Package 'statcheck'

August 18, 2016

Type Package

Title Extract Statistics from Articles and Recompute p Values

Version 1.2.2

Date 2016-08-18

Author Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>

Maintainer Michele B. Nuijten <m.b.nuijten@uvt.nl>

Depends R (>= 2.14.2)

Imports plyr, ggplot2

Description Extract statistics from articles and recompute p values.

License GPL-2

LazyLoad yes

ByteCompile yes

NeedsCompilation no

Repository CRAN

Date/Publication 2016-08-18 10:02:24

R topics documented:

statcheck-package *Extract statistics from articles and recompute p values*

Description

Extract statistics from articles and recompute p values.

Details

Package:	statcheck
Type:	Package
Title:	Extract statistics from articles and recompute p values
Version:	1.0.0
Date:	2014-11-15
Author:	Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>
Maintainer:	Michele B. Nuijten <m.b.nuijten@uvt.nl>
Depends:	R (>= 2.14.2), plyr
License:	GPL-2
LazyLoad:	yes
ByteCompile:	yes

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten

<m.b.nuijten@uvt.nl>

checkdir *Extract test statistics from all HTML and PDF files in a folder.*

Description

Extracts statistical references from a directory with HTML and PDF files. The "pdftotext" program is used to convert PDF files to plain text files. This must be installed and PATH variables must be properly set so that this program can be used from command line.

By default a gui window is opened that allows you to choose the directory (using tcltk).

Usage

```
checkdir(dir, subdir = TRUE, ...)
```

Arguments

dir	String indicating the directory to be used. If this is left empty, a window will pop up from which you can choose a directory.
subdir	Logical indicating whether you also want to check subfolders. Defaults to TRUE
...	Arguments sent to statcheck.

Details

See statcheck for more details. This function is a wrapper around both checkPDFdir for PDF files and checkHTMLdir for HTML files.

Depending on the PDF file the comparison operators (=/</>) can sometimes not be converted correctly, causing these to not be reported in the output. Using html versions of articles is reccomended for more stable results.

Note that the conversion to plain text and extraction of statistics can result in errors. Some statistical values can be missed, especially if the notation is unconventional. It is recommended to manually check some of the results.

Value

A data frame containing for each extracted statistic:

Source	Name of the file of which the statistic is extracted
Statistic	Character indicating the statistic that is extracted
df1	First degree of freedom
df2	Second degree of freedom (if applicable)
Test.Comparison	Reported comparison of the test statistic, when importing from pdf this will often not be converted properly
Value	Reported value of the statistic
Reported.Comparison	Reported comparison, when importing from pdf this might not be converted properly
Reported.P.Value	The reported p-value, or NA if the reported value was NS
Computed	The recomputed p-value
Raw	Raw string of the statistical reference that is extracted
Error	The computed p value is not congruent with the reported p value
DecisionError	The reported result is significant whereas the recomputed result is not, or vice versa.
OneTail	Logical. Is it likely that the reported p value resulted from a correction for one-sided testing?
OneTailedInTxt	Logical. Does the text contain the string "sided", "tailed", and/or "directional"?
CopyPaste	Logical. Does the exact string of the extracted raw results occur anywhere else in the article?

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>

See Also

statcheck, checkPDF, checkHTMLdir, checkHTML, checkHTMLdir

Examples

```
# with this command a menu will pop up from which you can select the directory with articles

# checkdir()



# you could also specify the directory beforehand

# for instance:

# DIR <- "C:/mydocuments/articles"

# checkdir(DIR)
```

---

checkHTML *Extract test statistics from HTML file.*

Description

Extracts statistical references from given HTML files.

Usage

```
checkHTML(files, ...)
```

Arguments

files	Vector of strings containing file paths to HTML files to check.
...	Arguments sent to statcheck.

Details

See statcheck for more details. Use checkHTMLdir to import al HTML files in a given directory at once.

