# Tutorial - Text data

## Ariane Ducellier

2023-12-05

```
Load R packages.

library(corrplot)

## corrplot 0.92 loaded

library(RColorBrewer)

library(tm)

## Loading required package: NLP

library(wordcloud)
```

### Make a word cloud

```
# Set number of colors and palette
pal = brewer.pal(6, "RdGy")
# Choose minimum frequency and the range of the size of the words
wordcloud("The objective of this course is to provide students with a comprehensive understanding of da
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
                                                 visualizations
students right cognition
based insights programming
                                                fundamentals COVEr exercises
 understanding objective make creating user clear scales agglot2 informed
            creating user clear scales engagement
networks interactivity tooltips packages Classhooard Stories theory coordinates
                      practices of the project should be projected by the project of the project should be provided by the project of the project should be projected by the project of the proje
                                                using techniques three clues
principlesprovide colors
                       advanced using
                                                                  enabling types handson
                                                             communicate
              visualization
```

To use a list of words and their frequencies.

# Growth America unemployment congress democracy Republicans inequality job economy

To read a text file and preprocess it, before doing the word cloud.

```
file = readLines("../data/syllabus.txt")
doc = Corpus(VectorSource(file))
doc = tm_map(doc, tolower)
## Warning in tm_map.SimpleCorpus(doc, tolower): transformation drops documents
doc = tm_map(doc, removePunctuation)
## Warning in tm_map.SimpleCorpus(doc, removePunctuation): transformation drops
## documents
doc = tm_map(doc, removeNumbers)
## Warning in tm_map.SimpleCorpus(doc, removeNumbers): transformation drops
## documents
doc = tm_map(doc, removeWords, stopwords("english"))
## Warning in tm_map.SimpleCorpus(doc, removeWords, stopwords("english")):
## transformation drops documents
wordcloud(as.character(doc), scale=c(2, 0.5))
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
          interactive
        visualizations
         principles
```

## Make two word clouds

```
files = DirSource("../data/debate/")
data = Corpus(DirSource("../data/debate/"))
```

```
data = tm_map(data, tolower)
data = tm_map(data, removePunctuation)
data = tm_map(data, removeNumbers)
data = tm_map(data, removeWords, c(stopwords("english"), "biden", "trump"))
data = TermDocumentMatrix(data)
data = as.matrix(data)
colnames(data) = c("biden", "trump")
comparison.cloud(data, max.words=100, title.size=2, colors=c("blue", "red"))#brewer.pal(3, "Set1"))
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : came could not be fit on page. It will not be plotted.
## Warning in text.default(x1, y1, words[i], cex = size[i], offset = 0, srt =
## rotWord * : font metrics unknown for Unicode character U+2026
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : closed could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : obamacare could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : look could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : judges could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : november could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : radical could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : never could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : together could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : states could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : ever could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : thing could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : economy could not be fit on page. It will not be plotted.
## Warning in comparison.cloud(data, max.words = 100, title.size = 2, colors =
## c("blue", : gave could not be fit on page. It will not be plotted.
```

```
discredited biden america knows american make tax everybody tax everybody number president able true plan one guy talk able true plan one guy talk able true plan one guy talk able true plan one covid per can estimate the car done don twant said the car done don twant said to good country dollars military forestreally much shut years law see won name trump call seen happened also
```

## Plot correlations between texts

```
data(crude)
data = tm_map(crude, content_transformer(tolower))
data = tm_map(data, removePunctuation)
data = tm_map(data, removeNumbers)
data = tm_map(data, removeWords, stopwords("english"))
data = TermDocumentMatrix(data)
data = as.matrix(data)
crf = cor(data)
corrplot(crf, method = c("ellipse"))
```

