Package ‘RJSplot’

May 7, 2026

Type Package

Version 2.7

Date 2021-11-30

Title Interactive Graphs with R

Description Creates interactive graphs with 'R'. It joins the data analysis power of R and the visualization libraries of JavaScript in one package.

URL `https://rjsplot.usal.es`

Depends R (>= 3.0.0)

Suggests IRdisplay

NeedsCompilation no

Author Carlos Prieto <`cprietos@usal.es`>,
David Barrios <`metal@usal.es`>

Maintainer Carlos Prieto <`cprietos@usal.es`>

Repository CRAN

Date/Publication 2021-12-01 08:40:02 UTC

Contents

`barplot_rjs` *Produce interactive bar plot(s).*

Description

`barplot_rjs` produce interactive bar plot(s) of the given values.

Usage

```
barplot_rjs(height, xlab = "", ylab = "", ylim = NULL, cex = 1,
            plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`height`	either a vector or matrix of values describing the bars which make up the plot.
`xlab`	a title for the x axis.
`ylab`	a title for the y axis.
`ylim`	a vector with two values which specifies the Y minimun and maximun values in the representation.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive barplot in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es
if(interactive()){
barplot_rjs(USArrests, "states", "arrests")
}
```

`boxplot_rjs` *Produce interactive box-and-whisker plot(s).*

Description

`boxplot_rjs` produce interactive box-and-whisker plot(s) of the given values.

Usage

```
boxplot_rjs(data, xlab = "", ylab = "",
            outline = TRUE, col = NULL, ylim = NULL, cex = 1,
            plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`data`	a Data Frame or Matrix from which the data for each box will be taken.
`outline`	if `outline` is not true, the outliers are not drawn.
`col`	if 'col' is non-null it is assumed to contain colors to be used to colour the bodies of the box plots.
`xlab`	a title for the x axis.
`ylab`	a title for the y axis.

`ylim`	a vector with two values which specifies the Y minimun and maximun values in the representation.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive boxplot in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es
if(interactive()){
boxplot_rjs(attitude, "attitude", "favourable responses")
}
```

`bubbles_rjs` *Produce interactive bubble plots.*

Description

`bubbles_rjs` produce interactive bubble plot(s) of the given values.

Usage

```
bubbles_rjs(data, size = NULL, plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`data`	a numeric Data Frame or Matrix with two columns and one row per bubble which contains represented values for each bubble. Row names will be used as bubble names.
`size`	a vector giving each bubble size.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive barplot in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es
if(interactive()){
bubbles_rjs(scale(mtcars[,c("mpg","hp")],FALSE),mtcars$wt)
}
```

`circularfv_rjs` *Pruduce a circular sequence viewer.*

Description

`circularfv_rjs` creates a circular sequence viewer.

Usage

```
circularfv_rjs(sequence, features = FALSE,
               plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`sequence`	a "character" string giving the DNA sequence.
`features`	Data Frame with the segments that will be seen in the viewer. It cotains 5 columns which describes the identifier, start coordinate, stop coordinate, segment type and segment color.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
if(interactive()){
## Creates a circular sequence viewer in a temporal directory of your local machine
## Generate test input data
sequence <- paste(sample(c("c","a","t","g"), 10000, replace = TRUE), collapse = "")
data <- matrix(round(runif(12,0,10000)),6,2)
features <- data.frame(id = 0:5, start = data[,1], stop = data[,2],
type = c("voluptate","non","voluptate","sit","et","proident"),
color = c("SandyBrown","Tan","Wheat","SteelBlue","LightSkyBlue","LightBlue"))

## Create the circular sequence viewver
circularfv_rjs(sequence, features)
}
```

`createAssembly` *Creates a genome assembly for* `genomemap_rjs` *or* `manhattan_rjs`.

Description

`createAssembly` create assemblies for their use as parameter of `genomemap_rjs` or `manhattan_rjs`.

