

Project1__presidents__inauguation

Hongyang Yang-hy2500

September 19, 2017

Section 0: check and install needed packages. Load the libraries and functions.

```
#packages.used=c("rvest", "tibble", "qdap",  
#               "sentimentr", "gplots", "dplyr",  
#               "tm", "syuzhet", "factoextra",  
#               "beeswarm", "scales", "RColorBrewer",  
#               "RANN", "tm", "topicmodels", "wordcloud",  
#               "RColorBrewer", "tidytext")  
  
# check packages that need to be installed.  
#packages.needed=setdiff(packages.used,  
#                         intersect(installed.packages()[,1],  
#                         packages.used))  
# install additional packages  
#if(length(packages.needed)>0){  
#  install.packages(packages.needed, dependencies = TRUE)  
#}  
  
library(wordcloud)  
library(RColorBrewer)  
library(dplyr)  
library(tidytext)  
  
# load packages  
library("rvest")  
library("tibble")  
library("qdap")  
library("sentimentr")  
library("gplots")  
library("dplyr")  
library("tm")  
library("syuzhet")  
library("factoextra")  
library("beeswarm")  
library("scales")  
library("RColorBrewer")  
library("RANN")  
library("tm")  
library("topicmodels")  
library("wordcloud")  
library("RColorBrewer")  
library("tidytext")
```

```
#library("SnowballC")
#library("wordnet")
#library("ggplot2")
#library("qdap")
#library("NLP")
#library("openNLP")
#library("cluster")
#library("fpc")

source("../lib/plotstacked.R")
source("../lib/speechFuncs.R")
```

This notebook was prepared with the following environmental settings.

```
print(R.version)

##
## platform      _
## arch          x86_64-mingw32
## os            mingw32
## system        x86_64, mingw32
## status
## major         3
## minor         4.1
## year          2017
## month         06
## day           30
## svn rev       72865
## language      R
## version.string R version 3.4.1 (2017-06-30)
## nickname      Single Candle
```

Section 1: Data Preperation

Step 1: scrap speech URLs from <http://www.presidency.ucsb.edu/>.

Following the example of Jerid Francom, we used Selectorgadget to choose the links we would like to scrap. For this project, we selected all inaugural addresses of past presidents

```
### Inauguaral speeches
main.page <- read_html(x = "http://www.presidency.ucsb.edu/inaugurals.php")
# Get link URLs
# f.speechlinks is a function for extracting links from the list of speeches.
inaug=f.speechlinks(main.page)
#head(inaug)
as.Date(inaug[,1], format="%B %e, %Y")

## [1] "1789-04-30" "1793-03-04" "1797-03-04" "1801-03-04" "1805-03-04"
## [6] "1809-03-04" "1813-03-04" "1817-03-04" "1821-03-04" "1825-03-04"
## [11] "1829-03-04" "1833-03-04" "1837-03-04" "1841-03-04" "1845-03-04"
## [16] "1849-03-05" "1853-03-04" "1857-03-04" "1861-03-04" "1865-03-04"
## [21] "1869-03-04" "1873-03-04" "1877-03-05" "1881-03-04" "1885-03-04"
```

```
## [26] "1889-03-04" "1893-03-04" "1897-03-04" "1901-03-04" "1905-03-04"
## [31] "1909-03-04" "1913-03-04" "1917-03-04" "1921-03-04" "1925-03-04"
## [36] "1929-03-04" "1933-03-04" "1937-01-20" "1941-01-20" "1945-01-20"
## [41] "1949-01-20" "1953-01-20" "1957-01-21" "1961-01-20" "1965-01-20"
## [46] "1969-01-20" "1973-01-20" "1977-01-20" "1981-01-20" "1985-01-21"
## [51] "1989-01-20" "1993-01-20" "1997-01-20" "2001-01-20" "2005-01-20"
## [56] "2009-01-20" "2013-01-21" "2017-01-20" NA

inaug=inaug[-nrow(inaug),] # remove the last line, irrelevant due to error.
```

Step 2: Using speech metadata posted on <http://www.presidency.ucsb.edu/>, we prepared CSV data sets for the speeches we will scrap.

```
inaug.list=read.csv("../data/inauglist.csv", stringsAsFactors = FALSE)
```

Step 3: we prepared CSV data sets for presidents' perosnality, education background, and economic recession.

<http://thesixteentypes.tumblr.com/post/60372674743/personality-types-of-all-us-presidents> https://en.wikipedia.org/wiki/List_of_recessions_in_the_United_States#Great_Depression_onward

If the president is introverted dominated set the personality to -1, extroverted dominated set the personality to 1. for example, Barack Obama is ENFP-Extraversion (E), Intuition (N), Thinking (T), Perception (P), that means he has an extroverted dominated personality.

If the inaugural speech happened during an economic recession, set the recession to 1. for example, the 2008 financial crisis happened during Dec 2007 – June 2009, Barack Obama delivered his 1st term inaugural speech on Jan, 2009. So his inaugural speech happened during an economic recession.

