Package ‘statquotes’

March 5, 2022

Title Quotes on Statistics, Data Visualization and Science

Version 0.2.6

Date 2022-03-03

Language en-US

Description Generates a random quotation from a data base of quotes on topics in statistics, data visualization and science. Other functions allow searching the quotes database or creating a word cloud.

Depends R (>= 3.5.0)

License GPL (>= 2)

Encoding UTF-8

LazyData true

Maintainer Michael Friendly <friendly@yorku.ca>

BugReports https://github.com/friendly/statquotes/issues

URL https://github.com/friendly/statquotes/

Imports stringr, tidytext, wordcloud

RoxygenNote 7.1.2

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Michael Friendly [aut, cre],
Phil Chalmers [ctb],
Matthew Sigal [ctb]

Repository CRAN

Date/Publication 2022-03-05 17:50:02 UTC

R topics documented:

as.latex *Function coerces statquote objects to strings suitable for LaTeX*

Description

This function coerces statquote objects to strings suitable for rendering in LaTeX. Quotes and (potential LaTeX) sources are placed within suitable "epigraph" output format via the sprintf function.

Usage

```
as.latex(quotes, form = "\\epigraph{%s}{%s}\n\n")
```

Arguments

quotes — an object of class statquote returned from functions such as search_quotes or statquote

form — structure of the LaTeX output for the text (first argument) and source (second argument) passed to sprintf

Value

character vector of formatted LaTeX quotes

Examples

```
ll <- search_quotes("Tukey")
as.latex(ll)
```

as.markdown *Function to transform statquote objects to strings suitable for markdown*

Description

This function coerces statquote objects to strings suitable for rendering in markdown. Quotes and sources are placed within output format via the sprintf function.

Usage

```
as.markdown(quotes, form = "> *%s* -- %s\n\n")
```

Arguments

quotes an object of class statquote returned from functions such as search_quotes or statquote

form structure of the markdown output for the text (first argument) and source (second argument) passed to sprintf

Value

character vector of formatted markdown quotes

Examples

```
ll <- search_quotes("Tukey")
as.markdown(ll)
```

find_duplicate_quotes *Check for duplicates in master .csv file*

Description

Returns a list with qid, source, and the text where strings are aggressively fuzzy matched.

Usage

```
find_duplicate_quotes()
```

Author(s)

Phil Chalmers

Examples

```
#  find_duplicate_quotes()
```

---

quotes *Quotes on statistics, data visualization and science*

Description

A data frame with over 200 quotations. The variables are:

Usage

```
data(quotes)
```

Format

A data frame with 231 rows and 5 variables

Details

- `qid` quote ID, a numeric vector
- `topic` main topic, a factor with levels `Computing` `Data` `Data visualization` `History` `Reviews` `Science` `Statistics` `Unclassified`
- `subtopic` sub topic, a factor with levels `Averages` `Box quotes` `Counts` `Design` `Ellipses` `Generalizations` `Milestones` `Pictures` `Tables` `Tidy data` `Time` `Tukey quotes`
- `text` text of the quote, a character vector
- `source` source of the quote, a character vector

---

quote_cloud *Function to generate word cloud based upon quote database*

Description

This function takes a search pattern (can use regular expressions) and generates a word cloud based upon that filter.

Usage

```
quote_cloud(search = ".*", max.words = 80, colors, ...)
```

Arguments

`search` A character string; used to search the database. Regular expression characters are allowed. Default is to search all quotes.

`max.words` Logical; designate maximum number of words to be plotted.

`colors` A character vector pertaining to the colors to be used to designate word frequency. The default is 5 levels, from light to dark green.

`...` additional arguments passed to `search_quotes` and `wordcloud`

Value

A wordcloud is plotted.

See Also

`statquote`, `quote_topics`, `quotes`, `search_quotes`. `wordcloud`

Examples

```
quote_cloud()
quote_cloud(search = "graph")
quote_cloud(max.words = 10)
```

`quote_topics` *List the topics of the quotes data base*

Description

List the topics of the quotes data base

Usage

```
quote_topics(subtopics = FALSE)
```

Arguments

`subtopics` logical; if `TRUE` the subtopics are printed as well with the associated topic

Examples

```
quote_topics()
quote_topics(TRUE)
```

search_quotes *Search the quote database for a string or regex pattern*

Description

This function takes a search pattern (can use regular expressions) and returns all quotes that match the pattern. By default all fields are included in the search. If fuzzy is FALSE, then only exact matches are returned (case sensitive).

A convenient wrapper for search quotes that by default returns all quotes

Usage

```
search_quotes(search, ignore_case = TRUE, fuzzy = FALSE, fields = NULL, ...)

search_text(search, fuzzy = FALSE, ...)

get_quotes(search = ".*", ...)
```

Arguments

`search`	A character string, used to search the database. Regular expression characters are allowed.
`ignore_case`	Logical; If `TRUE`, matching is done without regard to case.
`fuzzy`	Logical; If `TRUE`, the function uses `agrep` to allow approximate matches to the search string.
`fields`	A character vector pertaining to the particular fields to search. The default is to search everything: ‘c("topic", "subtopic", "text", "source", "TeXsource")‘.
`...`	additional arguments passed to `agrep` to fine-tune fuzzy search parameters.

Value

A data frame (also with class `'statquote'`) object containing all quotes that match the search parameters.

A data frame (also with class `'statquote'`) object containing all quotes. This is meant to be assigned to name rather than printed.

See Also

`agrep`, `statquote`, `quote_topics`, `quotes`

Examples

```
search_quotes("^D") # regex to find all quotes that start with "D"
search_quotes("Tukey") #all quotes with "Tukey"
search_quotes("bad answer", fuzzy = TRUE) # fuzzy match

# to a data.frame
out <- search_quotes("bad answer", fuzzy = TRUE)
as.data.frame(out)

qdb <- get_quotes()
nrow(qdb)
names(qdb)
```

statquote *Function to display a randomly chosen statistical quote*

Description

This function displays a randomly statistical quote from a collection. The quotations are classified by topics

Usage

```
statquote(ind, topic = NULL, source = NULL)

## S3 method for class 'statquote'
print(x, width = NULL, ...)

## S3 method for class 'statquote'
as.data.frame(x, row.names = NULL, optional = FALSE, ...)
```

Arguments

`ind` Optional index of a quote, or a vector of such integer indices If missing a random value is sampled from the available quotations.

`topic` A character string, used to select a subset of the quotes based on the assigned topics.

`source` A character string, used to select a subset of the quotes based on the source for the quote.

`x` object of class `'statquote'`

`width` Optional column width parameter

`...` Other optional arguments

`row.names` see `as.data.frame`

`optional` see `as.data.frame`

Value

A character vector containing one randomly selected quote from the included data set. It is of class `statquote` for which an S3 print method will be invoked.

See Also

`quote_topics`, `search_quotes`, `quotes`, Inspired by: `gaussfact` (`https://github.com/eddelbuettel/gaussfacts`), `fortune`

Examples

```
set.seed(1234)
statquote(123)
statquote(source="Tukey")
statquote(topic="science")
statquote(topic="history")
```

Index