Usage

```
createAssembly(name, size)
```

Arguments

`name`	a vector with the chomosome or scaffold names.
`size`	a vector with the chomosome or scaffold sizes.

Value

a data frame in BED format to be used as an assembly in `genomemap_rjs` or `manhattan_rjs`.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: `https://rjsplot.usal.es`

`genomemap_rjs`.

`manhattan_rjs`.

Examples

```
## Creates a fake genome assembly for its posterior use as parameter of genomemap or manhattan plot
## Live examples and full tutorial on https://rjsplot.usal.es

## Generate a test assembly with three scaffolds
assembly <- createAssembly(c("scaffold1","scaffold2","scaffold3"), c(1000000,800000,650000))
```

`dendrogram_rjs` *Draw a Dendrogram.*

Description

`dendrogram_rjs` creates a dendrogram.

Usage

```
dendrogram_rjs(data, metadata = NULL, method = "complete", cex = 1,
               plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`data`	a dissimilarity object of class dist (p.e. produced by function dist).
`metadata`	a data frame with metadata information for leaf features (p.e. sample phenotypes). Corresponding Leaf/Sample should be provided as row name.
`method`	Name of the agglomeration method to be used. It should be (an unambiguous abbreviation of) one of "ward.D", "ward.D2", "single", "complete", "average" (= UPGMA), "mcquitty" (= WPGMA), "median" (= WPGMC) or "centroid" (= UPGMC).
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	A "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create a dendrogram in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es
if(interactive()){
dendrogram_rjs(dist(USArrests),USArrests)
}
```

`densityplot_rjs` *Draw an interactive density plot.*

Description

`densityplot_rjs` creates density plots.

Usage

```
densityplot_rjs(data, xlab = "", ylab = "", cex = 1,
                plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`data`	a data frame from which distribution will be represented for each column.
`xlab`	a title for the x axis.
`ylab`	a title for the y axis.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Creates a density plot in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

if(interactive()){
## Generate test input data
data <- data.frame(Uni05 = (1:100)/21, Norm = rnorm(100), `5T` = rt(100, df = 5),
Gam2 = rgamma(100, shape = 2))

# Create the density plot
densityplot_rjs(data, "x", "y")
}
```

`genomemap_rjs` *Create an interative genome map.*

Description

`genomemap_rjs` creates an interactive genome map.

Usage

```
genomemap_rjs(assembly, track = NULL, plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`assembly`	a genome assembly data frame. RJSplot provides human assemblies (NCBI36, GRCh37, GRCh38), human assemblies with cytobands (GRCh37.bands or GRCh38.bands), or methods for creating your own assembly from a FASTA file `getAssemblyFromFasta`.
`track`	a data frame with values to represent on the genome map (chromosome, position, value).
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with Firefox. If you want to open this file stored in your local machine with other Web browser, please visit the help section on the RJSplot Web site (`https://rjsplot.usal.es`).

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: https://rjsplot.usal.es

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive Genome map in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

if(interactive()){
# Create test data
chr <- character()
pos <- numeric()

for(i in 1:nrow(GRCh38)){
  chr <- c(chr,as.character(rep(GRCh38[i,"chr"],100)))
  pos <- c(pos,sample(GRCh38[i,"start"]:GRCh38[i,"end"],100))
}

value <- round(rexp(length(pos)),2)

# Create a genome map
track <- data.frame(chr,pos,pos+1,NA,value)
genomemap_rjs(GRCh38.bands, track)
}
```

`getAssemblyFromFasta` *Create an assembly from a FASTA file for its use as parameter of* `genomemap_rjs` *or* `manhattan_rjs`.

Description

`getAssemblyFromFasta` create assemblies from a FASTA file for their use as parameter of `genomemap_rjs` or `manhattan_rjs`.

Usage

```
getAssemblyFromFasta(fasta)
```

Arguments

`fasta`	a "character" string representing the input Fasta file to be added in the genome browser.

Value

a data frame in BED format ready to use as an assembly in `genomemap_rjs` or `manhattan_rjs`.