If the presidents get any real threat assasination attempt.

```
personality_info=read.csv("../data/personality_info.csv", stringsAsFactors = FALSE,header=TRUE)

personality=personality_info$Personality
recession=personality_info$Recession

inaug.list=cbind(inaug.list,personality,recession)
```

We assemble all scrapped speeches into one list. Note here that we don't have the full text yet, only the links to full text transcripts.

Step 4: scrap the texts of speeches from the speech URLs.

```
inaug.list$type=c(rep("inaug", nrow(inaug.list)))
inaug.list=cbind(inaug.list, inaug)
```

Based on the list of speeches, we scrap the main text part of the transcript's html page. For simple html pages of this kind, Selectorgadget is very convenient for identifying the html node that `rvest` can use to scrap its content. For reproducibility, we also save our scrapped speeches into our local folder as individual speech files.

```

# Loop over each row in speech.list
inaug.list$fulltext=NA
for(i in seq(nrow(inaug.list))) {
  text <- read_html(inaug.list$urls[i]) %>% # load the page
    html_nodes(".displaytext") %>% # isolate the text
    html_text() # get the text
  inaug.list$fulltext[i]=text
  # Create the file name
  filename <- paste0("../data/fulltext/",
                    inaug.list$type[i],
                    inaug.list$File[i], "-",
                    inaug.list$Term[i], ".txt")
  sink(file = filename) %>% # open file to write
  cat(text) # write the file
  sink() # close the file
}

```

Step 5: data Processing — generate list of sentences

We will use sentences as units of analysis for this project, as sentences are natural language units for organizing thoughts and ideas. For each extracted sentence, we apply sentiment analysis using NRC sentiment lexicon. “The NRC Emotion Lexicon is a list of English words and their associations with eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive). The annotations were manually done by crowdsourcing.”

We assign an sequential id to each sentence in a speech (`sent.id`) and also calculated the number of words in each sentence as *sentence length* (`word.count`).

```

sentence.list=NULL
for(i in 1:nrow(inaug.list)){
  sentences=sent_detect(inaug.list$fulltext[i],
                      endmarks = c("?", ".", "!", "|", ";"))
  if(length(sentences)>0){
    emotions=get_nrc_sentiment(sentences)
    word.count=word_count(sentences)
    # colnames(emotions)=paste0("emo.", colnames(emotions))
    # in case the word counts are zeros?
    emotions=diag(1/(word.count+0.01))%*%as.matrix(emotions)
    sentence.list=rbind(sentence.list,
                      cbind(inaug.list[i,-ncol(inaug.list)],
                          sentences=as.character(sentences),
                          word.count,
                          emotions,
                          sent.id=1:length(sentences)
                        )
    )
  }
}

```

Some non-sentences exist in raw data due to erroneous extra end-of sentence marks.

```

sentence.list=
  sentence.list%>%
  filter(!is.na(word.count))

```

Section 2: What words can Presidents' personality tell us?

