyna Gorczak

`filterGenesOfInterest` *Filter genes of interest*

Description

Filter genes of interest

Usage

```
filterGenesOfInterest(
  input,
  featuresIdVar = character(),
  genesToHighlightVar,
  genesToHighlightThresholdPValue = 1,
  genesToHighlightThresholdLogFC = NULL
)
```

Arguments

`input`	long-format table
`featuresIdVar`	column name with feature ids, 'ENTREZID' by default.
`genesToHighlightVar`	column name with genes to highlight
`genesToHighlightThresholdPValue`	numeric, if specified keep among the genes to highlight, the genes which have a raw p-value lower (strict) than this threshold for at least one of the coefficient considered
`genesToHighlightThresholdLogFC`	numeric, if specified keep among the genes to highlight, the genes which have an absolute log FC higher (strict) than this threshold for at least one of the coefficient considered

Value

long-format table

Author(s)

Laure Cougnaud, Kirsten Van Hoorde, Katarzyna Gorczak

`filterLogFC` *Filter top tables based on logFC threshold*

Description

Filter top tables based on logFC threshold

Usage

```
filterLogFC(input, logFCrange)
```

Arguments

input	list of top tables
logFCrange	numeric vector with two values (low- and high-threshold for log-foldchange)

Value

list of top tables with subset of features based on logFC range

Author(s)

Katarzyna Gorczak

---

formatAesMA *Format aesthetics for the main plot with MA plot*

Description

Format aesthetics for the main plot with MA plot

Usage

```
formatAesMA(g, topTableOutput, direction, color, sizeVar, size)
```

Arguments

g	ggplot object with MA plot
topTableOutput	combined topTables for all coefficients
direction	logical whether to color significant up- and down-regulated genes
color	colors for points indicating direction (should be three colors: significant up- and down-regulated and non-significant)
sizeVar	column name used for the size, empty by default.
size	character or factor with specified size(s) (cex) for the points, replicated if needed. This is used only if sizeVar is empty. By default: '2.5' if sizeVar is not specified and default ggplot size(s) otherwise

Value

ggplot object

Author(s)

Katarzyna Gorczak

formatAesVP *Format aesthetics for the main plot with volcano plot*

Description

Format aesthetics for the main plot with volcano plot

Usage

```
formatAesVP(
  g,
  topTableOutput,
  colorVar,
  color,
  shapeVar,
  shape,
  alphaVar,
  alpha,
  alphaRange,
  sizeVar,
  size,
  sizeRange
)
```

Arguments

g	ggplot object with volcano plot
topTableOutput	combined topTables for all coefficients
colorVar	(optional) string with column name containing variable used for coloring. If not specified, coloring is based on the adjusted p-values.
color	color palette, only used if colorVar is specified.
shapeVar	(optional) string with column name containing variable used for shaping. If not specified, only points are used.
shape	shape palette, only used if shapeVar is specified.
alphaVar	column name used for the transparency, empty by default.
alpha	character or factor with specified transparency(s) for the points, replicated if needed. By default: '1' if alphaVar is not specified.
alphaRange	transparency (alpha) range used in the plot, possible only if the alphaVar is 'numeric' or 'integer'.
sizeVar	column name used for the size, empty by default.
size	character or factor with specified size(s) (cex) for the points, replicated if needed. This is used only if sizeVar is empty. By default: '2.5' if sizeVar is not specified and default ggplot size(s) otherwise
sizeRange	size (cex) range used in the plot, possible only if the sizeVar is 'numeric' or 'integer'

Value

ggplot object

Author(s)

Laure Cougnaud, Kirsten Van Hoorde, Katarzyna Gorczak

`formatAesWP` *Format aesthetics for the main plot with waterfall plot*

Description

Format aesthetics for the main plot with waterfall plot

Usage

```
formatAesWP(
  topTableOutput,
  colorVar,
  color,
  g,
  fillVar,
  fill,
  alphaVar,
  alpha,
  alphaRange
)
```

