Package ‘vidger’

October 17, 2020

Title Create rapid visualizations of RNAseq data in R

Version 1.8.0

Description The aim of vidger is to rapidly generate information-rich visualizations for the interpretation of differential gene expression results from three widely-used tools: Cuffdiff, DESeq2, and edgeR.

URL `https://github.com/btmonier/vidger`,
`https://bioconductor.org/packages/release/bioc/html/vidger.html`

BugReports `https://github.com/btmonier/vidger/issues`

Depends R (>= 3.5)

Imports Biobase, DESeq2, edgeR, GGally, ggplot2, ggrepel, knitr, RColorBrewer, rmarkdown, scales, stats, SummarizedExperiment, tidyr, utils

Suggests BiocStyle, testthat

Encoding UTF-8

LazyData true

VignetteBuilder knitr

RoxygenNote 6.0.1

License GPL-3 | file LICENSE

biocViews ImmunoOncology, Visualization, RNASeq, DifferentialExpression, GeneExpression, ImmunoOncology

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

git_branch RELEASE_3_11

git_last_commit a2dcfcd

git_last_commit_date 2020-04-27

Date/Publication 2020-10-16

Author Brandon Monier [aut, cre],
Adam McDermaid [aut],
Jing Zhao [aut],
Qin Ma [aut, fnd]

Maintainer Brandon Monier <`brandon.monier@gmail.com`>

R topics documented:

`df.cuff` *A ‘cuffdiff‘ example dataset*

Description

A ‘cuffdiff‘ example dataset

Usage

```
data(df.cuff)
```

`df.deseq` *A ‘DESeq2‘ example dataset*

Description

A ‘DESeq2‘ example dataset

Usage

```
data(df.deseq)
```

`df.edger` *A ‘edgeR‘ example dataset*

Description

A ‘edgeR‘ example dataset

Usage

```
data(df.edger)
```

`vsBoxPlot` *Box plot for log_10(FPKM or CPM) distributions*

Description

This function allows you to extract necessary results-based data from analytical objects to create a box plot comparing log10(FPKM or CPM) distributions for experimental treatments.

Usage

```
vsBoxPlot(data, d.factor = NULL, type = c("cuffdiff", "deseq", "edger"),
  title = TRUE, legend = TRUE, grid = TRUE, aes = c("box", "violin",
  "boxdot", "viodot", "viosumm", "notch"), fill.color = NULL,
  data.return = FALSE, xaxis.text.size = 10, yaxis.text.size = 10,
  xaxis.title.size = 12, yaxis.title.size = 12, main.title.size = 15,
  legend.text.size = 10, legend.title.size = 12)
```

Arguments

`data`	output generated from calling the main routines of either `cuffdiff`, `DESeq2`, or `edgeR` analyses. For `cuffdiff`, this will be a `*_exp.diff` file. For `DESeq2`, this will be a generated object of class `DESeqDataSet`. For `edgeR`, this will be a generated object of class `DGEList`.
`d.factor`	a specified factor; for use with `DESeq2` objects only. This input equates to the first parameter for the contrast argument when invoking the `results()` function in `DESeq2`. Defaults to `NULL`
`type`	an analysis classifier to tell the function how to process the data. Must be either `cuffdiff`, `deseq`, or `edger`. `cuffdiff` must be used with `cuffdiff` data; `deseq` must be used for `DESeq2` output; `edgeR` must be used with `edgeR` data. See the `data` parameter for further details.
`title`	display the main title of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no title will display in plot.
`legend`	display legend of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no legend will display in plot.
`grid`	display major and minor axis lines. Logical; defaults to `TRUE`. If set to `FALSE`, no axis lines will display in plot.
`aes`	changes overall layout of the graph. `box`: box plot; `violin`: violin plot; `boxdot`: box plot with dots; `viodot`: violin plot with dots; `viosumm`: violin plot with summary statistics; `notch`: box plots with notches. Defaults to `box`.
`fill.color`	changes the fill color for the plots. See `RColorBrewer::display.brewer.all()` function for further details. If `NULL`, colors will default to standard `ggplot2` aesthetics.
`data.return`	returns data output of plot. Logical; defaults to `FALSE`. If set to `TRUE`, a data frame will also be called. Assign to object for reproduction and saving of data frame. See final example for further details.
`xaxis.text.size`	change the font size of the `x`-axis text. Defaults to `10`.

`yaxis.text.size`	change the font size of the y-axis text. Defaults to `10`.
`xaxis.title.size`	change the font size of the x-axis title text. Defaults to `12`.
`yaxis.title.size`	change the font size of the y-axis title text. Defaults to `12`.
`main.title.size`	change the font size of the plot title text. Defaults to `15`.
`legend.text.size`	change the font size of the legend body text. Defaults to `10`.
`legend.title.size`	change the font size of the legend title text. Defaults to `12`.