Note that the conversion to plain text and extraction of statistics can result in errors. Some statistical values can be missed, especially if the notation is unconvetional. It is recommended to manually check some of the results.

Value

A data frame containing for each extracted statistic:

Source	Name of the file of which the statistic is extracted
Statistic	Character indicating the statistic that is extracted
df1	First degree of freedom
df2	Second degree of freedom (if applicable)
Value	Reported value of the statistic
Reported.Comparison	Reported comparison, when importing from pdf this will often not be converted properly
Reported.P.Value	The reported p-value, or NA if the reported value was NS
Computed	The recomputed p-value
Raw	Raw string of the statistical reference that is extracted
InExactError	Error in inexactly reported p values as compared to the recalculated p values
ExactError	Error in exactly reported p values as compared to the recalculated p values
DecisionError	The reported result is significant whereas the recomputed result is not, or vice versa.

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>

See Also

statcheck, checkPDF, checkPDFdir, checkHTMLdir, checkdir

Examples

```
  # given that my HTML file is called "article.html"


# and I saved it in "C:/mydocuments/articles"
```

```
#checkHTML("C:/mydocuments/articles/article.html")
```

---

checkHTMLdir *Extract test statistics from all HTML files in a folder.*

Description

Extracts statistical references from a directory with HTML versions of articles. By default a GUI window is opened that allows you to choose the directory (using tcltk).

Usage

```
checkHTMLdir(dir, subdir = TRUE, extension=TRUE, ...)
```

Arguments

dir	String indicating the directory to be used.
subdir	Logical indicating whether you also want to check subfolders. Defaults to TRUE
extension	Logical, indicating whether the HTML extension should be checked. Defaults to TRUE
...	Arguments sent to statcheck

Details

See statcheck for more details. Use checkHTML to import individual HTML files.

Note that the conversion to plain text and extraction of statistics can result in errors. Some statistical values can be missed, especially if the notation is unconventional. It is recommended to manually check some of the results.

Value

A data frame containing for each extracted statistic:

Source	Name of the file of which the statistic is extracted
Statistic	Character indicating the statistic that is extracted
df1	First degree of freedom
df2	Second degree of freedom (if applicable)
Test.Comparison	Reported comparison of the test statistic, when importing from pdf this will often not be converted properly

`Value`	Reported value of the statistic
`Reported.Comparison`	Reported comparison, when importing from pdf this might not be converted properly
`Reported.P.Value`	The reported p-value, or NA if the reported value was NS
`Computed`	The recomputed p-value
`Raw`	Raw string of the statistical reference that is extracted
`Error`	The computed p value is not congruent with the reported p value
`DecisionError`	The reported result is significant whereas the recomputed result is not, or vice versa.
`OneTail`	Logical. Is it likely that the reported p value resulted from a correction for one-sided testing?
`OneTailedInTxt`	Logical. Does the text contain the string "sided", "tailed", and/or "directional"?
`CopyPaste`	Logical. Does the exact string of the extracted raw results occur anywhere else in the article?

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>

See Also

`statcheck`, `checkPDF`, `checkPDFdir`, `checkHTML`, `checkdir`

Examples

```
# with this command a menu will pop up from which you can select the directory with HTML articles

 # checkHTMLdir()



# you could also specify the directory beforehand

# for instance:

# DIR <- "C:/mydocuments/articles"

# checkHTMLdir(DIR)
```

checkPDF *Extract statistics and recompute p-values from pdf files.*

Description

Extracts statistical values (currently only t and F statistics) from PDF files. To this end the "pdftotext" program is used to convert PDF files to plain text files. This must be installed and PATH variables must be properly set so that this program can be used from command line.

Usage

```
checkPDF(files, ...)
```

Arguments

files	Vector with paths to the PDF files.
...	Arguments sent to statcheck

Details

See statcheck for more details. Use checkPDFdir to import every PDF file in a given directory. Currently only statistics in the form "(stat (df1, df2) = value, p = value)" are extracted.

