

# Package ‘statquotes’

October 8, 2018

**Title** Quotes on Statistics, Data Visualization and Science

**Version** 0.2.2

**Description** Generates a random quotation from a data base of quotes on topics in statistics, data visualization and science.

**Depends** R (>= 3.2.5)

**License** GPL (>= 2)

**Encoding** UTF-8

**LazyData** true

**Maintainer** Michael Friendly <friendly@yorku.ca>

**BugReports** <https://github.com/friendly/statquotes/issues>

**Imports** stringr, tidytext, wordcloud

**RoxygenNote** 6.0.1

**NeedsCompilation** no

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

**Repository** CRAN

**Date/Publication** 2017-08-29 21:24:38 UTC

## R topics documented:

quotes . . . . .	2
quote_cloud . . . . .	2
quote_topics . . . . .	3
search_quotes . . . . .	3
statquote . . . . .	4
<b>Index</b>	<b>6</b>

---

 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 = NA, ...)
```

### 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 <a href="#">search_quotes</a> and <a href="#">wordcloud</a>

**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	<i>List the topics of the quotes data base</i>
--------------	--

---

**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	<i>Function to search quote database</i>
---------------	--

---

**Description**

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

**Usage**

```
search_quotes(search, fuzzy = FALSE, fields = c("topic", "subtopic", "text",
"source"), ...)
```

**Arguments**

search	A character string, used to search the database. Regular expression characters are allowed.
fuzzy	Logical; If TRUE, the function uses <a href="#">agrep</a> 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: <code>c("topic", "subtopic", "text", "source")</code> .
...	additional arguments passed to <a href="#">agrep</a> to fine-tune fuzzy search parameters.

**Value**

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

**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)
```

---

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; 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 <a href="#">as.data.frame</a>
optional	see <a href="#">as.data.frame</a>

**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/gaussfact) (<https://github.com/eddelbuettel/gaussfact>)  
[fortune](#)

**Examples**

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

# Index

## \*Topic **datasets**

quotes, [2](#)

agrep, [4](#)

as.data.frame, [5](#)

as.data.frame.statquote (statquote), [4](#)

fortune, [5](#)

print.statquote (statquote), [4](#)

quote\_cloud, [2](#)

quote\_topics, [3](#), [3](#), [4](#), [5](#)

quotes, [2](#), [3–5](#)

search\_quotes, [2](#), [3](#), [3](#), [5](#)

statquote, [3](#), [4](#), [4](#)

wordcloud, [2](#), [3](#)