Note

FASTA format is a text-based format for representing either nucleotide sequences or peptide sequences, in which nucleotides or amino acids are represented using single-letter codes.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: `https://rjsplot.usal.es`

`genomemap_rjs`.

`manhattan_rjs`.

Examples

```
## Live examples and full tutorial on https://rjsplot.usal.es
```

`GRCh37` *Length of human chromosomes based on GRCh37 assembly*

Description

Length of human chromosomes based on GRCh37 assembly.

Source

See `https://www.ncbi.nlm.nih.gov/assembly/GCF_000001405.13/`.

`GRCh37.bands` *Cytoband information based on GRCh37*

Description

A Data Frame containing cytobands of GRCh37 assembly.

Source

See `http://grch37.ensembl.org/Homo_sapiens/Location/Genome`.

`GRCh38` *Length of human chromosomes based on GRCh38 assembly*

Description

Length of human chromosomes based on GRCh38 assembly.

Source

See https://www.ncbi.nlm.nih.gov/assembly/883148/.

`GRCh38.bands` *Cytoband information based on GRCh38*

Description

A Data Frame containing cytobands of GRCh38 assembly.

Source

See http://www.ensembl.org/Homo_sapiens/Location/Genome.

`heatmap_rjs` *Produce an interactive heatmap.*

Description

`heatmap_rjs` creates interactive heatmaps.

Usage

```
heatmap_rjs(data, metadata = NULL, scale = c("row", "column", "none"),
  cluster = TRUE, color = c("Reds", "Greens", "Blues", "RdBkGr", "RdWhBu"),
  na.color = "transparent", cex = 1, plot = TRUE, jupyter = FALSE,
  dir = tempdir(), distfun = dist, hclustfun = hclust)
```

Arguments

`data`	numeric matrix of the values to be plotted.
`metadata`	a data frame with metadata information for column features (p.e. sample phenotypes). Samples are given in rows and features in columns.
`scale`	character indicating if the values should be centered and scaled in either the row direction or the column direction, or none. The default value is "none".
`cluster`	logical value to enable (`TRUE`) or disable (`FALSE`) data clustering.
`color`	character indicating the color scale to paint values. The default value is "Blues".
`na.color`	character indicating the color to use for missing values. Transparent by default.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.
`distfun`	function used to compute the distance between rows and columns.
`hclustfun`	function used to compute the hierarchical clustering.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive heatmap in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es
if(interactive()){
metadata <- data.frame(phenotype1 = sample(c("yes","no"),ncol(mtcars),TRUE),
                       phenotype2 = sample(1:5,ncol(mtcars),TRUE))

heatmap_rjs(data.matrix(mtcars), metadata, scale="column")
}
```

`hiveplot_rjs` *Produce interactive hive plot*

Description

`hiveplot_rjs` creates an interactive hive plot.

Usage

```
hiveplot_rjs(links, nodes = NULL, group = NULL, size = NULL, color = NULL,
             cex = 1, plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`links`	an edge list table stored in a data frame with three possible columns (source node id, target node id and link value).
`nodes`	node attributes stored in a data frame with rownames as node id.
`group`	column name of `nodes` data frame which describes visualized axes.
`size`	column name of `nodes` data frame which describes each node size.
`color`	column name of `nodes` data frame which describes each node color.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive hive plot in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

if(interactive()){
# Prepare data
x <- 1-cor(t(mtcars))

source <- rep(rownames(x),nrow(x))
target <- rep(rownames(x),rep(ncol(x),nrow(x)))
links <- data.frame(source=source,target=target,value=as.vector(x))

# Generate the hive plot
hiveplot_rjs(links, mtcars, group = "cyl", size = "wt", color = "carb")
}
```

`manhattan_rjs` *Create an interactive manhattan plot*

Description

`manhattan_rjs` creates an interactive Manhattan plot.

Usage