Note that this function is still in devellopment. Some statistical values can be missed, especially if the notation is unconvetional. It is recommended to manually check some of the results.

Value

A data frame containing for each extracted statistic:

Source	Name of the file of which the statistic is extracted
Statistic	Character indicating the statistic that is extracted
df1	First degree of freedom
df2	Second degree of freedom (if applicable)
Value	Reported value of the statistic
Reported.Comparison	Reported comparison, when importing from pdf this will often not be converted properly
Reported.P.Value	The reported p-value, or NA if the reported value was NS
Computed	The recomputed p-value
Raw	Raw string of the statistical reference that is extracted
InExactError	Error in inexactly reported p values as compared to the recalculated p values
ExactError	Error in exactly reported p values as compared to the recalculated p values
DecisionError	The reported result is significant whereas the recomputed result is not, or vice versa.

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>

See Also

statcheck, checkPDFdir

Examples

```
# given that my PDF file is called "article.pdf"

# and I saved it in "C:/mydocuments/articles"



 # checkPDF("C:/mydocuments/articles/article.pdf")
```

---

checkPDFdir *Extract statistics and recompute p values from a directory with pdf files.*

Description

Extracts statistical references from a directory with PDF files. The "pdftotext" program (http://www.foolabs.com/xpdf/downlo is used to convert PDF files to plain text files. This must be installed and PATH variables must be properly set so that this program can be used from command line.

By default a GUI window is opened that allows you to choose the directory (using tcltk).

Usage

```
checkPDFdir(dir, subdir = TRUE, ...)
```

Arguments

dir	String indicating the directory to be used.
subdir	Logical indicating whether you also want to check subfolders. Defaults to TRUE
...	Arguments sent to statcheck.

Details

See statcheck for more details. Use checkPDF to import individual PDF files. Currently only statistics in the form "stat (df1, df2) = value, p = value" are extracted. Because the Chi-square symbol can not be repressented in plain text it is often lost in the conversion. Because of this Chi-square values are extracted by finding all statistical references with one degree of freedom that do not follow the symbol "t" or "r". While this does extract most Chi-square values it is possible that other statistics, possibly due to unconventional notation, are also extracted and reported as chi-square values.

Depending on the PDF file the comparison operators can sometimes not be converted correctly, causing these to not be reported in the output. Using html versions of articles and the similar function checkHTMLdir is recommended for more stable results.

Note that the conversion to plain text and extraction of statistics can result in errors. Some statistical values can be missed, especially if the notation is unconventional. It is recommended to manually check some of the results.

Value

A data frame containing for each extracted statistic:

Source	Name of the file of which the statistic is extracted
Statistic	Character indicating the statistic that is extracted
df1	First degree of freedom
df2	Second degree of freedom (if applicable)
Test.Comparison	Reported comparison of the test statistic, when importing from pdf this will often not be converted properly
Value	Reported value of the statistic
Reported.Comparison	Reported comparison, when importing from pdf this might not be converted properly
Reported.P.Value	The reported p-value, or NA if the reported value was NS
Computed	The recomputed p-value
Raw	Raw string of the statistical reference that is extracted
Error	The computed p value is not congruent with the reported p value
DecisionError	The reported result is significant whereas the recomputed result is not, or vice versa.
OneTail	Logical. Is it likely that the reported p value resulted from a correction for one-sided testing?
OneTailedInTxt	Logical. Does the text contain the string "sided", "tailed", and/or "directional"?
CopyPaste	Logical. Does the exact string of the extracted raw results occur anywhere else in the article?