Words/Sentences analysis

First, let's calculate extroverted v.s. introverted total sentences, total words, and average words per sentences

```
extroverted_list=sentence.list[sentence.list$personality==1,]
introverted_list=sentence.list[sentence.list$personality==1,]

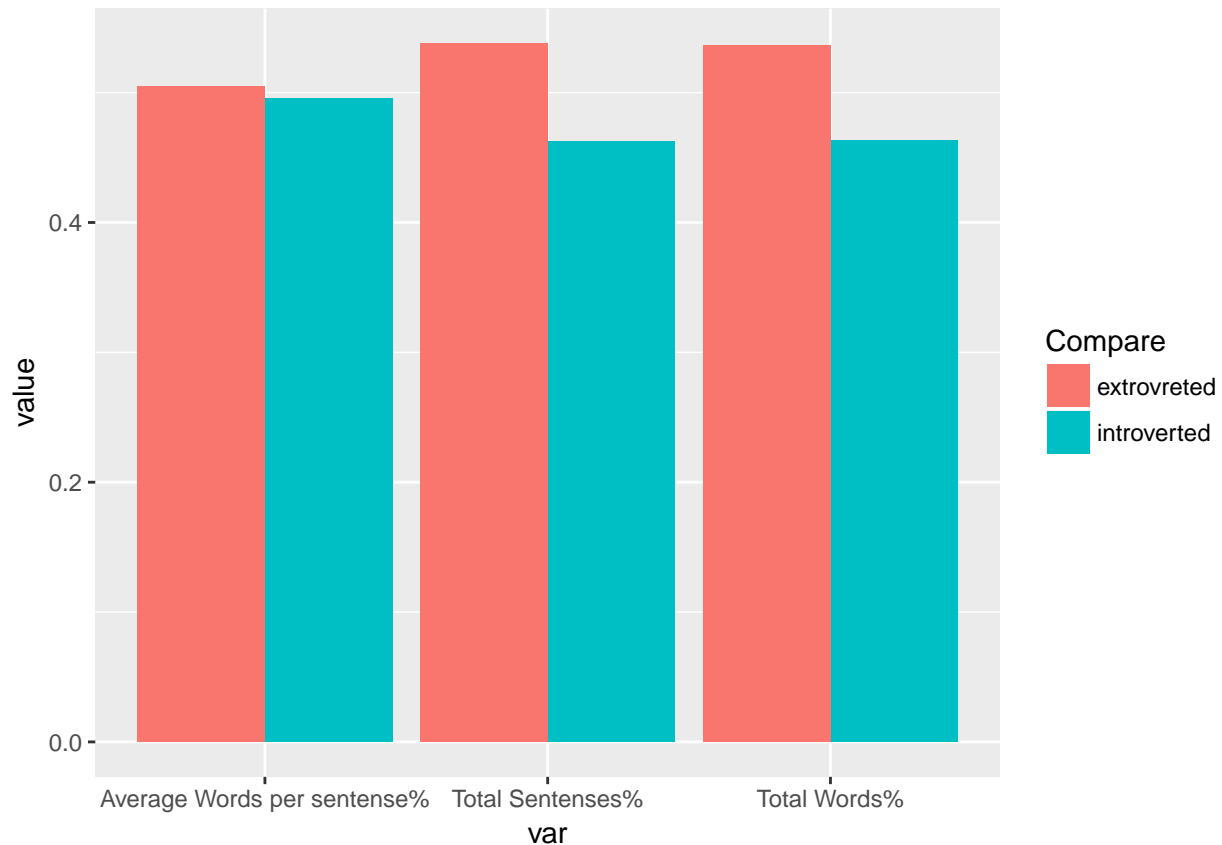
# number of sentences
extrovreted_sentences<-tapply(extroverted_list$sentences, extroverted_list$File, length)
# average words per sentence
extrovreted_average_words<-round(tapply(extroverted_list$word.count, extroverted_list$File, mean),0)
extrovreted_words<-tapply(extroverted_list$word.count, extroverted_list$File, sum)
extrovreted_count<-data.frame(extrovreted_sentences, extrovreted_average_words, extrovreted_words)

introverted_sentences<-tapply(introverted_list$sentences, introverted_list$File, length)
# average words per sentence
introverted_average_words<-round(tapply(introverted_list$word.count, introverted_list$File, mean),0)
introverted_words<-tapply(introverted_list$word.count, introverted_list$File, sum)
introverted_count<-data.frame(introverted_sentences, introverted_average_words, introverted_words)

e=colMeans(extrovreted_count)
i=colMeans(introverted_count)
dt=matrix(c(rep("extrovreted",3),rep("introverted",3)),nrow=6,ncol=1)
total_sentences=e[[1]]+i[[1]]
total_words=e[[3]]+i[[3]]
total_avg_words=e[[2]]+i[[2]]

dt=as.data.frame(dt)
colnames(dt)<-"Compare"
dt$var<-c(rep(c("Total Sentences%", "Total Words%", "Average Words per sentence%"),2))
dt$value<-c(e[[1]]/total_sentences,e[[3]]/total_words,e[[2]]/total_avg_words
            ,i[[1]]/total_sentences,i[[3]]/total_words,i[[2]]/total_avg_words)

#png('./figs/Compare.png')
ggplot(dt, aes(x = var, y = value, fill = Compare))+
  geom_bar(stat="identity", position=position_dodge())
```



```
#dev.off()
```

We can briefly conclude that an extroverted personality dominated president is more talkative than an introverted personality dominated president.

