M8L1\_Assignment

Xueyi Fan

July 8, 2016

# Assignment

Consider the text from Dr. Seuss below:

You have brains in your head. You have feet in your shoes. You can steer yourself in any direction you choose. You're on your own, and you know what you know. And you are the guy who'll decide where to go.

* Dr. Seuss

Perform the following tasks (either by hand or in R):

\* Create a term by document matrix for the Dr. Seuss quote. Assume each sentence is a new documment.   
 \* Calculate the td-idf for three terms in the text. Assume each sentence is a new documment.   
 \* Write a regular expression to segment the Dr. Seuss quote in to seperate sentences.   
 \* Write a regular expression to tokenize the Dr. Seuss quote.   
 \* Create a frequency signature for the Dr. Seuss quote. Assume each sentence is a new documment.

# Answer:

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

library("tm")

## Loading required package: NLP

docMatrix <- c("You have brains in your head.", "You have feet in your shoes.","You can steer yourself in any direction you choose.", "You're on your own, and you know what you know.", "And you are the guy who'll decide where to go.")  
doc.corpus<- Corpus(DataframeSource(data.frame(docMatrix)))  
inspect(doc.corpus)

## <<VCorpus>>  
## Metadata: corpus specific: 0, document level (indexed): 0  
## Content: documents: 5  
##   
## [[1]]  
## <<PlainTextDocument>>  
## Metadata: 7  
## Content: chars: 29  
##   
## [[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

doc.clean <- tm\_map(doc.corpus,stripWhitespace)  
termDocMatrix <- TermDocumentMatrix(doc.clean, control=list(removePunctuation=TRUE, tolower=T, minWordLength=1, stopwords(kind="en")))  
termDocMatrix

## <<TermDocumentMatrix (terms: 24, documents: 5)>>  
## Non-/sparse entries: 32/88  
## Sparsity : 73%  
## Maximal term length: 9  
## Weighting : term frequency (tf)

inspect(termDocMatrix)

## <<TermDocumentMatrix (terms: 24, documents: 5)>>  
## Non-/sparse entries: 32/88  
## Sparsity : 73%  
## Maximal term length: 9  
## Weighting : term frequency (tf)  
##   
## Docs  
## Terms 1 2 3 4 5  
## and 0 0 0 1 1  
## any 0 0 1 0 0  
## are 0 0 0 0 1  
## brains 1 0 0 0 0  
## can 0 0 1 0 0  
## choose 0 0 1 0 0  
## decide 0 0 0 0 1  
## direction 0 0 1 0 0  
## feet 0 1 0 0 0  
## guy 0 0 0 0 1  
## have 1 1 0 0 0  
## head 1 0 0 0 0  
## know 0 0 0 2 0  
## own 0 0 0 1 0  
## shoes 0 1 0 0 0  
## steer 0 0 1 0 0  
## the 0 0 0 0 1  
## what 0 0 0 1 0  
## where 0 0 0 0 1  
## wholl 0 0 0 0 1  
## you 1 1 2 2 1  
## your 1 1 0 1 0  
## youre 0 0 0 1 0  
## yourself 0 0 1 0 0

## Calculate the tf-idf for three terms in the text. Assume each sentence is a new documment.

To calculate the tf-idf for terms "shoes"(Doc2), "know"(Doc4), "you"(Doc1)

f\_td.shoes <- 1  
f\_td.know <- 2  
f\_td.you <- 1  
max\_doc2\_f <- 1  
max\_doc4\_f <- 2  
max\_doc1\_f <- 1  
  
#term "shoes"  
tf\_shoes <- 0.5 + (0.5\*f\_td.shoes)/max\_doc2\_f  
idf\_shoes <- log(5/1)  
tf\_idf\_shoes <- tf\_shoes \* idf\_shoes  
tf\_idf\_shoes

## [1] 1.609438

#term "know"  
  
tf\_know <- 0.5 + (0.5\*f\_td.know)/max\_doc4\_f  
idf\_know <- log(5/2)  
tf\_idf\_know <- tf\_know \* idf\_know  
tf\_idf\_know

## [1] 0.9162907

#term "you"  
tf\_you <- 0.5 + (0.5\*f\_td.you)/max\_doc1\_f  
idf\_you <- log(5/7)  
tf\_idf\_you <- tf\_you \* idf\_you  
tf\_idf\_you

## [1] -0.3364722

Here, I got the tf-idf score of three terms: "shoes", "know", and "you". The score of shoes is 1.609438; the score of know is 0.9162907, the score of you is -0.3364722. It is easy to find the word "you" is least important and the word "shoes" are most important among these three terms.

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

seuss <- "You have brains in your head. You have feet in your shoes. You can steer yourself in any direction you choose. You're on your own, and you know what you know. And you are the guy who'll decide where to go."  
  
docs <- strsplit(seuss, ". ", fixed = T)  
docs

## [[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."

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

words <- strsplit(seuss, "[ ]+", perl=TRUE)  
words

## [[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" "own,"   
## [26] "and" "you" "know" "what" "you"   
## [31] "know." "And" "you" "are" "the"   
## [36] "guy" "who'll" "decide" "where" "to"   
## [41] "go."

## Create a frequency signature for the Dr. Seuss quote. Assume each sentence is a new documment.

#convert the termDoc Matrix to a matrix form  
tdm\_matrix <- as.matrix(termDocMatrix)  
  
#sort the rowsums of the matrix  
tdm\_sort <- sort(rowSums(tdm\_matrix), decreasing = T)  
myNames <- names(tdm\_sort)  
  
#create a data-frame  
tdm\_frame <- data.frame(word = myNames, freq = tdm\_sort)  
tdm\_frame

## word freq  
## you you 7  
## your your 3  
## and and 2  
## have have 2  
## know know 2  
## any any 1  
## are are 1  
## brains brains 1  
## can can 1  
## choose choose 1  
## decide decide 1  
## direction direction 1  
## feet feet 1  
## guy guy 1  
## head head 1  
## own own 1  
## shoes shoes 1  
## steer steer 1  
## the the 1  
## what what 1  
## where where 1  
## wholl wholl 1  
## youre youre 1  
## yourself yourself 1