```
manhattan_rjs(data, assembly, axisv, distv, cell = 1e+06,
  outliers = TRUE, ylab = "score", yscale = NULL,
  cex = 1, plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`data`	a data frame with four columns (identifier, chromosome/scaffold, position, score).
`assembly`	a data frame with an assembly to be used. It should be one of the following human assemblies data frames provided by RJSplot: `NCBI36`, `GRCh37`, `GRCh38`.
`axisv`	value for the coordinate of the main horizontal axis.
`distv`	distance between the main horizontal axis and the secondary axes (which stablishes the limit for the outlier representation).
`cell`	segment size which is represented.
`outliers`	logical value to enable (`TRUE`) or disable (`FALSE`) outliers.
`ylab`	a title for the y axis.
`yscale`	a vector with two values which specifies the minimun and maximun limits in the representation. By default maximun and minimun scores are taken as the limits.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive Manhattan plot in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

if(interactive()){
# Create test data
chr <- character()
pos <- numeric()

for(i in 1:nrow(GRCh38)){
  chr <- c(chr,as.character(rep(GRCh38[i,"chr"],100)))
  pos <- c(pos,sample(GRCh38[i,"start"]:GRCh38[i,"end"],100))
}

value <- round(rexp(length(pos)),2)

# Create a manhattan plot
data <- data.frame(paste0("ProbeSet_",seq_along(pos)),chr,pos,value)
manhattan_rjs(data, GRCh38, 0, 1, 0, TRUE, "log2Ratio")
}
```

`NCBI36` *Length of human chromosomes based on NCBI36 assembly*

Description

Length of human chromosomes based on NCBI36 assembly.

Source

See `https://www.ncbi.nlm.nih.gov/assembly/GCF_000001405.12/`.

`network_rjs` *Produce interactive network graph*

Description

`network_rjs` creates an interactive network graph.

Usage

```
network_rjs(links, nodes = NULL,
            weight = NULL, linkLabel = NULL, linkColor = NULL,
            group = NULL, size = NULL, color = NULL, label = NULL,
            nodeColorScale = c("RdBk","RdBkGr"), cex = 1,
            plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`links`	an edge list table stored in a data frame with at least two columns (source node id and target node id).
`nodes`	a node list stored in a data frame object with rownames as node id.
`weight`	a vector giving the column of `links` to show as weight. If `NULL` but third `links` column is numeric, this one will be used as weight.
`linkLabel`	a vector giving the column of `links` to show as link text.
`linkColor`	a vector giving the column of `links` to compute link color or directly a vector of colors.
`group`	a vector giving the column of `nodes` to show as group.
`size`	a vector giving the column of `nodes` to show as size.
`color`	a vector giving the column of `nodes` to compute node color or directly a vector of colors.
`label`	a vector giving the column of `nodes` to show as label.
`nodeColorScale`	a "character" string representing the color scale applied to nodes by default.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive network graph in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

if(interactive()){
# Prepare data
x <- 1-cor(t(mtcars))

source <- rep(rownames(x),nrow(x))
target <- rep(rownames(x),rep(ncol(x),nrow(x)))
links <- data.frame(source=source,target=target,value=as.vector(x))

# Generate the network graph
network_rjs(links[links[,3]>0.1,], mtcars, group = "cyl", size = "hp", color = "mpg")
}
```

`newInput` *Produce a* `newInput` *object for* `R4web`.

Description

`newInput` produce `newInput` object for `R4web`.

Usage

```
newInput(name, type = c("character","numeric","matrix","data.frame","factor"),
         default = "", levels = NULL, multi = FALSE)
```

Arguments

`name`	character string with the variable name.
`type`	character string which describes the variable type. Possible values are "character", "numeric", "matrix", "data.frame" or "factor".
`default`	default value for this input.
`levels`	a vector with the possible input values. It is described only for factor data `type`.
`multi`	boolean value which indicates if multiple selection are enabled for this input. It is described only for factor data `type`.

Value

Return a `newInput` object.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

`piechart_rjs` *Produce interactive pie charts.*

Description

`piechart_rjs` produce interactive pie charts.

Usage