Clustering analysis

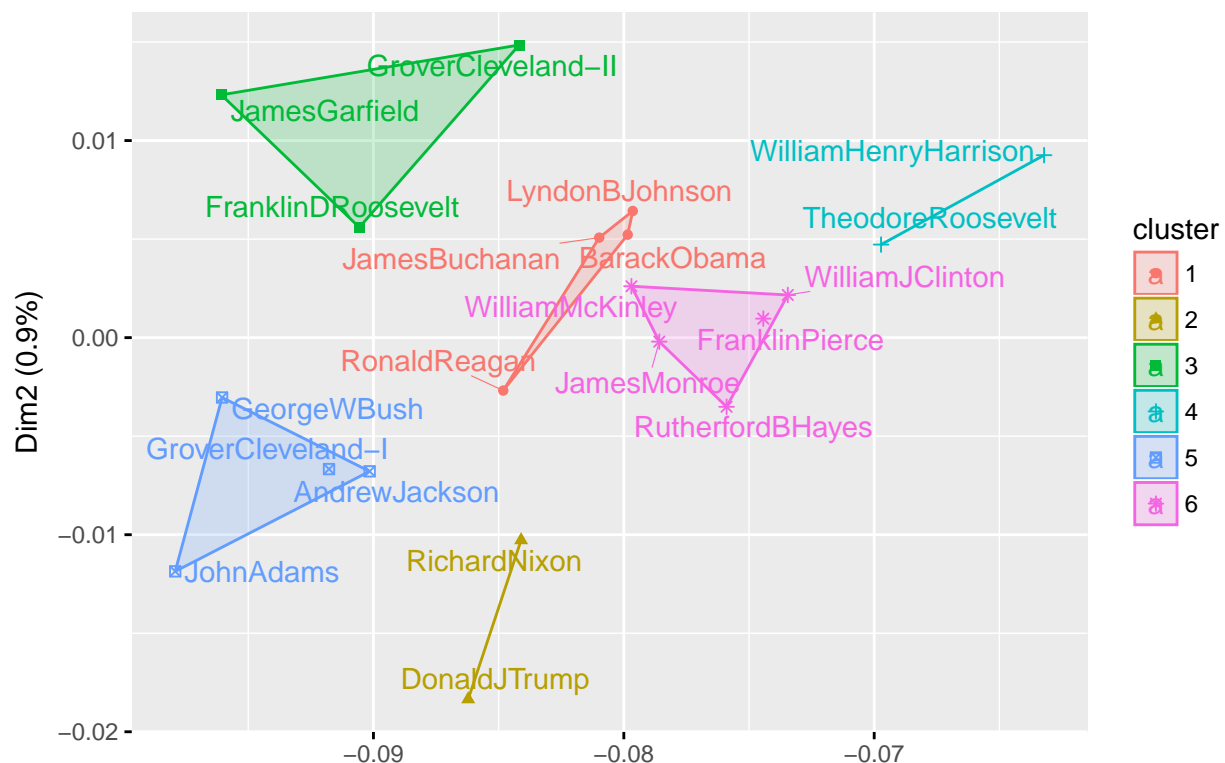
Second, extroverted people are usually easy to identify and compare, so let's use Kmeans to cluster extroverted presidents' speeches to see if any presidents have the same talking style.

```
#extrovreted
presid.summary=tbl_df(extrovreted_list)%>%
  #filter(File%in%sentence.list[sentence.list$personality==1,])%>%
  #group_by(paste0(type, File))%>%
  group_by(File)%>%
  summarise(
    anger=mean(anger),
    anticipation=mean(anticipation),
    disgust=mean(disgust),
    fear=mean(fear),
    joy=mean(joy),
    sadness=mean(sadness),
    surprise=mean(surprise),
    trust=mean(trust)
    #negative=mean(negative),
    #positive=mean(positive)
```

```
)

presid.summary=as.data.frame(presid.summary)
rownames(presid.summary)=as.character((presid.summary[,1]))
km.res=kmeans(presid.summary[,,-1], iter.max=200,
              6)
#png('../figs/fviz_cluster.png')
fviz_cluster(km.res,
             stand=F, repel= TRUE,
             data = presid.summary[,,-1], xlab="", xaxt="n",
             show.clust.cent=FALSE)
```

Cluster plot



I would say that Donald Trump and Richard Nixon have the same talking style, they are common in their ambition, Paranoia, and Media Distrust. They are both alpha male, have strange behavior and unwelcomed by most people.

Barack Obama and Ronald Reagan have a lot in common. Their personality are both attractive, they like to encourage people and tell jokes, their speeches are really impressive and influential.

Visualize president's sentence

We notice that the sentences in inaugural speeches are longer than those in nomination acceptance speeches.

```
extroverted_list$File=factor(extroverted_list$File)
extroverted_list$FileOrdered=reorder(extroverted_list$File,
                                     extroverted_list$word.count,
                                     mean,
```

```

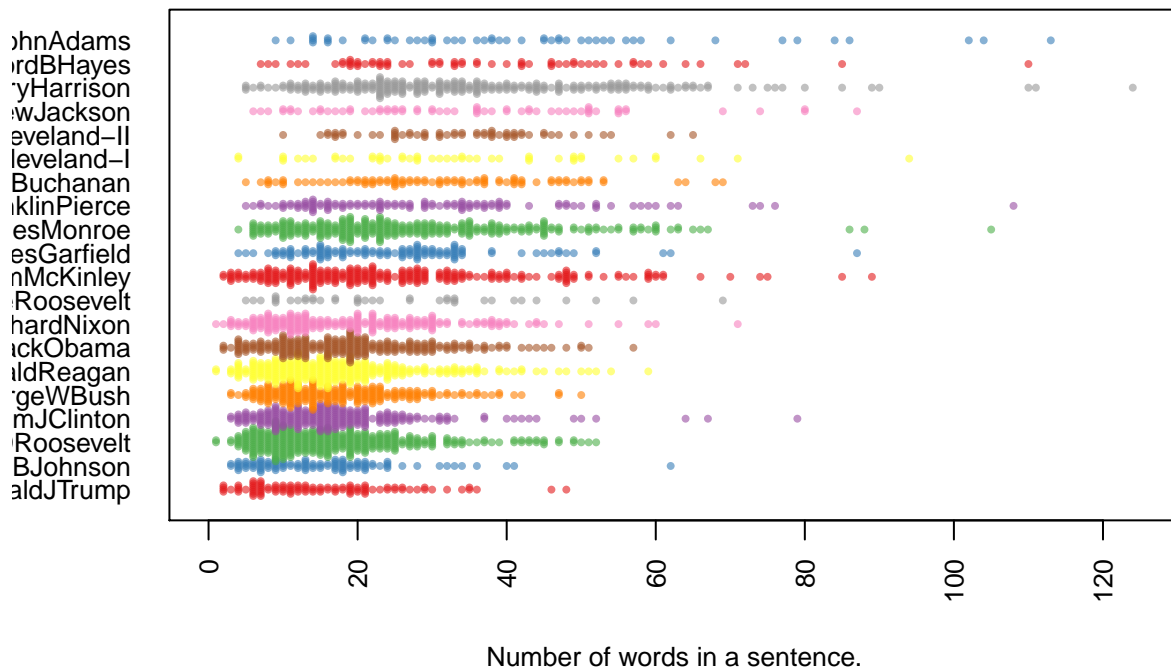
order=T)

#png('../figs/beeswarm_extroverted.png')

beeswarm(word.count~FileOrdered,
  data=extroverted_list,
  horizontal = TRUE,
  pch=16, col=alpha(brewer.pal(9, "Set1"), 0.6),
  cex=0.55, cex.axis=0.8, cex.lab=0.8,
  spacing=5/nlevels(extroverted_list$FileOrdered),
  las=2, ylab="", xlab="Number of words in a sentence.",
  main="Extroverted Presidents' Speeches")