Arguments

`topTableOutput`	combined topTables for all coefficients
`colorVar`	name of variable (in 'input') used for coloring, empty by default
`color`	character or factor with specified color(s) for the bar border, replicated if needed. By default: 'skyblue2' if colorVar is not specified and default ggplot palette otherwise.
`g`	ggplot object with volcano plot
`fillVar`	name of variable (in 'input') used for filling, empty by default
`fill`	character or factor with specified color(s) for the boxplot inside, replicated if needed. By default: 'skyblue2' if fillVar is not specified and default ggplot palette otherwise.
`alphaVar`	column name used for the transparency, empty by default.
`alpha`	character or factor with specified transparency(s) for the bars, replicated if needed. By default: '1' if alphaVar is not specified.
`alphaRange`	transparency (alpha) range used in the plot, possible only if the alphaVar is 'numeric' or 'integer'.

Value

ggplot object

Author(s)

Katarzyna Gorczak

formatManualScale *extend manual scale values if required*

Description

extend manual scale values if required

Usage

```
formatManualScale(x, valVar, nameVar)
```

Arguments

x	data.frame with nameVar
valVar	fixed value of variable of aesthetic
nameVar	name of variable for aesthetic

Value

vector of manual scales

Author(s)

Laure Cougnaud

formatTTcoef *Format top table for a specific coefficient*

Description

Format top table for a specific coefficient

Usage

```
formatTTcoef(
  topTableCoef,
  columns,
  mean,
  coefI,
  coef,
  featuresIdVar,
  hoverText
)
```

Arguments

topTableCoef	top table for one coefficient
columns	columns to extract
mean	logical (FALSE by default) whether to extract column with averaged expression or logCPM ("AveExpr" or "logCPM") depending on the input
coefI	coefficient of interest
coef	character vector with coefficients
featuresIdVar	column name with unique feature ids (empty by default)
hoverText	logical whether to add point label for interactive plot (FALSE by default)

Value

data.frame

Author(s)

Laure Cougnaud, Katarzyna Gorczak

---

formatVariableLength *shorten string length*

Description

shorten string length

Usage

```
formatVariableLength(x, var, length)
```

Arguments

x	data.frame with var
var	column name
length	max number of characters in each string

Value

string of length length

Character vector with shortened variable.

Author(s)

Katarzyna Gorczak

formatVariableSpace *remove space in variable name*

Description

remove space in variable name

Usage

```
formatVariableSpace(var)
```

Arguments

var	string

Value

String with formatted variable

getCoefColor *Create colors for coefficient labels*

Description

Create colors for coefficient labels

Usage

```
getCoefColor(coef, coefLabel, coefColor)
```

Arguments

coef	character, coefficient names.
coefLabel	character vector of labels or a function to transform the existing labels.
coefColor	color palette for coef labels.

Value

named character (named color palette)

Author(s)

Katarzyna Gorczak

getCoefLabel *Get coefficient label*

Description

Get coefficient label

Usage

```
getCoefLabel(coef, coefLabel)
```

Arguments

coef	character vector with coefficients
coefLabel	character vector of coefficient labels or a function to transform the coefficients

Value

labels of the coefficients; either a vector or a list with nested labels

getFeatureColor *Create colors for feature labels*

Description

Create colors for feature labels

Usage

```
getFeatureColor(
  topTableOutput,
  features,
  featuresIdVar,
  featuresColor,
  order = FALSE,
  plot = c("logRatio", "heatmap", "waterfall")
)
```

Arguments

topTableOutput	combined topTables for all coefficients
features	feature ids or numeric. If NULL (by default), show top 20 features according to the first coef. If numeric and input is a list, the tables are subsetted according to the row names of the table for the first coef.
featuresIdVar	column name with feature ids.
featuresColor	character vector specifying colors to use for the feature label text. Either length 1 or the same length as features, allowing particular features to be highlighted.
order	logical whether to order features (used only when plot is 'logRatio')
plot	plot

Value

named color palette for feature labels

Author(s)

Katarzyna Gorczak

---

getFeatures *Get features from input*

Description

Get features from input

Usage

```
getFeatures(input, featuresIdVar = character())
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
featuresIdVar	column with unique feature identifiers

Value

character with all features from input

Author(s)

Katarzyna Gorczak

---

getInputIdCoef *Get ID of the input* list *matching to the specific coefficient*

Description

Get ID of the input list matching to the specific coefficient

Usage

