

Package ‘BibPlots’

August 30, 2017

Title Plot Functions for Use in Bibliometrics

Version 0.0.2

Description Currently, the package provides two functions for plotting and analyzing bibliometric data (JIF, Journal Impact Factor, and paper percentile values) and a plot function to visualize the result of a reference publication year spectroscopy (RPYS) analysis performed in the free software CRExplorer (see <<http://crexplorer.net>>). Further extension to more plot variants is planned.

Depends R (>= 3.1.2)

License EUPL

Encoding UTF-8

LazyData true

NeedsCompilation no

Author Robin Haunschild [aut, cre]

Maintainer Robin Haunschild <R.Haunschild@fkf.mpg.de>

Repository CRAN

Date/Publication 2017-08-30 12:36:22 UTC

R topics documented:

BlandAltmanBibPlot	2
example_researcher	3
jpscatter	3
rpys	4
rpys_example_data	5
Index	6

BlandAltmanBibPlot *Create a Bland-Altman plot using journal and paper percentile values*

Description

Provide journal and paper percentile values in a data frame, e.g. `df`, and the function call `BlandAltmanBibPlot(df)` creates the Bland-Altman plot. `BlandAltmanBibPlot` takes some optional arguments to modify its behaviour, see arguments and details.

Usage

```
BlandAltmanBibPlot(df, off_set = 0, print_stats = TRUE, do_plot = TRUE,
  digits = 1, ...)
```

Arguments

<code>df</code>	data frame with journal and paper percentiles
<code>off_set</code>	determines the location of additional plotted information (number of points in each quadrant), values between 0 and 40 might be useful (optional parameter). The default value is 0.
<code>print_stats</code>	boolean variable (optional parameter) which determines if the additional statistical values are printed to the R console (T: yes print, F: no do not print). The default value is T.
<code>do_plot</code>	boolean variable (optional parameter) which determines if the Bland-Altman plot is actually produced (T: yes plot, F: no do not plot). The default value is T.
<code>digits</code>	integer value to determine the number of desired digits after the decimal point for statistical values (optional parameter). The default value is 1.
<code>...</code>	additional arguments to pass to the <code>plot</code> function

Details

`BlandAltmanBibPlot(df=data_frame, off_set=numeric_value, print_stats=boolean, do_plot=boolean)`
 Only the argument `df` is necessary. All other arguments are optional.

Literature:

- Bland, J. M., & Altman, D. G. (1986). Statistical Methods for Assessing Agreement between Two Methods of Clinical Measurement. *Lancet*, 1(8476), 307-310, <https://www.ncbi.nlm.nih.gov/pubmed/2868172>
- Bornmann, L., & Haunschild, R. (2017). Plots for visualizing paper impact and journal impact of single researchers in a single graph, <https://arxiv.org/abs/1707.04050>

An example data frame is provided as `example_researcher` in the package. It can be used to create a Bland-Altman plot using default values.

Examples

```
data(example_researcher)
```

```
BlandAltmanBibPlot(example_researcher)
```

example_researcher	<i>Example data set from publication for scatter plot and Bland Altman plot</i>
--------------------	---

Description

Contains the data set (example_researcher).

jpscat	<i>Create a scatter plot using journal and paper percentile values</i>
--------	--

Description

Provide journal and paper percentile values in a data frame, e.g. df, and the function call jpscat(df) creates the scatter plot. The function jpscat takes some optional arguments to modify its behaviour, see arguments and details.

Usage

```
jpscat(df, off_set = 0, print_stats = TRUE, do_plot = TRUE,
       digits = 1, ...)
```

Arguments

df	data frame with journal and paper percentiles
off_set	determines the location of additional plotted information (number of points in each quadrant), values between 0 and 40 might be useful (optional parameter). The default value is 0.
print_stats	boolean variable (optional parameter) which determines if the additional statistical values are printed to the R console (T: yes print, F: no do not print). The default value is T.
do_plot	boolean variable (optional parameter) which determines if the scatter plot is actually produced (T: yes plot, F: no do not plot). The default value is T.
digits	integer value to determine the number of desired digits after the decimal point for statistical values (optional parameter). The default value is 1.
...	additional arguments to pass to the plot function

Details

jpscat(df=data_frame, off_set=numeric_value, print_stats=boolean, do_plot=boolean, digits=integer)
Only the argument df is necessary. All other arguments are optional.

Literature:

- Bornmann, L., & Haunschild, R. (2017). Plots for visualizing paper impact and journal impact of single researchers in a single graph, <https://arxiv.org/abs/1707.04050>

An example data frame is provided as example_researcher in the package. It can be used to create a scatter plot using default values.

Examples

```
data(example_researcher)

jpsscatter(example_researcher)
```

rpys

Create a spectrogram using data from the free software CRExplorer

Description

Provide the contents of the CSV file from the CRExplorer in a data frame, e.g. `df`, and the function call `rpys(df, py1, py2)` creates the spectrogram. Here, `py1` and `py2` are the lowest and highest publication year to be used in the plot. The function `rpys` takes some optional arguments to modify its behaviour, see arguments and details.

Usage

```
rpys(df, py1, py2, col_cr = "red", col_med = "blue", smoothing = TRUE,
      par_pch = 20, ...)
```

Arguments

<code>df</code>	data frame with reference publication year, number of cited references, and median deviation as exported from the CRExplorer (File > Export > CSV (Graph)).
<code>py1</code>	determines lowest reference publication year which should be shown in the graph.
<code>py2</code>	determines highest reference publication year which should be shown in the graph.
<code>col_cr</code>	character color name value to determine color of the line and points of the number of cited references (optional parameter). The default value is "red".
<code>col_med</code>	character color name value to determine color of the line and points of the median deviation (optional parameter). The default value is "blue".
<code>smoothing</code>	boolean variable (optional parameter) which determines if the lines of the spectrogram are smoothed or not. (T: yes apply smoothing, F: no do not apply smoothing). The default value is T.
<code>par_pch</code>	integer value to set the point type (optional parameter). The default value is 20.
<code>...</code>	additional arguments to pass to the plot , points , and lines functions.

Details

`rpys(df=data_frame, py1=integer_value, py2=integer_value, smoothing=boolean, col_cr=character_color_name, col_med=character_color_name, par_pch=integer, ...)` Only the arguments `df`, `py1`, and `py2` are necessary. All other arguments are optional.

Literature:

- Thor, A., Bornmann, L., Marx, W., Haunschild, R., Leydesdorff, L., & Mutz, Ruediger (2017). Website of the free software CRExplorer, <http://www.crexplorer.net>

An example data frame is provided as `rpys_example_data` in the package. It can be used to create an example spectrogram.

Examples

```
data(rpys_example_data)
```

```
rpys(rpys_example_data, 1935, 2010)
```

<code>rpys_example_data</code>	<i>Example data set for the rpys function</i>
--------------------------------	---

Description

Contains the data sets (`rpys_example_data`).

Index

*Topic **data**

example_researcher, 3

rpys_example_data, 5

BlandAltmanBibPlot, 2

example_researcher, 3

jpscatter, 3

lines, 4

plot, 2-4

points, 4

rpys, 4

rpys_example_data, 5