# Week10 Synchronous Code

#

#

#install.packages("tm")

library(tm)

#install.packages("wordcloud")

library(wordcloud)

#

#

# Process in the MLK speech

#

mlk <- readLines("MLK.txt") # read in text file "MLK"

mlk <- mlk[which(mlk != "")] # remove all blank lines in the text

mlk[1:3]

# Create a term matrix

# interprets each element of the "mlk" as a document and create a vector source

words.vec <- VectorSource(mlk)

# create a Corpus, a "Bag of Words"

words.corpus <- Corpus(words.vec)

# first step transformation: make all of the letters in "words.corpus" lowercase

words.corpus <- tm\_map(words.corpus, content\_transformer(tolower))

# second step transformation: remove the punctuation in "words.corpus"

words.corpus <- tm\_map(words.corpus, removePunctuation)

# third step transformation: remove numbers in "words.corpus"

words.corpus <- tm\_map(words.corpus, removeNumbers)

# final step transformation: take out the "stop" words, such as "the", "a" and "at"

words.corpus <- tm\_map(words.corpus, removeWords, stopwords("english"))

#

# create a term-document matrix "tdm"

tdm <- TermDocumentMatrix(words.corpus)

# view term-document matrix "tdm"

tdm

#

# Create a list of counts for each word

# convert tdm into a matrix called "m"

m <- as.matrix(tdm)

m[1:10,]

# create a list of counts for each word named "wordCounts"

wordCounts <- rowSums(m)

wordCounts[1:10]

# sum the total number of words and store the value to "totalWords"

totalWords <- sum(wordCounts)

totalWords

# create a vector "words" that contains all the words in "wordCounts"

words <- names(wordCounts)

head(words)

##numDiffWords<-length(wordCounts)

##numDiffWords

# sort words in "wordCounts" by frequency

wordCounts <- sort(wordCounts, decreasing=TRUE)

# check the first several items in "wordCounts" to see if it is built correctly

head(wordCounts)

#

# Build Word Cloud

#

cloudFrame<-data.frame(word=names(wordCounts),freq=wordCounts)

cloudFrame[1:10,]

wordcloud(cloudFrame$word,cloudFrame$freq)

wordcloud(names(wordCounts),wordCounts,min.freq=3,max.words=50,rot.per=.35,colors=brewer.pal(8,"Dark2"))

#

#

# Sentiment / Affinity Score

#

# Read Afffinity file

#

AFINN<-read.delim("HW10AFINN111.txt",sep="\t",header = FALSE)

str(AFINN)

head(AFINN)

colnames(AFINN)<-c("Word", "Score")

AFINN[1:10,]

#

# join the df match with AFINN by "word" col in match and "Word" col in AFINN

#

AFINN[40,]

cloudFrame[1,]

mergedTable<-merge(cloudFrame,AFINN,by.x="word",by.y="Word")

mergedTable[1:10,]

str(mergedTable)

#

overallScore<-sum(mergedTable$freq\*mergedTable$Score)

overallScore

overallScore/totalWords

#

#

#