

Package ‘BibPlots’

May 23, 2019

Title Plot Functions for Use in Bibliometrics

Version 0.0.5

Description Currently, the package provides four functions for plotting and analyzing bibliometric data (JIF, Journal Impact Factor, and paper percentile values), beamplots with citations, and two 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

RoxygenNote 6.1.1

NeedsCompilation no

Author Robin Haunschild [aut, cre]

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

Repository CRAN

Date/Publication 2019-05-23 09:30:07 UTC

R topics documented:

beamplot	2
DAMBibPlot	3
example_researcher	4
jpsscatter	4
ncr_comp	5
rpys	6
rpys_example_data	7
Index	8

beamplot

Create a beamplot using raw citations from a WoS download

Description

Create a beamplot using raw citations from a WoS download. Use the format "Other File Format -> Tab-delimited (Win, UTF-8)" and provide the downloaded file name. A simple weighting of citation counts is also available for comparison of older with newer publications.

Usage

```
beamplot(wos_file, do_weight = FALSE, ...)
```

Arguments

wos_file	is the file name of the downloaded WoS export in the format Tab-delimited (Win, UTF-8).
do_weight	is a boolean to specify if citation counts should be weighted with their age. The older the publication, the smaller the weight. The weight depends on the difference between the year until that citations are counted (i.e., the current calendar year in the case of WoS downloads) and the publication year. A weighting factor of 1 is used for a difference of 0, 1/2 for a difference of 1, ..., and 1/11 for differences of ten or more.
...	further parameters passed to stripchart.

Details

beamplot(wos_file="WoS_savedrecs.txt", do_weight=boolean) Only the argument wos_file is mandatory. The argument do_weight is optional and FALSE by default.

Literature:

- Haunschild, R., Bornmann, L., & Adams, J. (2019). R package for producing beamplots as a preferred alternative to the h index when assessing single researchers (based on downloads from Web of Science), preprint: <https://arxiv.org/abs/1905.09095>

Examples

```
## Not run: beamplot("WoS_savedrecs.txt")
```

DAMBibPlot	<i>Create a difference against mean 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 `DAMBibPlot(df)` creates the difference against mean plot. `DAMBibPlot` takes some optional arguments to modify its behaviour, see arguments and details.

Usage

```
DAMBibPlot(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 difference against mean 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

`DAMBibPlot(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>

Cleveland, W. S. (1985). *The elements of graphing data*. Monterey, CA: Wadsworth Advanced Books and Software.

- Bornmann, L., & Haunschild, R. (2017). Plots for visualizing paper impact and journal impact of single researchers in a single graph, DOI: 10.1007/s11192-018-2658-1, preprint: <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 difference against mean plot using default values.

Examples

```
data(example_researcher)

DAMBibPlot(example_researcher)
```

example_researcher	<i>Example data set from publication for scatter plot and difference against mean plot</i>
--------------------	--

Description

Contains the data set (example_researcher).

jpscatter	<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 jpscatter(df) creates the scatter plot. The function jpscatter takes some optional arguments to modify its behaviour, see arguments and details.

Usage

```
jpscatter(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

`jpscatter(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, DOI: 10.1007/s11192-018-2658-1, preprint: <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)
```

```
jpscatter(example_researcher)
```

ncr_comp

Create a spectrogram using data from the free software CRExplorer

Description

Provide the contents of CSV files from the 'CRExplorer' in data frames, e.g. `df1` and `df2`, and the function call `ncr_comp(df1, df2, py1, py2)` creates a plot with both sets of NCR values. Here, `py1` and `py2` are the lowest and highest publication year to be used in the plot. The function `ncr_comp` takes some optional arguments to modify its behaviour, see arguments and details.

Usage

```
ncr_comp(df1, df2, py1, py2, col_cr = "red", smoothing = TRUE,
         par_pch = 20, ...)
```

Arguments

<code>df1</code>	data frame 1 with reference publication year and number of cited references, e.g., as exported from the CRExplorer (File > Export > CSV (Graph)).
<code>df2</code>	data frame 2 with reference publication year and number of cited references, e.g., 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".

smoothing	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.
par_pch	integer value to set the point type (optional parameter). The default value is 20.
...	additional arguments to pass to the plot , points , and lines functions.

Details

```
ncr_comp <- function(df1, df2, py1, py2, col_cr = "red", smoothing = TRUE, par_pch = 20, ...)
```

Only the arguments `df1`, `df2`, `py1`, and `py2` are necessary. All other arguments are optional. Please use the function `legend` to add a user-defined legend. The solid curve represents the data from `df1` and the dotted curve represents the data from `df2`.

Literature:

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

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, plot_NCR = TRUE, plot_Med = TRUE,
      ...)
```

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".

smoothing	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.
par_pch	integer value to set the point type (optional parameter). The default value is 20.
plot_NCR	boolean variable (optional parameter) which determines the NCR curve should be plotted.
plot_Med	boolean variable (optional parameter) which determines the median deviation curve should be plotted.
...	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, plot_NCR=boolean, plot_Med=boolean, ...) 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)
```

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

Description

Contains the data sets (`rpys_example_data`).

Index

*Topic **data**

example_researcher, 4

rpys_example_data, 7

beamplot, 2

DAMBibPlot, 3

example_researcher, 4

jpscat, 4

lines, 6, 7

ncr_comp, 5

plot, 3, 4, 6, 7

points, 6, 7

rpys, 6

rpys_example_data, 7