```
getInputIdCoef(input, coef)
```

Arguments

input	input list
coef	character vector of length 1 with coefficient

Value

numeric of length 1 specifying which element of the list contains the coefficient

---

`getModelCoefs` *Get coefficients in input*

Description

Get coefficients in input

Usage

```
getModelCoefs(input, coef = NULL)
```

Arguments

`input`	object of class `MArrayLM` (see `limma`), `DGELRT` (see `edgeR`) or `DESeqResults` (see `DESeq2`)
`coef`	(optional) coefficient of interest

Value

Character vector with coefficients

`getNumberOfRegulatedGenes` *Get number of significant genes for coefficient*

Description

Get number of significant genes for coefficient

Usage

```
getNumberOfRegulatedGenes(
  input,
  fdr = 0.05,
  logFCrange = NULL,
  dir = c("up", "down")
)
```

Arguments

`input`	a list of top tables.
`fdr`	threshold for adjusted p-value.
`logFCrange`	numeric, upper and lower bound for logFC
`dir`	direction, select either up- or down-regulated genes ('up' - genes with logFC larger than 0, 'down' - genes with logFC smaller than 0).

Value

named numeric vector with number of up- or down-regulated genes

Author(s)

Katarzyna Gorczak

`getNumberOfSignificantGenes` *Get number of significant genes for coefficient*

Description

Get number of significant genes for coefficient

Usage

```
getNumberOfSignificantGenes(input, logFCrange = NULL, fdr = 0.05)
```

Arguments

`input`	a list of top tables.
`logFCrange`	numeric, upper and lower bound for logFC
`fdr`	threshold for adjusted p-value.

Value

named numeric vector with number of significant genes

Author(s)

Katarzyna Gorczak

`getSEModel` *Get standard error from a limma model.*

Description

The standard errors are computed, according to BioC support question 70175, as:

$$stdev.unscaled * \sqrt{s2.post}$$

with:

- stdev.unscaled: unscaled standard deviations
- s2.post: posterior values for sigma^2

Usage

```
getSEModel(input, coef)
```

Arguments

input	object of class MArrayLM (see limma)
coef	character vector with one single coefficient

Value

If input is MArrayLM, a numeric vector with SE named by feature ID, NA otherwise.

Author(s)

Laure Cougnaud

Examples

```
exampleData <- createExampleData(path = ".", output = "limma")
model <- exampleData$limma
getSEModel(input = model, coef = "B.LvsP")
```

---

getTopFeatures *Get top features*

Description

Get top features

Usage

```
getTopFeatures(input, coef, featuresIdVar = character(), n = 10)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
coef	character vector with coefficient
featuresIdVar	column with unique feature identifiers
n	numeric, number of top features

Value

top n features based on their adjusted p-value

Author(s)

Laure Cougnaud, Katarzyna Gorczak

getTopGenes *Get top genes with highest significance and top genes with highest logFC*

Description

Get top genes with highest significance and top genes with highest logFC

Usage

```
getTopGenes(input, topGenes, featuresIdVar = character())
```

Arguments

input	long-format table
topGenes	number of top genes to extract
featuresIdVar	column with feature identifier (must be unique)

Value

data.frame with top genes

Author(s)

Laure Cougnaud, Katarzyna Gorczak

getTopTableFromModel *Extract top tables from a model*

Description

Extract top tables from a model

Usage

```
getTopTableFromModel(input, coef, se = FALSE)
```

Arguments

input	object of class MArrayLM (see limma), DGELRT (see edgeR) or DESeqResults (see DESeq2)
coef	coefficient of interest
se	Logical (FALSE by default), should standard errors be extracted?

Value

data.frame with top tables for coefficient of interest.
The features are ordered in original (unsorted) order in the model.
Standard errors, if requested are in the column: 'SE'.