```
piechart_rjs(data, cex = 1, plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`data`	a Data Frame or Matrix from which the data for each pie will be taken.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: https://rjsplot.usal.es

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive pie chart in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es
if(interactive()){
piechart_rjs(VADeaths)
}
```

`R4web` *Generate a PHP website which runs an R function.*

Description

`R4web` generate a PHP website which runs an R function.

Usage

```
R4web(...,fnfile,dir)
```

Arguments

`...`	one or more `newInput` objects which will be included as form fields. Form values will be used as parameters for the R script.
`fnfile`	a character string representing the R script file to be executed with the website.
`dir`	a "character" string representing the directory where the PHP code will be saved.

Value

The function creates a folder in the working directory of your computer with some PHP files. Resulting folder should be added to the Apache applications directory and enable writting permissions.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See https://bioinfo.usal.es/

See Also

The ‘RJSplot’ Website: https://rjsplot.usal.es

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

`report_rjs` *Create a report web page.*

Description

`report_rjs` creates a report web page.

Usage

```
report_rjs(dir = "RJSreport")
addCSS2report(report,css)
addHTML2report(report,HTML)
addParagraph2report(report,p)
addPlot2report(report,plot,height=600)
addImage2report(report,img)
```

Arguments

`dir`	a "character" string representing the directory where the report will be saved.
`report`	a RJSreport object produced by the function `report_rjs`.
`css`	a character string representing the CSS path to be added.
`HTML`	a character string giving the HTML code to be added.
`p`	a character string giving the new paragraph to be added.
`plot`	a character string representing the plot directory to be added.
`height`	an integer vector giving a default height of the plot.
`img`	a character string representing the image path to be added.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the report. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

`RJSplot` *Interactive graphs with R*

Description

Create interactive graphs with 'R'.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

`scatter3d_rjs` *Produce and interactive 3D Scatter plot (XYZ plot).*

Description

`scatter3d_rjs` generates an interactive 3D Scatter Plot.

Usage

```
scatter3d_rjs(x, y, z, color="#000", xlab = "x", ylab = "y", zlab = "z",
              plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`x`	a vector with the x data of each point.
`y`	a vector with the y data of each point.
`z`	a vector with the z data of each point.
`xlab`	a title for the x axis.
`ylab`	a title for the y axis.
`zlab`	a title for the z axis.
`color`	a vector with the colour of each represented point.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive 3D Scatter plot in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

# 3D Scatter plot creation
if(interactive()){
scatter3d_rjs(iris$Sepal.Width, iris$Sepal.Length, iris$Petal.Width,
  color = iris$Species, xlab = "Sepal Width (cm)", ylab = "Sepal Length (cm)",
  zlab = "Petal Width (cm)")
}
```

`scatterplot_rjs` *Produce and interactive Scatter plot (XY plot).*

Description

`scatterplot_rjs` generates an interactive Scatter Plot.

Usage

```
scatterplot_rjs(x, y, id = NULL, xlab = "", ylab = "",
                xlim = NULL, ylim = NULL, col = NULL,
                pch = NULL, abline.x = NULL, abline.y = NULL, cex = 1,
                plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`x`	a vector with the x data of each point.
`y`	a vector with the y data of each point.
`id`	a vector giving the id of each point.
`xlab`	a title for the x axis.
`ylab`	a title for the y axis.
`xlim`	a vector with two values which specifies the X minimun and maximun values in the representation.
`ylim`	a vector with two values which specifies the Y minimun and maximun values in the representation.
`col`	a vector with the colour of each represented point.
`pch`	a character vector with the shape of each point. It should be one of: 1-"circle", 2-"triangle-up", 3-"cross", 4-"square", 5-"diamond", 6-"triangle-down".
`abline.x`	a numeric vector with x coordenates where a vertical line will be placed.
`abline.y`	a numeric vector with y coordenates where a horizontal line will be placed.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive Scatter plot in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

if(interactive()){
# Scatter plot creation
```

```
scatterplot_rjs(iris$Sepal.Width, iris$Sepal.Length,
  abline.x = c(3.4,3.8), abline.y = c(5.8,7),
  col = iris$Species, pch = as.numeric(iris$Species), id = iris$Species,
  xlab = "Sepal Width (cm)", ylab = "Sepal Length (cm)")
}
```

`surface3d_rjs` *Produce and interactive 3D surface.*

Description

`surface3d_rjs` generates an interactive 3D surface.

Usage