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>

See Also

statcheck, checkPDF, checkHTMLdir, checkHTML, checkdir

Examples

```
 # with this command a menu will pop up from which you can select the directory with PDF articles

# checkPDFdir()



# you could also specify the directory beforehand

# for instance:

# DIR <- "C:/mydocuments/articles"

# checkPDFdir(DIR)
```

---

identify.statcheck *Identify specific points in a* statcheck *plot.*

Description

With this function you can simply point and click on the datapoints in the plot to see the corresponding statcheck details, such as the paper from which the data came and the exact statistical results.

Usage

```
## S3 method for class 'statcheck'
identify(x, alpha = 0.05, ...)
```

Arguments

x	a statcheck object.
alpha	assumed level of significance in the scanned texts. Defaults to .05.
...	additional arguments to be passed on to the plot method.

Value

This function returns both a plot and a dataframe. For the contents of the dataframe see statcheck.

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>

See Also

statcheck

Examples

```
# given that the articles of interest are saved in "DIR"

# DIR <- "C:/mydocuments/articles"

# stat_result <- checkdir(DIR)



# identify(stat_result)



## Further instructions:

# click on one or multiple points of interest

# press Esc

# a dataframe with information on the selected points will appear
```

`plot.statcheck` *Plot method for "statcheck"*

Description

Function for plotting of "statcheck" objects. Reported p values are plotted against recalculated p values, which allows the user to easily spot if articles contain miscalculations of statistical results.

Usage

```
## S3 method for class 'statcheck'
plot(x, alpha = 0.05, APAstyle = TRUE, group = NULL,

    ...)
```

Arguments

`x`	a "statcheck" object. See `statcheck`.
`alpha`	assumed level of significance in the scanned texts. Defaults to .05.
`APAstyle`	if TRUE, prints plot in APA style
`group`	indicate grouping variable to facet plot. Only works when APAstyle==TRUE
`...`	arguments to be passed to methods, such as graphical parameters (see `par`).

Details

If APAstyle = FALSE, inconsistencies between the reported and the recalculated p value are indicated with an orange dot. Recalculations of the p value that render a previously non significant result (p >= .5) as significant (p < .05), and vice versa, are considered gross errors, and are indicated with a red dot. Exactly reported p values (i.e. p = ..., as opposed to p < ... or p > ...) are indicated with a diamond.

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten

<m.b.nuijten@uvt.nl>. Many thanks to John Sakaluk who adapted the plot code to create graphs in APA style.

See Also

`statcheck`

`statcheck` *Extract statistics and recompute p-values.*

Description

This function extracts statistics from strings and returns the extracted values, reported p-values and recomputed p-values. The package relies on the program "pdftotext", see the paragraph "Note" for details on the installation.

Usage

```
statcheck(x, stat = c("t", "F", "cor", "chisq", "Z"),

   OneTailedTests = FALSE, alpha = 0.05, pEqualAlphaSig = TRUE, pZeroError = TRUE,

    OneTailedTxt = FALSE, AllPValues = FALSE)
```

Arguments

`x`	A vector of strings.
`stat`	"t" to extract t-values, "F" to extract F-values, "cor" to extract correlations, "chisq"to extract chi-square values, and "Z" to extract Z-values.
`OneTailedTests`	Logical. Do we assume that all reported tests are one tailed (TRUE) or two tailed (FALSE, default)?
`alpha`	Assumed level of significance in the scanned texts. Defaults to .05.
`pEqualAlphaSig`	Logical. If TRUE, statcheck counts p <= alpha as significant (default), if FALSE, statcheck counts p < alpha as significant
`pZeroError`	Logical. If TRUE, statcheck counts p=.000 as an error (because a p-value is never exactly zero, and should be reported as < .001), if FALSE, statcheck does not count p=.000 automatically as an error.
`OneTailedTxt`	Logical. If TRUE, statcheck searches the text for "one-sided", "one-tailed", and "directional" to identify the possible use of one-sided tests. If one or more of these strings is found in the text AND the result would have been correct if it was a one-sided test, the result is assumed to be indeed one-sided and is counted as correct.
`AllPValues`	Logical. If TRUE, the output will consist of a dataframe with all detected p values, also the ones that were not part of the full results in APA format