Value

An object created by `ggplot`

Author(s)

Brandon Monier, `<brandon.monier@sdstate.edu>`

Examples

```
# Cuffdiff example
data("df.cuff")
vsBoxPlot(
     data = df.cuff, d.factor = NULL, type = "cuffdiff", title = TRUE,
     legend = TRUE, grid = TRUE
)

# DESeq2 example
data("df.deseq")
require(DESeq2)
vsBoxPlot(
     data = df.deseq, d.factor = "condition", type = "deseq",
     title = TRUE, legend = TRUE, grid = TRUE
)

# edgeR example
data("df.edger")
require(edgeR)
vsBoxPlot(
     data = df.edger, d.factor = NULL, type = "edger", title = TRUE,
     legend = TRUE, grid = TRUE
)

# Display different colors for plot
data("df.edger")
vsBoxPlot(
     data = df.edger, d.factor = NULL, type = "edger", title = TRUE,
     legend = TRUE, grid = TRUE, fill.color = "RdGy",
     data.return = FALSE
)

# Extract data frame from visualization
```

```
data("df.edger")
require(edgeR)
tmp <- vsBoxPlot(
     data = df.edger, d.factor = NULL, type = "edger", title = TRUE,
     legend = TRUE, grid = TRUE, data.return = FALSE
)
df_box <- tmp[[1]] ## or use tmp$data
head(df_box)

# Show plot from object (see prior example for more details)
tmp[[2]] ## or use tmp$plot
```

`vsDEGMatrix` *Differential gene expression matrix*

Description

This function allows you to visualize the number of differentially expressed genes (DEG) at a given adjusted P-value for each experimental treatment level. Higher color intensity correlates to a higher number of DEGs.

Usage

```
vsDEGMatrix(data, padj = 0.05, d.factor = NULL, type = c("cuffdiff",
  "deseq", "edger"), title = TRUE, legend = TRUE, grid = TRUE,
  data.return = FALSE, grey.scale = FALSE, xaxis.text.size = 10,
  yaxis.text.size = 10, main.title.size = 15, legend.text.size = 10,
  legend.title.size = 12)
```

Arguments

`data`	output generated from calling the main routines of either `cuffdiff`, `DESeq2`, or `edgeR` analyses. For `cuffdiff`, this will be a `*_exp.diff` file. For `DESeq2`, this will be a generated object of class `DESeqDataSet`. For `edgeR`, this will be a generated object of class `DGEList`.
`padj`	a user defined adjusted p-value cutoff point. Defaults to `0.05`.
`d.factor`	a specified factor; for use with `DESeq2` objects only. This input equates to the first parameter for the contrast argument when invoking the `results()` function in `DESeq2`. Defaults to `NULL`
`type`	an analysis classifier to tell the function how to process the data. Must be either `cuffdiff`, `deseq`, or `edger`. `cuffdiff` must be used with `cuffdiff` data; `deseq` must be used for `DESeq2` output; `edgeR` must be used with `edgeR` data. See the `data` parameter for further details.
`title`	display the main title of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no title will display in plot.
`legend`	display legend of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no legend will display in plot.
`grid`	display major and minor axis lines. Logical; defaults to `TRUE`. If set to `FALSE`, no axis lines will display in plot.

`data.return`	returns data output of plot. Logical; defaults to `FALSE`. If set to `TRUE`, a data frame will also be called. Assign to object for reproduction and saving of data frame. See final example for further details.
`grey.scale`	displays grey color scheme instead of blue if set to `TRUE`. Logical; defaults to `FALSE`.
`xaxis.text.size`	change the font size of the `x`-axis text. Defaults to `10`.
`yaxis.text.size`	change the font size of the `y`-axis text. Defaults to `10`.
`main.title.size`	change the font size of the plot title text. Defaults to `15`.
`legend.text.size`	change the font size of the legend body text. Defaults to `10`.
`legend.title.size`	change the font size of the legend title text. Defaults to `12`.

Value

An object created by `ggplot`

Author(s)

Brandon Monier, `<brandon.monier@sdstate.edu>`

Examples

```
# cuffdiff example
data("df.cuff")
vsDEGMatrix(
     df.cuff, padj = 0.05, d.factor = NULL, type = "cuffdiff",
     title = TRUE, legend = TRUE, grid = TRUE
)

# DESeq2 example
data("df.deseq")
require(DESeq2)
vsDEGMatrix(
     df.deseq, padj = 0.05, d.factor = "condition", type = "deseq",
     title = TRUE, legend = TRUE, grid = TRUE
)

# edgeR example
data("df.edger")
require(edgeR)
vsDEGMatrix(
     df.edger, padj = 0.05, d.factor = NULL, type = "edger",
     title = TRUE, legend = TRUE, grid = TRUE
)

# Extract data frame from visualization
data("df.edger")
require(edgeR)
tmp <- vsDEGMatrix(
     df.edger, padj = 0.05, d.factor = NULL, type = "edger",
```

```
    title = TRUE, legend = TRUE, grid = TRUE
)
df_deg <- tmp[[1]] ## or use tmp$data
head(df_deg)

# Show plot from object (see prior example for more details)
tmp[[2]] ## or use tmp$plot
```

`vsFourWay` *Four-Way plot for comparison of log fold changes in a multiple factor RNA-seq experiment from different analytical objects*

Description

This function allows you to extract necessary results-based data from a DESeq object class to create a four-way plot to compare log fold changes in various treatments using ggplot2 aesthetics.