```

Extroverted Presidents' Speeches



```

introverted_list$File=factor(introverted_list$File)
introverted_list$FileOrdered=reorder(introverted_list$File,
  introverted_list$word.count,
  mean,
  order=T)

#png('../figs/beeswarm_introverted.png')

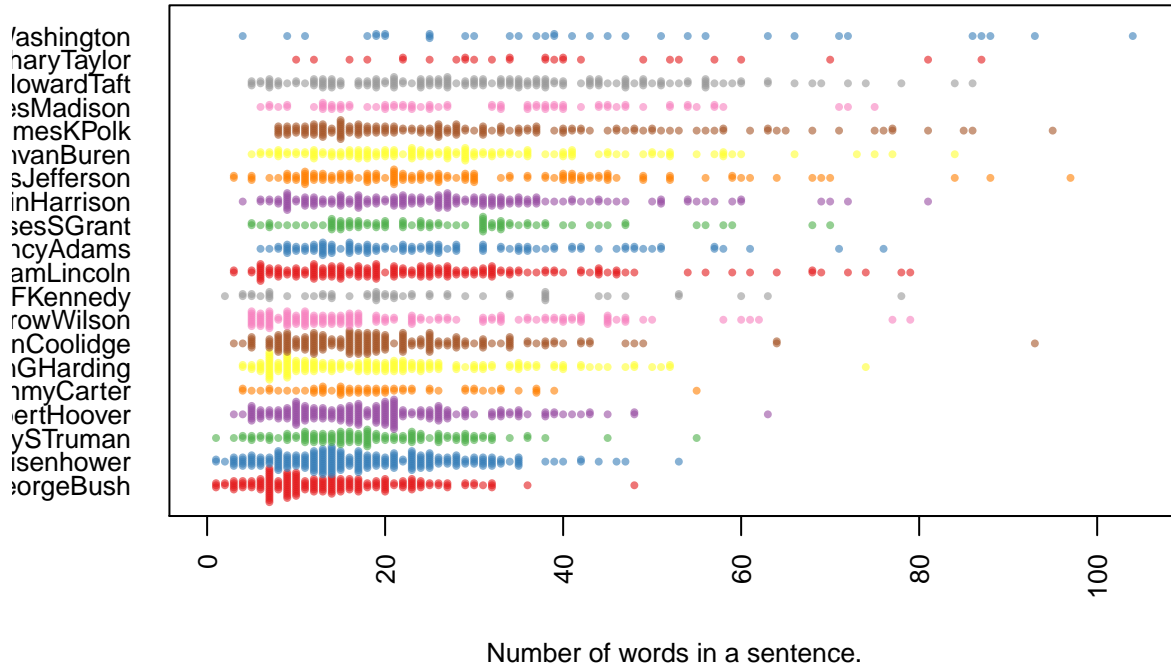
beeswarm(word.count~FileOrdered,
  data=introverted_list,
  horizontal = TRUE,
  pch=16, col=alpha(brewer.pal(9, "Set1"), 0.6),
  cex=0.55, cex.axis=0.8, cex.lab=0.8,
  spacing=5/nlevels(introverted_list$FileOrdered),

```



```
las=2, ylab="", xlab="Number of words in a sentence.",
main="Introverted Presidents' Speeches")
```

Introverted Presidents' Speeches



We notice that there are only one sentences that are longer than 100 words in introverted presidents' speeches, but there are few sentences that are longer than 100 words in extroverted presidents' speeches.

What words did extroverted/introverted Presidents say?

```
folder.path = "../data/fulltext/"
speeches = list.files(path = folder.path, pattern = "*.txt")

#only get inaugural speeches
speeches = speeches[substr(speeches, 1, 5) == 'inaug']

prez.out = substr(speeches, 6, nchar(speeches)-4)

introverted_index = NULL
speeches.president = substr(speeches, 6, nchar(speeches)-6)

#length.speeches = rep(NA, length(speeches))
ff.all <- Corpus(DirSource(folder.path, encoding = "UTF-8"))
ff.all <- tm_map(ff.all, stripWhitespace)
ff.all <- tm_map(ff.all, content_transformer(tolower))
ff.all <- tm_map(ff.all, removeWords, stopwords("english"))
```

```

ff.all<-tm_map(ff.all, removeWords, character(0))
ff.all<-tm_map(ff.all, removePunctuation)

dtm <- DocumentTermMatrix(ff.all,
                           control = list(weighting =
                                           function(x)
                                             weightTfIdf(x, normalize =
                                                           FALSE),
                                           stopwords = TRUE))

ff.dtm=tidy(dtm)

#tdm.overall=summarise(group_by(tdm.tidy, term), sum(count))

#match speeches to personality
personality.list=data.frame(File=inaug.list$File,Term=inaug.list$Term,personality=inaug.list$personality)

speeches.personality=cbind(personality.list[order(personality.list$File),],prez.out)

personality.index=which(speeches.personality$personality==1)