Details

statcheck uses regular expressions to find statistical results in APA format. When a statistical result deviates from APA format, statcheck will not find it. The APA formats that statcheck uses are: t(df) = value, p = value; F(df1,df2) = value, p = value; r(df) = value, p = value; [chi]2 (df, N = value) = value, p = value (N is optional, delta G is also included); Z = value, p = value. All regular

expressions take into account that test statistics and p values may be exactly (=) or inexactly (< or >) reported. Different spacing has also been taken into account.

This function can be used if the text of articles has already been imported in R. To import text from pdf files and automatically send the results to this function use `checkPDFdir` or `checkPDF`. To import text from HTML files use the similar functions `checkHTMLdir` or `checkHTML`. Finally, `checkdir` can be used to import text from both PDF and HTML files in a folder.

Note that the conversion from PDF (and sometimes also HTML) to plain text and extraction of statistics can result in errors. Some statistical values can be missed, especially if the notation is unconventional. It is recommended to manually check some of the results.

PDF files should automatically be converted to plain text files. However, if this does not work, it might help to manually install the program "pdftotext". You can obtain pdftotext from `http://www.foolabs.com/xpdf/down` Download and unzip the precompiled binaries. Next, add the folder with the binaries to the PATH variables so that this program can be used from command line.

Also, note that a seemingly inconsistent p value can still be correct when we take into account that the test statistic might have been rounded after calculating the corresponding p value. For instance, a reported t value of 2.35 could correspond to an actual value of 2.345 to 2.354 with a range of p values that can slightly deviate from the recomputed p value. Statcheck will not count cases like this as errors.

Value

A data frame containing for each extracted statistic:

`Source`	Name of the file of which the statistic is extracted
`Statistic`	Character indicating the statistic that is extracted
`df1`	First degree of freedom (if applicable)
`df2`	Second degree of freedom
`Test.Comparison`	Reported comparison of the test statistic, when importing from pdf this will often not be converted properly
`Value`	Reported value of the statistic
`Reported.Comparison`	Reported comparison, when importing from pdf this might not be converted properly
`Reported.P.Value`	The reported p-value, or NA if the reported value was NS
`Computed`	The recomputed p-value
`Raw`	Raw string of the statistical reference that is extracted
`Error`	The computed p value is not congruent with the reported p value
`DecisionError`	The reported result is significant whereas the recomputed result is not, or vice versa.
`OneTail`	Logical. Is it likely that the reported p value resulted from a correction for one-sided testing?
`OneTailedInTxt`	Logical. Does the text contain the string "sided", "tailed", and/or "directional"?
`CopyPaste`	Logical. Does the exact string of the extracted raw results occur anywhere else in the article?

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>

See Also

checkPDF, checkHTMLdir, checkHTML, checkdir

Examples

```
txt <- "blablabla the effect was very significant (t(100)=1, p < 0.001)"

statcheck(txt)
```

---

summary.statcheck *Summary method for* statcheck.

Description

Gives the summaries for a statcheck object.

Usage

```
## S3 method for class 'statcheck'
summary(object, ...)
```

Arguments

object	a statcheck object.
...	additional arguments affecting the summary produced.

Value

A data frame containing for each extracted statistic:

Source	Name of the file of which the statistic is extracted
pValues	The number of reported p values per article
Errors	The number of errors per article
DecisionErrors	The number of errors that caused a non-significant result to be reported as significant (or vice versa) per article

Author(s)

Sacha Epskamp <mail@sachaepskamp.com> & Michele B. Nuijten <m.b.nuijten@uvt.nl>

See Also

statcheck

Examples

```
  Text <- "blablabla the effect was very significant (t(100)=1, p < 0.001)"

Stat <- statcheck(Text)



summary(Stat)
```

Index