Usage

```
vsFourWay(x, y, control, data, d.factor = NULL, type = c("cuffdiff",
  "deseq", "edger"), padj = 0.05, x.lim = NULL, y.lim = NULL,
  lfc = NULL, legend = TRUE, title = TRUE, grid = TRUE,
  highlight = NULL, data.return = FALSE, xaxis.text.size = 10,
  yaxis.text.size = 10, xaxis.title.size = 10, yaxis.title.size = 10,
  main.title.size = 15, legend.text.size = 9)
```

Arguments

`x`	treatment `x` for comparison (log_2(x/control)). This will be a factor level in your data.
`y`	treatment `y` for comparison (log_2(y/control)). This will be a factor level in your data.
`control`	`control` treatment for comparisons of the `x` and `y` axes. This will be a factor level in your data.
`data`	output generated from calling the main routines of either `cuffdiff`, `DESeq2`, or `edgeR` analyses. For `cuffdiff`, this will be a `*_exp.diff` file. For `DESeq2`, this will be a generated object of class `DESeqDataSet`. For `edgeR`, this will be a generated object of class `DGEList`.
`d.factor`	a specified factor; for use with `DESeq2` objects only. This input equates to the first parameter for the contrast argument when invoking the `results()` function in `DESeq2`. Defaults to `NULL`.
`type`	an analysis classifier to tell the function how to process the data. Must be either `cuffdiff`, `deseq`, or `edger`. `cuffdiff` must be used with `cuffdiff` data; `deseq` must be used for `DESeq2` output; `edgeR` must be used with `edgeR` data. See the `data` parameter for further details.
`padj`	a user defined adjusted p-value cutoff point. Defaults to `0.05`.
`x.lim`	set manual limits (boundaries) to the x axis. Defaults to `NULL`.
`y.lim`	set manual limits (boundaries) to the y axis. Defaults to `NULL`.

`lfc`	log fold change level for setting conditonals. If no user input is added (`NULL`), value defaults to `1`.
`legend`	display legend of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no legend will display in plot.
`title`	display the main title of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no title will display in plot.
`grid`	display major and minor axis lines. Logical; defaults to `TRUE`. If set to `FALSE`, no axis lines will display in plot.
`highlight`	character string of IDs that will be highlighted. Set to `NULL` if you do not want highlighted data.
`data.return`	returns data output of plot. Logical; defaults to `FALSE`. If set to `TRUE`, a data frame will also be called. Assign to object for reproduction and saving of data frame. See final example for further details.
`xaxis.text.size`	change the font size of the `x`-axis text. Defaults to `10`.
`yaxis.text.size`	change the font size of the `y`-axis text. Defaults to `10`.
`xaxis.title.size`	change the font size of the `x`-axis title text. Defaults to `10`.
`yaxis.title.size`	change the font size of the `y`-axis title text. Defaults to `10`.
`main.title.size`	change the font size of the plot title text. Defaults to `15`.
`legend.text.size`	change the font size of the legend body text. Defaults to `9`.

Details

This function allows the user to extract various elements from a different analytical object class which in turn, creates a temporary data frame to plot the necessary ggplot aesthetics. In order for this function to work, RNA seq experiments must have multiple factors (i.e. two treatments and a control) and levels including treatments and controls. By having the recommended criteria, this function will extract the necessary data dependent on the analysis performed. Data points with "extreme" values that exceed the default viewing frame of the plot will change character classes (i.e. points of interest a substantially large log fold change).

Value

An object created by `ggplot`

Author(s)

Brandon Monier, `<brandon.monier@sdstate.edu>`

Examples

```
# Cuffdiff example
data("df.cuff")
vsFourWay(
     x = "hESC", y = "iPS", control = "Fibroblasts", data = df.cuff,
     d.factor = NULL, type = "cuffdiff", padj = 0.05, x.lim = NULL,
```

```
     y.lim = NULL, lfc = 2, title = TRUE, grid = TRUE,
     data.return = FALSE
)

# DESeq2 example
data("df.deseq")
vsFourWay(
     x = "treated_paired.end", y = "untreated_paired.end",
     control = "untreated_single.read", data = df.deseq,
     d.factor = "condition", type = "deseq", padj = 0.05,
     x.lim = NULL, y.lim = NULL, lfc = 2, title = TRUE, grid = TRUE,
     data.return = FALSE
)

# edgeR example
data("df.edger")
require(edgeR)
vsFourWay(
     x = "WM", y = "WW", control = "MM", data = df.edger,
     d.factor = NULL, type = "edger", padj = 0.05, x.lim = NULL,
     y.lim = NULL, lfc = 2, title = TRUE, grid = TRUE,
     data.return = FALSE
)

# Highlight IDs
data("df.edger")
require(edgeR)
hl <- c(
    "ID_639",
    "ID_518",
    "ID_602",
    "ID_449",
    "ID_076"
)
vsFourWay(
    x = "WM", y = "WW", control = "MM", data = df.edger,
    d.factor = NULL, type = "edger", padj = 0.05, x.lim = NULL,
    y.lim = NULL, lfc = 2, title = TRUE, grid = TRUE,
    data.return = FALSE, highlight = hl
)

# Extract data frame from visualization
data("df.cuff")
tmp <- vsFourWay(
     x = "WM", y = "WW", control = "MM", data = df.edger,
     d.factor = NULL, type = "edger", padj = 0.05,
     x.lim = NULL, y.lim = NULL, lfc = 2, title = TRUE,
     grid = TRUE, data.return = TRUE
)
df_four <- tmp[[1]] ## or use tmp$data
head(df_four)

# Show plot from object (see prior example for more details)
tmp[[2]] ## or use tmp$plot
```

`vsMAMatrix` *MA plot matrix from log_2 fold changes and $-log\_10$(p-values)*

Description

This function allows you to generate MA plots for all possible treatment combinations for a given factor in either a cuffdiff, DESeq2, or edgeR data set.