```

Let's compare some famous extroverted and introverted presidents' words. Do you think an extroverted president is more likely to use positive emotion words, and an introverted president is more likely to use rational or negative emotion words?

bleed borders flush
victories protected
jobs stops stir
left dreams glorious
Obama america today
transferring loyalty
trillions across
factories arrives
righteous breath
beautiful
ripped redistributed
fill back

bar moments principle sheet
opinion minds fellow citizens fold
morals limits covered
licentiousness
due defamation
whatsoever
fellow citizens
trusted authorities worn
inspires false expenses
sap retire lessen reason
bridle experiment billows
diffusion



Sentiment Analysis of Inauguration speeches

In the sentiment analysis part, we want to measure what is the proportion of the sentiment was positive or negative during the whole speech process.

DonaldJTrump



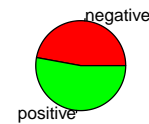
ThomasJefferson



FranklinDRoosevelt



AbrahamLincoln



We can see that Donald Trump (extroverted) has more positive words than Thomas Jefferson (introverted), and William Clinton (extroverted) has more positive words than Thomas Jefferson (Abraham Lincoln).

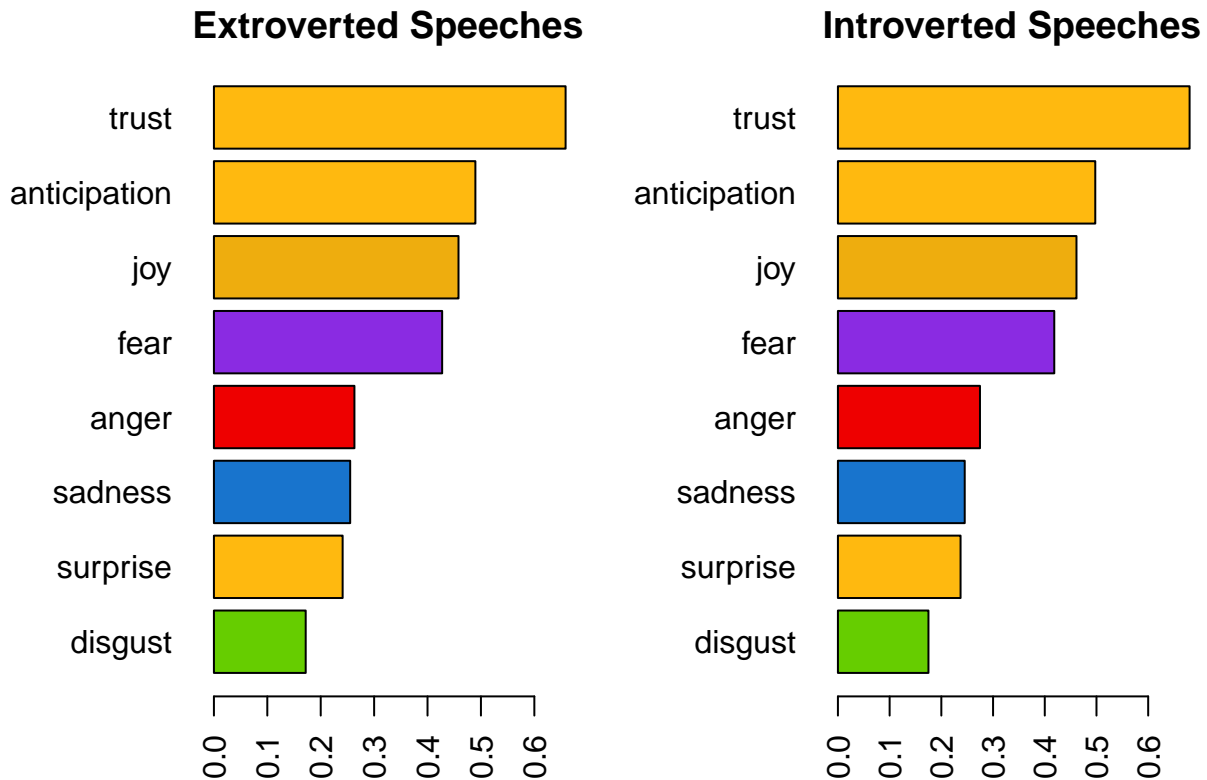
Compare of extroverted/introverted presidents' emotions in general

```
#png('../figs/barchart_emotions.png')

par(mfrow=c(1,2))

par(mar=c(4, 6, 2, 1))
extroverted_emo.means=colMeans(select(extroverted_list, anger:trust)>0.01)
col.use=c("red2", "darkgoldenrod1",
          "chartreuse3", "blueviolet",
          "darkgoldenrod2", "dodgerblue3",
          "darkgoldenrod1", "darkgoldenrod1")
barplot(extroverted_emo.means[order(extroverted_emo.means)], las=2, col=col.use[order(extroverted_emo.m

introverted_emo.means=colMeans(select(introverted_list, anger:trust)>0.0)
col.use=c("red2", "darkgoldenrod1",
          "chartreuse3", "blueviolet",
          "darkgoldenrod2", "dodgerblue3",
          "darkgoldenrod1", "darkgoldenrod1")
barplot(introverted_emo.means[order(introverted_emo.means)], las=2, col=col.use[order(introverted_emo.m
```



It seems they have little difference over using the positive and negative words.