```
surface3d_rjs(x, color = "#fff", xlab = "x", ylab = "y", zlab = "z",
              plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`x`	numeric matrix of the values to be plotted.
`color`	a vector with colors to construct the color scale.
`xlab`	a title for the x axis.
`ylab`	a title for the y axis.
`zlab`	a title for the z axis.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive 3D Surface in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

if(interactive()){
# 3D surface creation
surface3d_rjs(volcano,color=c("red","green"))
}
```

`symheatmap_rjs` *Creates a Symetric Heatmap representation.*

Description

`symheatmap_rjs` creates an interactive symetric heatmap.

Usage

```
symheatmap_rjs(links, nodes = NULL, group = NULL, label = NULL, cex = 1,
                plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`links`	an edge list table stored in a data frame with three possible columns (source node id, target node id and a link value).
`nodes`	a node list stored in a data frame object with rownames as node id.
`group`	a vector giving the column of nodes to show as group.
`label`	a vector giving the column of nodes to show as label.
`cex`	number indicating the amount by which plotting text and symbols should be scaled relative to the default.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: https://rjsplot.usal.es

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive symetric heatmap in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

if(interactive()){
# prepare data
x <- 1-cor(t(mtcars))

source <- rep(rownames(x),nrow(x))
target <- rep(rownames(x),rep(ncol(x),nrow(x)))
links <- data.frame(source=source,target=target,value=as.vector(x))

# Create symetric heatmap
symheatmap_rjs(links, mtcars, group = "cyl")
}
```

`tables_rjs` *Produce interactive html tables.*

Description

`tables_rjs` produce interactive html tables of the given values.

Usage

```
tables_rjs(data, plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`data`	a Data Frame or Matrix to display.
`plot`	open resulting table in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the table. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The 'RJSplot' Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive html table in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es
if(interactive()){
tables_rjs(swiss)
}
```

`wordcloud_rjs` *Draw a wordcloud.*

Description

`wordcloud_rjs` creates a wordcloud.

Usage

```
wordcloud_rjs(data, plot = TRUE, jupyter = FALSE, dir = tempdir())
```

Arguments

`data`	a two column data Frame with the words to display and their frequency.
`plot`	open resulting graph in your browser.
`jupyter`	embed the graph as an iframe into a Jupyter Notebook.
`dir`	a "character" string representing the directory where the graph will be saved.

Value

The function creates a folder in your computer with an HTML document named index.html which contains the graph. This file can be directly opened with your browser.

Author(s)

David Barrios and Carlos Prieto. Bioinformatics Service of Nucleus, University of Salamanca. See `https://bioinfo.usal.es/`

See Also

The ‘RJSplot’ Website: `https://rjsplot.usal.es`

`dendrogram_rjs`, `densityplot_rjs`, `genomemap_rjs`, `heatmap_rjs`, `manhattan_rjs`, `network_rjs`, `scatterplot_rjs`, `symheatmap_rjs`, `wordcloud_rjs`, `boxplot_rjs`, `bubbles_rjs`, `hiveplot_rjs`, `piechart_rjs`, `barplot_rjs`, `tables_rjs`, `surface3d_rjs`, `scatter3d_rjs`.

Examples

```
## Create an interactive heatmap in a temporal directory of your local machine
## Live examples on https://rjsplot.usal.es

## Format test data
words <- data.frame(word = rownames(USArrests), freq = USArrests[,4])

# Create WordCloud
if(interactive()){
wordcloud_rjs(words)
}
```

Index