

M8L1 Homework Assignment

Joshua Conte

November 12, 2017

1 M8L1 Homework Assignment

R studio was configured with the following parameters before beginning the project:

```
# clears the console in RStudio
cat("\014")
```

```
# clears environment
rm(list = ls())

# Load required packages
library(RTextTools)
library(tm)
library(wordcloud)
library(SnowballC)
```

1.1 Perform the following tasks:

1.1.1 Create a term by document matrix for the Dr. Seuss quote. Assume each sentence is a new document.

The tm package has a number of operations for Term-Document Matrices (like clustering, classifications, etc.).

```
quote<-"You have brains in your head. You have feet in your shoes. You can steer yourself in any directi
seuss <- strsplit(quote, "[.]")
seuss.corpus <- Corpus(DataframeSource(data.frame(seuss)))
seuss.corpus
```

```
## <<VCorpus>>
## Metadata: corpus specific: 0, document level (indexed): 0
## Content: documents: 5
```

```
inspect(seuss.corpus)
```

```
## <<VCorpus>>
## Metadata: corpus specific: 0, document level (indexed): 0
## Content: documents: 5
##
## [[1]]
## <<PlainTextDocument>>
## Metadata: 7
## Content: chars: 28
##
## [[2]]
## <<PlainTextDocument>>
## Metadata: 7
## Content: chars: 28
##
## [[3]]
## <<PlainTextDocument>>
## Metadata: 7
## Content: chars: 51
##
## [[4]]
```

```
## <<PlainTextDocument>>
## Metadata: 7
## Content: chars: 47
##
## [[5]]
## <<PlainTextDocument>>
## Metadata: 7
## Content: chars: 46
seuss.corpus[1]
```

```
## <<VCorpus>>
## Metadata: corpus specific: 0, document level (indexed): 0
## Content: documents: 1
```

```
writeLines(as.character(seuss.corpus[1]))
```

```
## list(list(content = "You have brains in your head", meta = list(author = character(0), datetimestamp
## list()
## list()
```

```
writeLines(as.character(seuss.corpus[1:3]))
```

```
## list(list(content = "You have brains in your head", meta = list(author = character(0), datetimestamp
## wday = 6, yday = 314, isdst = 0), description = character(0), heading = character(0), id = "2",
## list()
## list()
```

```
# Eliminating Extra Whitespace
```

```
seuss.clean<-tm_map(seuss.corpus, stripWhitespace)
```

```
# Convert to Lower Case
```

```
seuss.clean.lc <- tm_map(seuss.clean, content_transformer(tolower))
```

```
writeLines(as.character(seuss.clean.lc[1]))
```

```
## list(list(content = "you have brains in your head", meta = list(author = character(0), datetimestamp
## list()
## list()
```

```
# Remove Stopwords
```

```
seuss.clean <- tm_map(seuss.clean.lc, removeWords, stopwords("english"))
```

```
writeLines(as.character(seuss.clean.lc[1]))
```

```
## list(list(content = "you have brains in your head", meta = list(author = character(0), datetimestamp
## list()
## list()
```

```
# Building a Document-Term Matrix
```

```
seuss.tdm <- TermDocumentMatrix(seuss.clean, control = list(minWordLength = 1))
```

```
seuss.tdm
```

```
## <<TermDocumentMatrix (terms: 12, documents: 5)>>
## Non-/sparse entries: 12/48
## Sparsity : 80%
## Maximal term length: 9
## Weighting : term frequency (tf)
```

1.1.2 Calculate the td-idf for three terms in the text. Assume each sentence is a new document.

This can be completed by inspecting the term by document matrix

```
inspect(seuss.tdm[1:3,1:5])

## <<TermDocumentMatrix (terms: 3, documents: 5)>>
## Non-/sparse entries: 3/12
## Sparsity           : 80%
## Maximal term length: 6
## Weighting          : term frequency (tf)
## Sample            :
##      Docs
## Terms   1 2 3 4 5
##  'll    0 0 0 0 1
##  brains 1 0 0 0 0
##  can    0 0 1 0 0
```

1.1.3 Write a regular expression to segment the Dr. Seuss quote in to seperate sentences.

The easiest and fastest way to do this is using strsplit:

```
strsplit(quote, "[.]" )

## [[1]]
## [1] "You have brains in your head"
## [2] " You have feet in your shoes"
## [3] " You can steer yourself in any direction you choose"
## [4] " You're on your own, and you know what you know"
## [5] " And you are the guy who'll decide where to go"
```

1.1.4 Write a regular expression to tokenize the Dr. Seuss quote.

Tokenization is the task of chopping it up into pieces, called tokens. This is often done by throwing away certain characters, such as punctuation.

```
# Remove punctuation
token<-gsub('[:punct:] ]+', ' ',quote)

# Breakup into seperate chunks
strsplit(token, " ")

## [[1]]
## [1] "You"      "have"      "brains"    "in"        "your"
## [6] "head"     "You"       "have"     "feet"      "in"
## [11] "your"     "shoes"     "You"      "can"       "steer"
## [16] "yourself" "in"       "any"      "direction" "you"
## [21] "choose"   "You"      "re"       "on"        "your"
## [26] "own"      "and"      "you"      "know"      "what"
## [31] "you"      "know"     "And"      "you"       "are"
## [36] "the"      "guy"      "who"      "ll"        "decide"
## [41] "where"    "to"       "go"
```

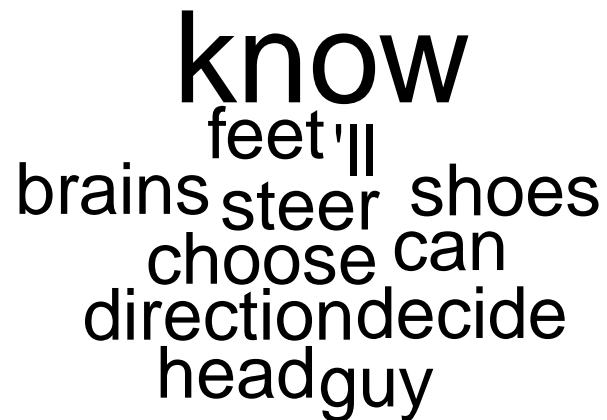
1.1.5 Create a frequency signature for the Dr. Seuss quote. Assume each sentence is a new document.

This can be done with wordcloud:

```
# Operations on Term-Document Matrices  
# Frequent Terms and Associations  
findFreqTerms(seuss.tdm, lowfreq=3)
```

```
## character(0)
```

```
# Word Cloud  
m <- as.matrix(seuss.tdm)  
# calculate the frequency of words  
v <- sort(rowSums(m), decreasing=TRUE)  
myNames <- names(v)  
d <- data.frame(word=myNames, freq=v)  
wordcloud(d$word, d$freq, min.freq=3)
```



know
feet
brains steer shoes
choose can
direction decide
head guy