Usage

```
vsMAMatrix(data, d.factor = NULL, type = c("cuffdiff", "deseq", "edger"),
  padj = 0.05, y.lim = NULL, lfc = NULL, title = TRUE, legend = TRUE,
  grid = TRUE, counts = TRUE, data.return = FALSE, xaxis.text.size = 9,
  yaxis.text.size = 9, xaxis.title.size = 10, yaxis.title.size = 10,
  main.title.size = 15, legend.text.size = 9, facet.title.size = 10)
```

Arguments

`data`	output generated from calling the main routines of either `cuffdiff`, `DESeq2`, or `edgeR` analyses. For `cuffdiff`, this will be a `*_exp.diff` file. For `DESeq2`, this will be a generated object of class `DESeqDataSet`. For `edgeR`, this will be a generated object of class `DGEList`.
`d.factor`	a specified factor; for use with `DESeq2` objects only. This input equates to the first parameter for the contrast argument when invoking the `results()` function in `DESeq2`. Defaults to `NULL`.
`type`	an analysis classifier to tell the function how to process the data. Must be either `cuffdiff`, `deseq`, or `edger`. `cuffdiff` must be used with `cuffdiff` data; `deseq` must be used for `DESeq2` output; `edgeR` must be used with `edgeR` data. See the `data` parameter for further details.
`padj`	a user defined adjusted p-value cutoff point. Defaults to `0.05`.
`y.lim`	set manual limits (boundaries) to the y axis. Defaults to `NULL`.
`lfc`	log fold change level for setting conditonals. If no user input is added (`NULL`), value defaults to `1`.
`title`	display the main title of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no title will display in plot.
`legend`	display legend of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no legend will display in plot.
`grid`	display major and minor axis lines. Logical; defaults to `TRUE`. If set to `FALSE`, no axis lines will display in plot.
`counts`	displays the number of differentially expressed genes for each treatment comparison. Defaults to `TRUE`.
`data.return`	returns data output of plot. Logical; defaults to `FALSE`. If set to `TRUE`, a data frame will also be called. Assign to object for reproduction and saving of data frame. See final example for further details.
`xaxis.text.size`	change the font size of the `x`-axis text. Defaults to `9`.
`yaxis.text.size`	change the font size of the `y`-axis text. Defaults to `9`.

`xaxis.title.size`	change the font size of the `x`-axis title text. Defaults to `10`.
`yaxis.title.size`	change the font size of the `y`-axis title text. Defaults to `10`.
`main.title.size`	change the font size of the plot title text. Defaults to `15`.
`legend.text.size`	change the font size of the legend body text. Defaults to `9`.
`facet.title.size`	change the font size of the facet wrap title text. Defaults to `10`.

Value

An object created by `ggplot`

Author(s)

Brandon Monier, <`brandon.monier@sdstate.edu`>

Examples

```
# Cuffdiff example
data("df.cuff")
vsMAMatrix(
  data = df.cuff, d.factor = NULL, type = "cuffdiff",
  padj = 0.05, y.lim = NULL, lfc = 1, title = TRUE,
  grid = TRUE, counts = TRUE, data.return = FALSE
)

# DESeq2 example
data("df.deseq")
require(DESeq2)
vsMAMatrix(
  data = df.deseq, d.factor = "condition", type = "deseq",
  padj = 0.05, y.lim = NULL, lfc = 1, title = TRUE,
  grid = TRUE, counts = TRUE, data.return = FALSE
)

# edgeR example
data("df.edger")
require(edgeR)
vsMAMatrix(
  data = df.edger, d.factor = NULL, type = "edger",
  padj = 0.05, y.lim = NULL, lfc = 1, title = TRUE,
  grid = TRUE, counts = TRUE, data.return = FALSE
)

# Extract data frame from visualization
data("df.cuff")
tmp <- vsMAMatrix(
  data = df.cuff, d.factor = NULL,
  type = "cuffdiff", padj = 0.05, y.lim = NULL,
  lfc = 1, title = TRUE, grid = TRUE,
  counts = TRUE, data.return = TRUE
)
```

```
df_mamat <- tmp[[1]] ## or use tmp$data
head(df_mamat)

# Show plot from object (see prior example for more details)
tmp[[2]] ## or use tmp$plot
```

`vsMAPlot` *MA plot from mean expression and log fold changes from different analytical objects*

Description

This function allows you to extract necessary results-based data from different output files to create a MA plot (i.e. a scatter plot) of log_2 fold changes versus normalized mean counts while implementing ggplot2 aesthetics.