Author(s)

Katarzyna Gorczak

`ggPlotTheme` *Customized ggplot theme*

Description

Customized ggplot theme

Usage

```
ggPlotTheme(
  ggObject,
  panelBackground = NULL,
  panelBorder = "black",
  title = NULL,
  titleSize = 2,
  xTextSize = 1.2,
  xTextColor = "black",
  xTitle = NULL,
  xTitleSize = 1.4,
  xTextAngle = 0,
  xTextHjust = 0.5,
  yTextSize = 1.2,
  yTextColor = "black",
  yTitle = NULL,
  yTitleSize = 1.4,
  facetLabelSize = 1.5,
  facetLabelColor = "black",
  gridMajor = TRUE,
  gridMinor = TRUE,
  yTicks = TRUE,
  xTicks = TRUE,
  legendPosition = "right",
  legendTitleSize = 1.3,
  legendTextSize = 1.1
)
```

Arguments

`ggObject`	ggplot object
`panelBackground`	panel background of the entire plot
`panelBorder`	border around plotting area, by default black
`title`	plot title
`titleSize`	cex for plot title, relative to the parent
`xTextSize`	cex for axis text
`xTextColor`	color font on the x axis, by default black

xTitle	title of the x axis
xTitleSize	cex for the x axis title
xTextAngle	angle of the x axis labels
xTextHjust	numeric for horizontal adjustment, 0.5 by default
yTextSize	cex for axis text
yTextColor	color font on the y axis, by default black
yTitle	title of the y axis, NULL by default
yTitleSize	cex for the y axis title
facetLabelSize	cex for the facet labels
facetLabelColor	color of the text in facets
gridMajor	logical whether to show the major grid lines
gridMinor	logical whether to show the minor grid lines
yTicks	logical whether to show ticks on the y axis
xTicks	logical whether to show ticks on the x axis
legendPosition	legend position, by default right
legendTitleSize	cex for the legend title
legendTextSize	cex for the legend

Value

ggplot object with customized theme

---

isModel *Check if input is a model*

Description

Check if input is a model

Usage

```
isModel(input)
```

Arguments

input	object of class MArrayLM (see limma), DGELRT (see edgeR) or DESeqResults (see DESeq2)

Value

logical

isTopTable — *Check if input is a topTable containing colums with logFC, p-value and adjusted p-value*

Description

Check if input is a topTable containing colums with logFC, p-value and adjusted p-value

Usage

```
isTopTable(input)
```

Arguments

input	a top table (see `limma::topTable`)

Value

logical

labelCorr — *Add label with correlation to scatter plot*

Description

Add label with correlation to scatter plot

Usage

```
labelCorr(topTableOutput, g, correlationCex)
```

Arguments

topTableOutput	combined topTables for all coefficients
g	ggplot object
correlationCex	cex for the text with correlation value

Value

ggplot object

Author(s)

Katarzyna Gorczak

`labelGenesOfInterest` *Add layer with genes of interest to ggplot object*

Description

Add layer with genes of interest to ggplot object

Usage

```
labelGenesOfInterest(
  g,
  topTableOutputGenesOfInterest,
  typePlot,
  color = "red",
  sizeVar = character(),
  size = if (length(sizeVar) > 0) numeric() else 2,
  genesToHighlightCex,
  ...
)
```

Arguments

`g`	ggplot object
`topTableOutputGenesOfInterest`	data.frame with `genesToHighlight`
`typePlot`	plot type can be one of "static" or "interactive".
`color`	color palette, only used if `colorVar` is specified.
`sizeVar`	column name used for the size, empty by default.
`size`	character or factor with specified size(s) (cex) for the points, replicated if needed. This is used only if sizeVar is empty. By default: '2.5' if sizeVar is not specified and default ggplot size(s) otherwise
`genesToHighlightCex`	cex for genesToHighlight
`...`	Extra parameters passed to `geom_text_repel` to customize the position of the gene labels.