What about under some pressure like an economic recession?

What are the most used terms in an economic recession?

Compare of extroverted/introverted presidents' emotions under economic recession

```
extroverted_recession=sentence.list[sentence.list$personality==1&sentence.list$recession==1,]
introverted_recession=sentence.list[sentence.list$personality==1&sentence.list$recession==1,]
```

```
#png('../figs/barchart_emotions.png')
```

```
par(mfrow=c(1,2))
```

```
par(mar=c(4, 6, 2, 1))
```

```
extroverted_emo.means=colMeans(select(extroverted_recession, anger:trust)>0.01)
```

```
col.use=c("red2", "darkgoldenrod1",
          "chartreuse3", "blueviolet",
          "darkgoldenrod2", "dodgerblue3",
          "darkgoldenrod1", "darkgoldenrod1")
```

```
barplot(extroverted_emo.means[order(extroverted_emo.means)], las=2, col=col.use[order(extroverted_emo.m
```

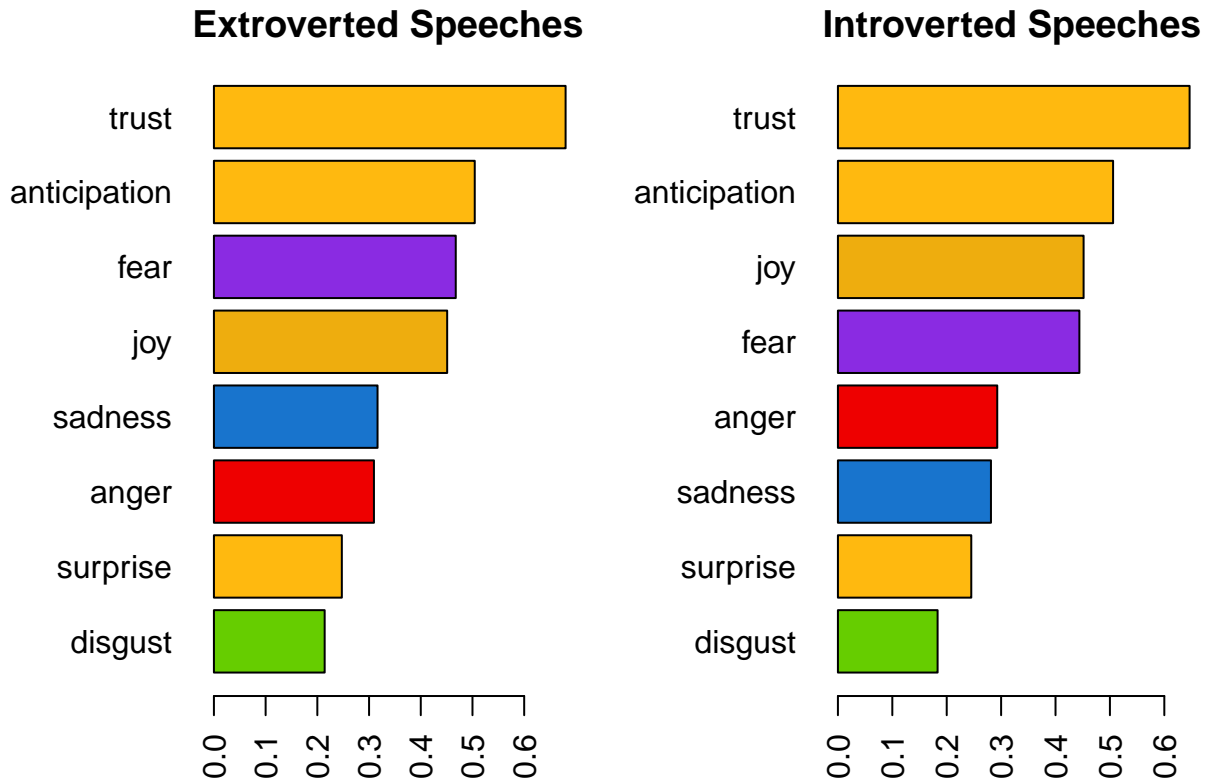
```
introverted_emo.means=colMeans(select(introverted_recession, anger:trust)>0.01)
```

```
col.use=c("red2", "darkgoldenrod1",
```

```

"chartreuse3", "blueviolet",
"darkgoldenrod2", "dodgerblue3",
"darkgoldenrod1", "darkgoldenrod1")
barplot(introverted_emo.means[order(introverted_emo.means)], las=2, col=col.use[order(introverted_emo.m

```



Extroverted presidents put fear before joy under economic recession. That may suggest an extroverted person like to motivate or influence people when there is a pressure.