Usage

```
vsMAPlot(x, y, data, d.factor = NULL, type = c("cuffdiff", "deseq",
  "edger"), padj = 0.05, y.lim = NULL, lfc = NULL, title = TRUE,
  legend = TRUE, grid = TRUE, highlight = NULL, data.return = FALSE,
  xaxis.text.size = 10, yaxis.text.size = 10, xaxis.title.size = 10,
  yaxis.title.size = 10, main.title.size = 15, legend.text.size = 9)
```

Arguments

`x`	treatment `x` for comparison (log_2(x/control)). This will be a factor level in your data.
`y`	treatment `y` for comparison (log_2(y/control)). This will be a factor level in your data.
`data`	output generated from calling the main routines of either `cuffdiff`, `DESeq2`, or `edgeR` analyses. For `cuffdiff`, this will be a `*_exp.diff` file. For `DESeq2`, this will be a generated object of class `DESeqDataSet`. For `edgeR`, this will be a generated object of class `DGEList`.
`d.factor`	a specified factor; for use with `DESeq2` objects only. This input equates to the first parameter for the contrast argument when invoking the `results()` function in `DESeq2`. Defaults to `NULL`.
`type`	an analysis classifier to tell the function how to process the data. Must be either `cuffdiff`, `deseq`, or `edger`. `cuffdiff` must be used with `cuffdiff` data; `deseq` must be used for `DESeq2` output; `edgeR` must be used with `edgeR` data. See the `data` parameter for further details.
`padj`	a user defined adjusted p-value cutoff point. Defaults to `0.05`.
`y.lim`	set manual limits (boundaries) to the y axis. Defaults to `NULL`.
`lfc`	log fold change level for setting conditonals. If no user input is added (`NULL`), value defaults to `1`.
`title`	display the main title of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no title will display in plot.
`legend`	display legend of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no legend will display in plot.

`grid`	display major and minor axis lines. Logical; defaults to `TRUE`. If set to `FALSE`, no axis lines will display in plot.
`highlight`	character string of IDs that will be highlighted. Set to `NULL` if you do not want highlighted data.
`data.return`	returns data output of plot Logical; defaults to `FALSE`. If set to `TRUE`, a data frame will also be called. Assign to object for reproduction and saving of data frame. See final example for further details.
`xaxis.text.size`	change the font size of the `x`-axis text. Defaults to `10`.
`yaxis.text.size`	change the font size of the `y`-axis text. Defaults to `10`.
`xaxis.title.size`	change the font size of the `x`-axis title text. Defaults to `10`.
`yaxis.title.size`	change the font size of the `y`-axis title text. Defaults to `10`.
`main.title.size`	change the font size of the plot title text. Defaults to `15`.
`legend.text.size`	change the font size of the legend body text. Defaults to `9`.

Value

An object created by `ggplot`

Author(s)

Brandon Monier, `<brandon.monier@sdstate.edu>`

Examples

```
# Cuffdiff example
data("df.cuff")
vsMAPlot(
     x = "hESC", y = "iPS", data = df.cuff, d.factor = NULL,
     type = "cuffdiff", padj = 0.05, y.lim = NULL, lfc = 1,
     title = TRUE, legend = TRUE, grid = TRUE, data.return = FALSE
)

# DESeq2 example
data("df.deseq")
require(DESeq2)
vsMAPlot(
     x = "treated_paired.end", y = "untreated_paired.end",
     data = df.deseq, d.factor = "condition", type = "deseq",
     padj = 0.05, y.lim = NULL, lfc = NULL, title = TRUE,
     legend = TRUE, grid = TRUE, data.return = FALSE
)

# edgeR example
data("df.edger")
require(edgeR)
vsMAPlot(
     x = "WM", y = "MM", data = df.edger, d.factor = NULL,
```

```
     type = "edger", padj = 0.1, y.lim = NULL, lfc = 1,
     title = FALSE, legend = TRUE, grid = TRUE, data.return = FALSE
)

# Highlight IDs
data("df.deseq")
require(DESeq2)
hl <- c(
    "FBgn0022201",
    "FBgn0003042",
    "FBgn0031957",
    "FBgn0033853",
    "FBgn0003371"
)
vsMAPlot(
     x = "treated_paired.end", y = "untreated_paired.end",
     data = df.deseq, d.factor = "condition", type = "deseq",
     padj = 0.05, y.lim = NULL, lfc = NULL, title = TRUE,
     legend = TRUE, grid = TRUE, data.return = FALSE, highlight = hl
)

# Extract data frame from visualization
data("df.cuff")
tmp <- vsMAPlot(
     x = "hESC", y = "iPS", data = df.cuff,
     d.factor = NULL, type = "cuffdiff", padj = 0.05,
     y.lim = NULL, lfc = 1, title = TRUE, grid = TRUE,
     data.return = TRUE
)
df_ma <- tmp[[1]] ## or use tmp$data
head(df_ma)

# Show plot from object (see prior example for more details)
tmp[[2]] ## or use tmp$plot
```

`vsScatterMatrix` *Scatter plot matrix of log_10(FPKM or CPM) values*

Description

This function will generate a matrix of scatterplots for all possible treatment combinations with additional distribution info.

Usage

```
vsScatterMatrix(data, d.factor = NULL, type = c("cuffdiff", "deseq",
  "edger"), comp = NULL, title = TRUE, grid = TRUE, man.title = NULL,
  data.return = FALSE, xaxis.text.size = 9, yaxis.text.size = 9,
  xaxis.title.size = 10, yaxis.title.size = 10, main.title.size = 15,
  facet.title.size = 10)
```