Value

ggplot object

Author(s)

Laure Cougnaud, Katarzyna Gorczak

labelTextLRP *Add optional text to the log-ratio plot*

Description

Add optional text to the log-ratio plot

Usage

```
labelTextLRP(topTableOutput, errorBars, g, textVar, textVarCex)
```

Arguments

topTableOutput	combined topTables for all coefficients
errorBars	logical whether to add error bars to the plot (+/- one standard error). If input (or each element within a input list) is a: • model: only supported for MArrayLM • top table: a 'se' column should be available
g	ggplot object with volcano plot
textVar	(optional) String with name of a column to display as text.
textVarCex	cex for the text next to bars

Value

ggplot object

Author(s)

Laure Cougnaud, Kirsten Van Hoorde, Katarzyna Gorczak

labelTopGenes *Add point labels to ggplot object*

Description

Add point labels to ggplot object

Usage

```
labelTopGenes(
  g,
  includeTableGenesOfInterest,
  topTableOutputTopGenes,
  topTableOutputGenesOfInterest,
  colorVar,
  topGenesCex,
  ...
)
```

Arguments

g	ggplot object
includeTableGenesOfInterest	whether to label genesToHighlight
topTableOutputTopGenes	data.frame with top genes
topTableOutputGenesOfInterest	data.frame with genesToHighlight
colorVar	string with column name containing variable used for coloring
topGenesCex	cex for topGenes labels
...	Extra parameters passed to geom_text_repel to customize the position of the gene labels

Value

ggplot object

Author(s)

Laure Cougnaud, Katarzyna Gorczak

---

mainH *Create main plot object with heatmap*

Description

Create main plot object with heatmap

Usage

```
mainH(typePlot, topTableOutput, color, colorNA)
```

Arguments

typePlot	plot type, can be one of "static" or "interactive".
topTableOutput	combined topTables for all coefficients
color	palette for gradient to fill the heatmap.
colorNA	color for missing values.

Value

ggplot object

Author(s)

Katarzyna Gorczak

mainLRP *Create main plot object with log-ratio plot*

Description

Create main plot object with log-ratio plot

Usage

```
mainLRP(typePlot, color, topTableOutput, errorBars)
```

Arguments

typePlot	plot type, can be one of "static" or "interactive".
color	string with colors for the bars, one per coefficient.
topTableOutput	combined topTables for all coefficients
errorBars	logical whether to add error bars to the plot (+/- one standard error). If input (or each element within a input list) is a: • model: only supported for MArrayLM • top table: a 'se' column should be available

Value

ggplot object

Author(s)

Laure Cougnaud, Kirsten Van Hoorde, Katarzyna Gorczak

mainMA *Create main plot object with MA plot*

Description

Create main plot object with MA plot

Usage

```
mainMA(typePlot, direction, color, sizeVar, topTableOutput, alpha, size)
```

Arguments

typePlot	plot type can be one of "static" or "interactive".
direction	logical whether to color significant up- and down-regulated genes
color	colors for points indicating direction (should be three colors: significant up- and down-regulated and non-significant)
sizeVar	column name used for the size, empty by default.
topTableOutput	combined topTables for all coefficients
alpha	transparency level for the points, 0.4 by default.
size	character or factor with specified size(s) (cex) for the points, replicated if needed. This is used only if sizeVar is empty. By default: '2.5' if sizeVar is not specified and default ggplot size(s) otherwise

Value

ggplot object

Author(s)

Katarzyna Gorczak

---

mainSP *Create main plot object with scatter plot*

Description

Create main plot object with scatter plot

Usage

```
mainSP(
  typePlot,
  color,
  topTableOutput,
  xlab,
  coef,
  ylab,
  multiplePlot,
  pointSize,
  alpha
)
```

Arguments

typePlot	plot type, can be one of "static" or "interactive"
color	color palette to distinguish significance groups. Four colors must be specified.
topTableOutput	combined topTables for all coefficients
xlab	x-axis title, NULL by default.
coef	character, coefficient names.

ylab	y-axis title, NULL by default.
multiplePlot	logical whether to use facet on coefficients
pointSize	point size, 2 by default.
alpha	transparency level for the points, 0.4 by default.

