

# Package ‘wordcloud’

February 15, 2012

**Type** Package

**Title** Word Clouds

**Version** 2.0

**Date** 2012-01-09

**Author** Ian Fellows

**Maintainer** Ian Fellows <ian@fellstat.com>

**Description** Pretty word clouds

**License** LGPL-2.1

**LazyLoad** yes

**Depends** methods, Rcpp (>= 0.9.4), RColorBrewer

**Suggests** tm

**URL** <http://www.fellstat.com> <http://research.cens.ucla.edu/>

**LinkingTo** Rcpp

**Repository** CRAN

**Date/Publication** 2012-01-10 14:06:01

## R topics documented:

commonality.cloud . . . . .	2
comparison.cloud . . . . .	3
SOTU . . . . .	4
wordcloud . . . . .	4

<b>Index</b>	<b>6</b>
--------------	----------

commonality.cloud      *Plot a commonality cloud*

---

## Description

Plot a cloud of words shared across documents

## Usage

```
commonality.cloud(term.matrix, comonality.measure=min, max.words=300, ...)
```

## Arguments

`term.matrix`      A term frequency matrix whose rows represent words and whose columns represent documents.

`comonality.measure`  
A function taking a vector of frequencies for a single term, and returning a common frequency

`max.words`      Maximum number of words to be plotted. least frequent terms dropped

`...`      Additional parameters to be passed to wordcloud.

## Value

nothing

## Examples

```
if(require(tm)){
  data(SOTU)
  corp <- SOTU
  corp <- tm_map(corp, removePunctuation)
  corp <- tm_map(corp, removePunctuation)
  corp <- tm_map(corp, tolower)
  corp <- tm_map(corp, removeNumbers)
  corp <- tm_map(corp, function(x)removeWords(x, stopwords()))

  term.matrix <- TermDocumentMatrix(corp)
  term.matrix <- as.matrix(term.matrix)
  colnames(term.matrix) <- c("SOTU 2010", "SOTU 2011")
  comparison.cloud(term.matrix, max.words=300, random.order=FALSE)
  commonality.cloud(term.matrix, random.order=FALSE)
}
```

---

comparison.cloud	<i>Plot a comparison cloud</i>
------------------	--------------------------------

---

**Description**

Plot a cloud comparing the frequencies of words across documents.

**Usage**

```
comparison.cloud(term.matrix, scale=c(4, .5), max.words=300, random.order=FALSE,
rot.per=.1, colors=brewer.pal(ncol(term.matrix), "Dark2"), use.r.layout=FALSE, title.size=3, ...)
```

**Arguments**

term.matrix	A term frequency matrix whose rows represent words and whose columns represent documents.
scale	A vector of length 2 indicating the range of the size of the words.
max.words	Maximum number of words to be plotted. least frequent terms dropped
random.order	plot words in random order. If false, they will be plotted in decreasing frequency
rot.per	proportion words with 90 degree rotation
colors	color words from least to most frequent
use.r.layout	if false, then c++ code is used for collision detection, otherwise R is used
title.size	Size of document titles
...	Additional parameters to be passed to text (and strheight, strwidth).

**Details**

Let  $p_{i,j}$  be the rate at which word  $i$  occurs in document  $j$ , and  $p_j$  be the average across documents ( $\sum_i p_{i,j}/ndocs$ ). The size of each word is mapped to its maximum deviation ( $max_i(p_{i,j} - p_j)$ ), and its angular position is determined by the document where that maximum occurs.

**Value**

nothing

**Examples**

```
if(require(tm)){
data(SOTU)
corp <- SOTU
corp <- tm_map(corp, removePunctuation)
corp <- tm_map(corp, removePunctuation)
corp <- tm_map(corp, tolower)
corp <- tm_map(corp, removeNumbers)
corp <- tm_map(corp, function(x)removeWords(x, stopwords()))
```

```

term.matrix <- TermDocumentMatrix(corp)
term.matrix <- as.matrix(term.matrix)
colnames(term.matrix) <- c("SOTU 2010", "SOTU 2011")
comparison.cloud(term.matrix, max.words=300, random.order=FALSE)
commonality.cloud(term.matrix, random.order=FALSE)
}

```

---

SOTU

*United States State of the Union Addresses (2010 and 2011)*


---

### Description

Transcripts of the state of the union speeches. saved as a tm Corpus.

### Usage

```
data(SOTU)
```

---

wordcloud

*Plot a word cloud*


---

### Description

Plot a word cloud

### Usage

```
wordcloud(words, freq, scale=c(4, .5), min.freq=3, max.words=Inf, random.order=TRUE, random.color=FALSE,
rot.per=.1, colors="black", ordered.colors=FALSE, use.r.layout=FALSE, ...)
```

### Arguments

words	the words
freq	their frequencies
scale	A vector of length 2 indicating the range of the size of the words.
min.freq	words with frequency below min.freq will not be plotted
max.words	Maximum number of words to be plotted. least frequent terms dropped
random.order	plot words in random order. If false, they will be plotted in decreasing frequency
random.color	choose colors randomly from the colors. If false, the color is chosen based on the frequency
rot.per	proportion words with 90 degree rotation
colors	color words from least to most frequent
ordered.colors	if true, then colors are assigned to words in order
use.r.layout	if false, then c++ code is used for collision detection, otherwise R is used
...	Additional parameters to be passed to text (and strheight, strwidth).

**Value**

nothing

**See Also**

[text](#)

**Examples**

```

if(require(tm)){
  data(crude)
  crude <- tm_map(crude, removePunctuation)
  crude <- tm_map(crude, function(x)removeWords(x, stopwords()))
  tdm <- TermDocumentMatrix(crude)
  m <- as.matrix(tdm)
  v <- sort(rowSums(m),decreasing=TRUE)
  d <- data.frame(word = names(v),freq=v)

  wordcloud(d$word,d$freq)

  #A bigger cloud with a minimum frequency of 2
  wordcloud(d$word,d$freq,c(8, .3),2)

  #Now lets try it with frequent words plotted first
  wordcloud(d$word,d$freq,c(8, .5),2,,FALSE,.1)

  ##### with colors #####
  if(require(RColorBrewer)){

    pal <- brewer.pal(9, "BuGn")
    pal <- pal[-(1:4)]
    wordcloud(d$word,d$freq,c(8, .3),2,,FALSE,,.15,pal)

    pal <- brewer.pal(6, "Dark2")
    pal <- pal[-(1)]
    wordcloud(d$word,d$freq,c(8, .3),2,,TRUE,,.15,pal)

    #random colors
    wordcloud(d$word,d$freq,c(8, .3),2,,TRUE,TRUE,.15,pal)
  }
  ##### with font #####

  wordcloud(d$word,d$freq,c(8, .3),2,,TRUE,,.15,pal,
    vfont=c("gothic english","plain"))

  wordcloud(d$word,d$freq,c(8, .3),2,100,TRUE,,.15,pal,vfont=c("script", "plain"))

  wordcloud(d$word,d$freq,c(8, .3),2,100,TRUE,,.15,pal,vfont=c("serif", "plain"))
}

```

# Index

[commonality.cloud](#), 2

[comparison.cloud](#), 3

[SOTU](#), 4

[text](#), 5

[wordcloud](#), 4