Arguments

`data`	output generated from calling the main routines of either `cuffdiff`, `DESeq2`, or `edgeR` analyses. For `cuffdiff`, this will be a `*_exp.diff` file. For `DESeq2`, this will be a generated object of class `DESeqDataSet`. For `edgeR`, this will be a generated object of class `DGEList`.
`d.factor`	a specified factor; for use with `DESeq2` objects only. This input equates to the first parameter for the contrast argument when invoking the `results()` function in `DESeq2`. Defaults to `NULL`.
`type`	an analysis classifier to tell the function how to process the data. Must be either `cuffdiff`, `deseq`, or `edger`. `cuffdiff` must be used with `cuffdiff` data; `deseq` must be used for `DESeq2` output; `edgeR` must be used with `edgeR` data. See the `data` parameter for further details.
`comp`	treatments you would like to compare in the form of a vector. If no parameter is specified, all possible treatment comparisons will be made. Defaults to `NULL`.
`title`	display the main title of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no title will display in plot.
`grid`	display major and minor axis lines. Logical; defaults to `TRUE`. If set to `FALSE`, no axis lines will display in plot.
`man.title`	a manually specified title at the authors discretion. Defaults to `NULL`.
`data.return`	returns data output of plot. Logical; defaults to `FALSE`. If set to `TRUE`, a data frame will also be called. Assign to object for reproduction and saving of data frame. See final example for further details.
`xaxis.text.size`	change the font size of the `x`-axis text. Defaults to `9`.
`yaxis.text.size`	change the font size of the `y`-axis text. Defaults to `9`.
`xaxis.title.size`	change the font size of the `x`-axis title text. Defaults to `10`.
`yaxis.title.size`	change the font size of the `y`-axis title text. Defaults to `10`.
`main.title.size`	change the font size of the plot title text. Defaults to `15`.
`facet.title.size`	change the font size of the facet wrap title text. Defaults to `10`.

Value

An object created by `ggplot`

Author(s)

Brandon Monier, `<brandon.monier@sdstate.edu>`

Examples

```
# Cuffdiff example
data("df.cuff")
vsScatterMatrix(
     data = df.cuff, d.factor = NULL, type = "cuffdiff",
     comp = NULL, title = TRUE, grid = TRUE,
```

```
        man.title = "Example title"
)

# DESeq2 example
data("df.deseq")
require(DESeq2)
vsScatterMatrix(
        data = df.deseq, d.factor = "condition", type = "deseq",
        comp = NULL, title = TRUE, grid = FALSE, man.title = NULL
)

# edgeR example
data("df.edger")
require(edgeR)
vsScatterMatrix(
        data = df.edger, d.factor = NULL, type = "edger",
        comp = c("WM", "MM"), title = TRUE, grid = TRUE,
        man.title = NULL
)

# Extract data frame from visualization
data("df.edger")
tmp <- vsScatterMatrix(
        data = df.edger, d.factor = NULL, type = "edger",
        comp = c("WM", "MM"), title = TRUE, grid = TRUE,
        man.title = NULL, data.return = TRUE
)
df_scatmat <- tmp[[1]] ## or use tmp$data
head(df_scatmat)

# Show plot from object (see prior example for more details)
tmp[[2]] ## or use tmp$plot
```

`vsScatterPlot` *Scatter plot of log_10(FPKM or CPM) values*

Description

This function allows you to visualize comparisons of log_{10} values of either FPKM or CPM measurements of two treatments depending on analytical type.

Usage

```
vsScatterPlot(x, y, data, d.factor = NULL, type = c("cuffdiff", "deseq",
  "edger"), title = TRUE, grid = TRUE, highlight = NULL,
  data.return = FALSE, xaxis.text.size = 10, yaxis.text.size = 10,
  xaxis.title.size = 12, yaxis.title.size = 12, main.title.size = 15)
```

Arguments

`x`	treatment `x` for comparison (log_2(x/control)). This will be a factor level in your data.
`y`	treatment `y` for comparison (log_2(y/control)). This will be a factor level in your data.

`data`	output generated from calling the main routines of either `cuffdiff`, `DESeq2`, or `edgeR` analyses. For `cuffdiff`, this will be a `*_exp.diff` file. For `DESeq2`, this will be a generated object of class `DESeqDataSet`. For `edgeR`, this will be a generated object of class `DGEList`.
`d.factor`	a specified factor; for use with `DESeq2` objects only. This input equates to the first parameter for the contrast argument when invoking the `results()` function in `DESeq2`. Defaults to `NULL`.
`type`	an analysis classifier to tell the function how to process the data. Must be either `cuffdiff`, `deseq`, or `edger`. `cuffdiff` must be used with `cuffdiff` data; `deseq` must be used for `DESeq2` output; `edgeR` must be used with `edgeR` data. See the `data` parameter for further details.
`title`	display the main title of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no title will display in plot.
`grid`	display major and minor axis lines. Logical; defaults to `TRUE`. If set to `FALSE`, no axis lines will display in plot.
`highlight`	character string of IDs that will be highlighted. Set to `NULL` if you do not want highlighted data.
`data.return`	returns data output of plot Logical; defaults to `FALSE`. If set to `TRUE`, a data frame will also be called. Assign to object for reproduction and saving of data frame. See final example for further details.
`xaxis.text.size`	change the font size of the `x`-axis text. Defaults to `10`.
`yaxis.text.size`	change the font size of the `y`-axis text. Defaults to `10`.
`xaxis.title.size`	change the font size of the `x`-axis title text. Defaults to `12`.
`yaxis.title.size`	change the font size of the `y`-axis title text. Defaults to `12`.
`main.title.size`	change the font size of the plot title text. Defaults to `15`.