Value

ggplot object

Author(s)

Katarzyna Gorczak

`mainVP` *Create main plot object with volcano plot*

Description

Create main plot object with volcano plot

Usage

```
mainVP(
  colorVar,
  shapeVar,
  alphaVar,
  sizeVar,
  topTableOutput,
  typePlot,
  color,
  shape,
  alpha,
  size
)
```

Arguments

colorVar	(optional) string with column name containing variable used for coloring. If not specified, coloring is based on the adjusted p-values.
shapeVar	(optional) string with column name containing variable used for shaping. If not specified, only points are used.
alphaVar	column name used for the transparency, empty by default.
sizeVar	column name used for the size, empty by default.
topTableOutput	combined topTables for all coefficients
typePlot	plot type can be one of "static" or "interactive".
color	color palette, only used if `colorVar` is specified.
shape	shape palette, only used if `shapeVar` is specified.

alpha	character or factor with specified transparency(s) for the points, replicated if needed. By default: '1' if alphaVar is not specified.
size	character or factor with specified size(s) (cex) for the points, replicated if needed. This is used only if sizeVar is empty. By default: '2.5' if sizeVar is not specified and default ggplot size(s) otherwise

Value

ggplot object

Author(s)

Laure Cougnaud, Kirsten Van Hoorde, Katarzyna Gorczak

`mainWP` *Create main plot object with waterfall plot*

Description

Create main plot object with waterfall plot

Usage

```
mainWP(
  typePlot,
  colorVar,
  fillVar,
  alphaVar,
  topTableOutput,
  fill,
  color,
  alpha
)
```

Arguments

typePlot	plot type, can be one of "static" or "interactive".
colorVar	name of variable (in 'input') used for coloring, empty by default
fillVar	name of variable (in 'input') used for filling, empty by default
alphaVar	column name used for the transparency, empty by default.
topTableOutput	combined topTables for all coefficients
fill	character or factor with specified color(s) for the boxplot inside, replicated if needed. By default: 'skyblue2' if fillVar is not specified and default ggplot palette otherwise.
color	character or factor with specified color(s) for the bar border, replicated if needed. By default: 'skyblue2' if colorVar is not specified and default ggplot palette otherwise.
alpha	character or factor with specified transparency(s) for the bars, replicated if needed. By default: '1' if alphaVar is not specified.

Value

ggplot object

Author(s)

Katarzyna Gorczak

---

makeElementsUnique *Make elements unique*

Description

Make elements unique

Usage

```
makeElementsUnique(x)
```

Arguments

x	vector with values

Value

vector with unique values, for duplicated values a suffix '_' is added

---

modelExampleData *Normalize data, create model matrix and contrasts*

Description

Normalize data, create model matrix and contrasts

Usage

```
modelExampleData(eset)
```

Arguments

eset	ExpressionSet

Value

a list with data, model, design matrix and contrast matrix

Author(s)

Katarzyna Gorczak

orderComparison *Sort the comparison columns according to the coefficients*

Description

Sort the comparison columns according to the coefficients

Usage

```
orderComparison(df, coefLabel)
```

Arguments

df	top table for one coefficient
coefLabel	character vector with coefficient labels or list with such vector (in case of multiple labels)

Value

data.frame

Author(s)

Laure Cougnaud, Katarzyna Gorczak

orderFeatures *Order features based on similarity or significance*

Description

Order features based on similarity or significance

Usage

```
orderFeatures(input, featuresOrder, featuresIdVar)
```

Arguments

input	list of top tables
featuresOrder	if not NULL, features in the graph are re-ordered, either based on similarity ('similarity') or significance ('significance') of the statistics. See section 'Feature ordering'.
featuresIdVar	column name with feature ids.

Value

ordered features

Feature ordering

The features are ordered based on:

- 'similarity': a hierarchical clustering of the (Euclidean) distances between the statistics. The statistics are for:
  - limma: the t-statistic of each coefficient
  - edgeR: the (overall) F-statistic is considered
- 'significance': decreasing average (-log10) p-values across coefficients. The p-values are for:
  - limma: the significance of each coefficient
  - edgeR: the (overall) p-value of the model

---

processFeatures *Process table for visualizations.*

Description

Some can be missing across different coefficients (either different subsets of to tables are provided or provided 'features' are not present in each top table). The table will be processed for each coefficient:

- 1: missing features will be added (if a missing feature in coef X is present in coef Y, this label will be used)
- 2: feature missing in all coefs will obviously not be plotted
- 3: the same 'featuresVar' can happen for different 'featuresIdVar' - add index to duplicates as 'featuresVar' is used for plotting