Value

An object created by `ggplot`

Author(s)

Brandon Monier, `<brandon.monier@sdstate.edu>`

Examples

```
# Cuffdiff example
data("df.cuff")
vsScatterPlot(
     x = "hESC", y = "iPS", data = df.cuff, d.factor = NULL,
     type = "cuffdiff", title = TRUE, grid = TRUE
)

# DESeq2 example
data("df.deseq")
require(DESeq2)
```

```
vsScatterPlot(
     x = "treated_paired.end", y = "untreated_paired.end",
     data = df.deseq, d.factor = "condition", type = "deseq",
     title = TRUE, grid = TRUE
)

# edgeR example
data("df.edger")
require(edgeR)
vsScatterPlot(
     x = "WW", y = "WM", data = df.edger, d.factor = NULL,
     type = "edger", title = TRUE, grid = TRUE
)

# Highlight IDs
data("df.cuff")
hl <- c(
     "XLOC_000033",
     "XLOC_000099",
     "XLOC_001414",
     "XLOC_001409"
)
vsScatterPlot(
     x = "hESC", y = "iPS", data = df.cuff, d.factor = NULL,
     type = "cuffdiff", title = TRUE, grid = TRUE, highlight = hl
)

# Extract data frame from visualization
data("df.cuff")
tmp <- vsScatterPlot(
     x = "hESC", y = "iPS", data = df.cuff, d.factor = NULL,
     type = "cuffdiff", title = TRUE, grid = TRUE, data.return = TRUE
)
df_scatter <- tmp[[1]] ## or use tmp$data
head(df_scatter)

# Show plot from object (see prior example for more details)
tmp[[2]] ## or use tmp$plot
```

`vsVolcano` *Volcano plot from log_2 fold changes and $-log\_10$(p-values)*

Description

This function allows you to extract necessary results-based data from either a `DESeq2` object, edgeR object, or `cuffdiff` data frame to create a volcano plot (i.e. a scatter plot) of the negative log of the p-value versus the log of the fold change while implementing ggplot2 aesthetics.

Usage

```
vsVolcano(x, y, data, d.factor = NULL, type = c("cuffdiff", "deseq",
  "edger"), padj = 0.05, x.lim = NULL, lfc = NULL, title = TRUE,
  legend = TRUE, grid = TRUE, highlight = NULL, data.return = FALSE,
  xaxis.text.size = 10, yaxis.text.size = 10, xaxis.title.size = 10,
  yaxis.title.size = 10, main.title.size = 15, legend.text.size = 9)
```

Arguments

`x`	treatment `x` for comparison (log_2(x/control)). This will be a factor level in your data.
`y`	treatment `y` for comparison (log_2(y/control)). This will be a factor level in your data.
`data`	output generated from calling the main routines of either `cuffdiff`, `DESeq2`, or `edgeR` analyses. For `cuffdiff`, this will be a `*_exp.diff` file. For `DESeq2`, this will be a generated object of class `DESeqDataSet`. For `edgeR`, this will be a generated object of class `DGEList`.
`d.factor`	a specified factor; for use with `DESeq2` objects only. This input equates to the first parameter for the contrast argument when invoking the `results()` function in `DESeq2`. Defaults to `NULL`.
`type`	an analysis classifier to tell the function how to process the data. Must be either `cuffdiff`, `deseq`, or `edger`. `cuffdiff` must be used with `cuffdiff` data; `deseq` must be used for `DESeq2` output; `edgeR` must be used with `edgeR` data. See the `data` parameter for further details.
`padj`	a user defined adjusted p-value cutoff point. Defaults to `0.05`.
`x.lim`	set manual limits (boundaries) to the x axis. Defaults to `NULL`.
`lfc`	log fold change level for setting conditonals. If no user input is added (`NULL`), value defaults to `1`.
`title`	display the main title of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no title will display in plot.
`legend`	display legend of plot. Logical; defaults to `TRUE`. If set to `FALSE`, no legend will display in plot.
`grid`	display major and minor axis lines. Logical; defaults to `TRUE`. If set to `FALSE`, no axis lines will display in plot.
`highlight`	character string of IDs that will be highlighted. Set to `NULL` if you do not want highlighted data.
`data.return`	returns data output of plot Logical; defaults to `FALSE`. If set to `TRUE`, a data frame will also be called. Assign to object for reproduction and saving of data frame. See final example for further details.
`xaxis.text.size`	change the font size of the `x`-axis text. Defaults to `10`.
`yaxis.text.size`	change the font size of the `y`-axis text. Defaults to `10`.
`xaxis.title.size`	change the font size of the `x`-axis title text. Defaults to `10`.
`yaxis.title.size`	change the font size of the `y`-axis title text. Defaults to `10`.
`main.title.size`	change the font size of the plot title text. Defaults to `15`.
`legend.text.size`	change the font size of the legend body text. Defaults to `9`.