Usage

```
processFeatures(tbl, coef, featuresIdVar, order)
```

Arguments

`tbl`	logn-form table (top tables for specified coefs)
`coef`	character, coefficient names.
`featuresIdVar`	column name with feature ids.
`order`	character vector with feature ids in a specific order

Value

data.frame with processed features

Author(s)

Katarzyna Gorczak

reshapeTable *Reshape table*

Description

Reshape table

Usage

```
reshapeTable(table, timevar)
```

Arguments

table	long- or wide-format table
timevar	column name in long-format table that differentiates multiple records from the same group

Value

a list of tables

Author(s)

Katarzyna Gorczak

runDESeq2 *Fit the model with DESeq2*

Description

Fit the model with DESeq2

Usage

```
runDESeq2(input)
```

Arguments

input	list of objects (data, contrasts and model matrix)

Value

DESeq2 output from results() function

Author(s)

Katarzyna Gorczak

runEdgeR *Fit the model with edgeR*

Description

Fit the model with edgeR

Usage

```
runEdgeR(input)
```

Arguments

input	list of objects (data, contrasts and model matrix)

Value

edgeR output from glmQLFTest() function

Author(s)

Katarzyna Gorczak

runLimma *Fit the model with limma*

Description

Fit the model with limma

Usage

```
runLimma(input)
```

Arguments

input	list of objects (data, contrasts and model matrix)

Value

limma outpur from eBayes() function

Author(s)

Katarzyna Gorczak

runTopTable *Fit the model and return top table*

Description

Fit the model and return top table

Usage

```
runTopTable(input)
```

Arguments

input	list of objects (data, contrasts and model matrix)

Value

a list of top tables

Author(s)

Katarzyna Gorczak

setCategoricalElement *check if variable is not numeric (or integer)*

Description

check if variable is not numeric (or integer)

Usage

```
setCategoricalElement(x, typeVar)
```

Arguments

x	data.frame
typeVar	column name in x

Value

Logical, TRUE if the variable is not a numeric or integer

setFixElement *check if the aesthetic is fixed (e.g. color, shape, size 'palette')*

Description

check if the aesthetic is fixed (e.g. color, shape, size 'palette')

Usage

```
setFixElement(typeVar, valVar)
```

Arguments

typeVar	name of variable for aesthetic
valVar	fixed value of variable of aesthetic

Value

logical, if TRUE the element is fixed

Author(s)

Laure Cougnaud

setGradientScale *check if manual aesthetic for the gradient should be set*

Description

This is the case only if typeVar and valVar are specified, and if the variable is numeric or integer

Usage

```
setGradientScale(x, typeVar, valVar)
```

Arguments

x	data.frame with typeVar
typeVar	name of variable for aesthetic
valVar	fixed value of variable of aesthetic

Value

logical, if TRUE the manual scale should be set

Author(s)

Katarzyna Gorczak

setManualScale *check if manual aesthetic should be set*

Description

This is the case only if typeVar and valVar are specified, and if the variable is not numeric or integer (doesn't work with ggplot2)

Usage

```
setManualScale(x, typeVar, valVar)
```

Arguments

x	data.frame with typeVar
typeVar	name of variable for aesthetic
valVar	fixed value of variable of aesthetic

Value

logical, if TRUE the manual scale should be set

Author(s)

Laure Cougnaud

subsetFeatures *Subset input with specific features*

Description

Subset input with specific features

Usage

```
subsetFeatures(input, features, featuresIdVar = character())
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
features	features of interest
featuresIdVar	column with unique feature identifiers

Value

input with subset of features

Author(s)

Katarzyna Gorczak

---

topFeaturesDeseq *Extract top features from* DESeq2 *output*

Description

Extract top features from DESeq2 output

Usage

```
topFeaturesDeseq(input, n, featuresIdVar)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
n	numeric, number of top features
featuresIdVar	column with unique feature identifiers

Value

character vector with feature names

Author(s)

Katarzyna Gorczak

---

topFeaturesEdger *Extract top features from* edgeR *output*

Description

Extract top features from edgeR output

Usage

```
topFeaturesEdger(input, n, featuresIdVar)
```

Arguments

input	model or a list with top tables named with coefficients. For model: object of class MArrayLM (linear model, see eBayes), DGELRT (see edgeR) or DESeqResults (see DESeq2) are supported. See more details on the top table format in section: Top table format.
n	numeric, number of top features
featuresIdVar	column with unique feature identifiers