Value

An object created by `ggplot`

Author(s)

Brandon Monier, `<brandon.monier@sdstate.edu>`

Examples

```
# Cuffdiff example
data("df.cuff")
vsVolcano(
     x = "hESC", y = "iPS", data = df.cuff, d.factor = NULL,
     type = "cuffdiff", padj = 0.05, x.lim = NULL, lfc = 2,
     title = TRUE, grid = TRUE, data.return = FALSE
)

# DESeq2 example
data("df.deseq")
require(DESeq2)
vsVolcano(
     x = "treated_paired.end", y = "untreated_paired.end",
     data = df.deseq, d.factor = "condition",
     type = "deseq", padj = 0.05, x.lim = NULL, lfc = NULL,
     title = TRUE, grid = TRUE, data.return = FALSE
)

# edgeR example
data("df.edger")
require(edgeR)
vsVolcano(
     x = "WM", y = "MM", data = df.edger, d.factor = NULL,
     type = "edger", padj = 0.1, x.lim = NULL, lfc = 2,
     title = FALSE, grid = TRUE, data.return = FALSE
)

data("df.deseq")
require(DESeq2)
hl <- c(
    "FBgn0036248",
    "FBgn0026573",
    "FBgn0259742",
    "FBgn0038961",
    "FBgn0038928"
)
vsVolcano(
    x = "treated_paired.end", y = "untreated_paired.end",
    data = df.deseq, d.factor = "condition",
    type = "deseq", padj = 0.05, x.lim = NULL, lfc = NULL,
    title = TRUE, grid = TRUE, data.return = FALSE, highlight = hl
)

# Extract data frame from visualization
data("df.cuff")
tmp <- vsVolcano(
     x = "hESC", y = "iPS", data = df.cuff,
     d.factor = NULL, type = "cuffdiff", padj = 0.05,
     x.lim = NULL, lfc = 2, title = TRUE, grid = TRUE,
     data.return = TRUE
)
```

```
df_volcano <- tmp[[1]] ## or use tmp$data
head(df_volcano)

# Show plot from object (see prior example for more details)
tmp[[2]] ## or use tmp$plot
```

`vsVolcanoMatrix` *Volcano plot matrix from log_2 fold changes and $-log\_10$(p-values)*

Description

This function allows you to extract necessary results-based data from either a `DESeq2` object, `edgeR` object, or `cuffdiff` data frame to create a volcano plot (i.e. a scatter plot) of the negative log of the p-value versus the log of the fold change while implementing ggplot2 aesthetics for all possible combinations of treatments.

Usage

```
vsVolcanoMatrix(data, d.factor = NULL, type = c("cuffdiff", "deseq",
  "edger"), padj = 0.05, x.lim = NULL, lfc = NULL, title = TRUE,
  legend = TRUE, grid = TRUE, counts = TRUE, data.return = FALSE,
  xaxis.text.size = 9, yaxis.text.size = 9, xaxis.title.size = 10,
  yaxis.title.size = 10, main.title.size = 15, legend.text.size = 9,
  facet.title.size = 10)
```

Arguments

`data`	a cuffdiff, DESeq2, or edgeR object.
`d.factor`	a specified factor; for use with DESeq2 objects only. Defaults to `NULL`
`type`	an analysis classifier to tell the function how to process the data. Must be either `cuffdiff`, `deseq`, or `edgeR`.
`padj`	a user defined adjusted p-value cutoff point. Defaults to `0.05`.
`x.lim`	set manual limits to the x axis. Defaults to `NULL`.
`lfc`	log fold change level for setting conditonals. If no user input is added (`NULL`), value defaults to `1`.
`title`	show title of plot. Defaults to `TRUE`.
`legend`	shows legend of plot. Defaults to `TRUE`.
`grid`	show major and minor axis lines. Defaults to `TRUE`.
`counts`	displays the number of differentially expressed genes for each treatment comparison. Defaults to `TRUE`.
`data.return`	returns data output of plot if set to `TRUE`. Defaults to `FALSE`.
`xaxis.text.size`	change the font size of the `x`-axis text. Defaults to `9`.
`yaxis.text.size`	change the font size of the `y`-axis text. Defaults to `9`.
`xaxis.title.size`	change the font size of the `x`-axis title text. Defaults to `10`.

`yaxis.title.size`
: change the font size of the y-axis title text. Defaults to `10`.

`main.title.size`
: change the font size of the plot title text. Defaults to `15`.

`legend.text.size`
: change the font size of the legend body text. Defaults to `9`.

`facet.title.size`
: change the font size of the facet wrap title text. Defaults to `10`.

Value

An object created by `ggplot`

Author(s)

Brandon Monier, `<brandon.monier@sdstate.edu>`

Examples

```
# Cuffdiff example
data("df.cuff")
vsVolcanoMatrix(
     data = df.cuff, d.factor = NULL, type = "cuffdiff",
     padj = 0.05, x.lim = NULL, lfc = 2, title = TRUE,
     grid = TRUE, counts = TRUE, data.return = FALSE
)

# DESeq2 example
data("df.deseq")
require(DESeq2)
vsVolcanoMatrix(
     data = df.deseq, d.factor = "condition", type = "deseq",
     padj = 0.05, x.lim = NULL, lfc = 2, title = TRUE,
     grid = TRUE, counts = TRUE, data.return = FALSE
)

# edgeR example
data("df.edger")
require(edgeR)
vsVolcanoMatrix(
     data = df.edger, d.factor = NULL, type = "edger",
     padj = 0.05, x.lim = NULL, lfc = 2, title = TRUE,
     grid = TRUE, counts = TRUE, data.return = FALSE
)

# Extract data frame from visualization
data("df.cuff")
tmp <- vsVolcanoMatrix(
     data = df.cuff, d.factor = NULL,
     type = "cuffdiff", padj = 0.05, x.lim = NULL,
     lfc = 2, title = TRUE, grid = TRUE,
     counts = TRUE, data.return = TRUE
)
df_vmat <- tmp[[1]]
head(df_vmat)
```

```
# Show plot from object (see prior example for more details)
tmp[[2]] ## or use tmp$plot
```

Index