Value

character vector with feature names

Author(s)

Katarzyna Gorczak

`topFeaturesLimma` *Extract top features from* `limma` *output*

Description

Extract top features from `limma` output

Usage

```
topFeaturesLimma(input, coef, n, featuresIdVar)
```

Arguments

`input`	model or a list with top tables named with coefficients. For model: object of class `MArrayLM` (linear model, see `eBayes`), `DGELRT` (see `edgeR`) or `DESeqResults` (see `DESeq2`) are supported. See more details on the top table format in section: Top table format.
`coef`	character vector with coefficient
`n`	numeric, number of top features
`featuresIdVar`	column with unique feature identifiers

Value

character vector with feature names

Author(s)

Katarzyna Gorczak

Index

addHoverText, 4
arrangeTopTables, 4

calcCorrelation, 5
callBarplot, 6
callHeatmap, 7
callLogRatioPlot, 8
callMAplot, 9
callScatterPlot, 11
callVolcanoPlot, 12
callWaterfallPlot, 15
cbindFill, 16
checkCoef, 17
checkColumns, 17
checkInput, 18
colorBlindPalette, 18
concatenateVars, 19
createDataBarplot, 19
createDataHeatmap, 20
createDataLogRatioPlot, 21
createDataMAplot, 22
createDataScatterPlot, 23
createDataUpsetPlot, 25
createDataVolcanoPlot, 26
createDataWaterfallPlot, 27
createExampleData, 29
createExpressionSet, 29
createPairData, 30
createSampleAnnotation, 30
createTopTableGenesOfInterest, 31

daHeatmapLogFC, 32
daLogRatioPlot, 34
daMAplot, 36
daScatterPlot, 39
daSignificantGenesBarplot, 42
daUpset, 43
daVis-common-args, 45
daVis-common-doc, 45
daVolcanoPlot, 46
daWaterfallPlot, 49
downloadData, 51

eBayes, *17, 18*, *20, 21*, *23–26*, *28*, *32*, *34*, *37*, *40*, *42*, *44, 45*, *47*, *50*, *55, 56*, *66*, *69*, *88–90*
expansion, *8*, *35*
extractColsOfInterest, 52
extractFeatures, 53
extractPairs, 53
extractQueryList, 54
extractTopTables, 55
extractTTcoef, 56

facet, 57
filterCommonFeatures, 57
filterGenesOfInterest, 58
filterLogFC, 58
formatAesMA, 59
formatAesVP, 60
formatAesWP, 61
formatManualScale, 62
formatTTcoef, 62
formatVariableLength, 63
formatVariableSpace, 64

getCoefColor, 64
getCoefLabel, 65
getFeatureColor, 65
getFeatures, 66
getInputIdCoef, 66
getModelCoefs, 67
getNumberOfRegulatedGenes, 67
getNumberOfSignificantGenes, 68
getSEModel, 68
getTopFeatures, 69
getTopGenes, 70
getTopTableFromModel, 70
ggPlotTheme, 71

isModel, 72
isTopTable, 73

labelCorr, 73
labelGenesOfInterest, 74
labelTextLRP, 75
labelTopGenes, 75

mainH, 76

mainLRP, 77
mainMA, 77
mainSP, 78
mainVP, 79
mainWP, 80
makeElementsUnique, 81
modelExampleData, 81

orderComparison, 82
orderFeatures, 82

processFeatures, 83

reshapeTable, 84
runDESeq2, 84
runEdgeR, 85
runLimma, 85
runTopTable, 86

setCategoricalElement, 86
setFixElement, 87
setGradientScale, 87
setManualScale, 88
subsetFeatures, 88

Top table format, *17, 18*, *20–26*, *28*, *32*, *34, 35*, *37*, *40*, *42*, *44, 45*, *47*, *50*, *52*, *55, 56*, *66*, *69*, *88–90*
topFeaturesDeseq, 89
topFeaturesEdger, 89
topFeaturesLimma, 90