Section 3: Topic modeling for extroverted presidents during economic recession.

Extroverted presidents in recession

For topic modeling, we prepare a corpus of sentence snippets as follows. For each speech, we start with sentences and prepare a snippet with a given sentence with the flanking sentences.

```

corpus.list=extroverted_recession[2:(nrow(extroverted_recession)-1), ]
sentence.pre=extroverted_recession$sentences[1:(nrow(extroverted_recession)-2)]
sentence.post=extroverted_recession$sentences[3:(nrow(extroverted_recession)-1)]
corpus.list$snippets=paste(sentence.pre, corpus.list$sentences, sentence.post, sep=" ")
rm.rows=(1:nrow(corpus.list))[corpus.list$sent.id==1]
rm.rows=c(rm.rows, rm.rows-1)
corpus.list=corpus.list[-rm.rows, ]

```

Text mining

```
docs <- Corpus(VectorSource(corpus.list$snippets))
writeLines(as.character(docs[[sample(1:nrow(corpus.list), 1)]]))
```

To-day the executive branch of the Government is transferred to new keeping. But this is still the G

Text basic processing

Adapted from <https://eight2late.wordpress.com/2015/09/29/a-gentle-introduction-to-topic-modeling-using-r/>.

```
#remove potentially problematic symbols
docs <- tm_map(docs, content_transformer(tolower))

#remove punctuation
docs <- tm_map(docs, removePunctuation)

#Strip digits
docs <- tm_map(docs, removeNumbers)

#remove stopwords
docs <- tm_map(docs, removeWords, stopwords("english"))

meaningless <- c("all", "can", "say", "one", "way", "use", "also", "howev", "tell", "will"
, "much", "need", "take", "tend", "even", "like", "particular", "rather", "said", "get"
, "well", "make", "ask", "come", "end", "first", "two", "often", "may", "might",
"see", "someth", "thing", "point", "post", "look", "right", "now", "think", "anoth", "yes",
"day", "quit", "sinc", "attempt", "bit", "entir", "lot", "must", "shall", "from", "are", "its", "no"
"our", "that", "the", "their", "this", "which", "with")
docs <- tm_map(docs, removeWords, meaningless)

#remove whitespace
docs <- tm_map(docs, stripWhitespace)

#Stem document
docs <- tm_map(docs, stemDocument)
```

Topic modeling

Generate document-term matrices.

```
dtm <- DocumentTermMatrix(docs)
#convert rownames to filenames#convert rownames to filenames
rownames(dtm) <- paste(corpus.list$type, corpus.list$File,
                      corpus.list$Term, corpus.list$sent.id, sep="_")

rowTotals <- apply(dtm, 1, sum) #Find the sum of words in each Document

dtm <- dtm[rowTotals > 0, ]
corpus.list = corpus.list[rowTotals > 0, ]
```


Run LDA

```
#Set parameters for Gibbs sampling
burnin <- 4000
iter <- 2000
thin <- 500
seed <-list(2003,5,63,100001,765)
nstart <- 5
best <- TRUE

#Number of topics
k <- 15

#Run LDA using Gibbs sampling
ldaOut <-LDA(dtm, k, method="Gibbs", control=list(nstart=nstart,
                                                  seed = seed, best=best,
                                                  burnin = burnin, iter = iter,
                                                  thin=thin))

#write out results
#docs to topics
ldaOut.topics <- as.matrix(topics(ldaOut))
table(c(1:k, ldaOut.topics))

##
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
## 54 102 79 76 69 126 57 79 54 73 57 79 79 99 54

#write.csv(ldaOut.topics,file=paste("../out/LDAGibbs",k,"DocsToTopics.csv"))

#top 6 terms in each topic
ldaOut.terms <- as.matrix(terms(ldaOut,20))
#write.csv(ldaOut.terms,file=paste("../out/LDAGibbs",k,"TopicsToTerms.csv"))

#probabilities associated with each topic assignment
topicProbabilities <- as.data.frame(ldaOut@gamma)
#write.csv(topicProbabilities,file=paste("../out/LDAGibbs",k,"TopicProbabilities.csv"))

terms.beta=ldaOut@beta
terms.beta=scale(terms.beta)
topics.terms=NULL
for(i in 1:k){
  topics.terms=rbind(topics.terms, ldaOut@terms[order(terms.beta[i,], decreasing = TRUE)[1:7]])
}
topics.terms

##      [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] "measur" "error" "proclaim" "prepar" "month"
## [2,] "believ" "togeth" "capac" "disast" "era"
## [3,] "foreign" "cours" "neighbor" "settlement" "washington"
## [4,] "privileg" "inde" "path" "depriv" "district"
## [5,] "prevent" "price" "accumul" "oppress" "beget"
## [6,] "work" "profit" "hero" "cost" "answer"
## [7,] "past" "extens" "north" "west" "everywher"
## [8,] "oath" "respons" "fulli" "humbl" "guid"
```

```
## [9,] "territori" "vast" "understood" "decid" "navi"
## [10,] "legisl" "local" "hous" "judiciari" "enlighten"
## [11,] "love" "like" "beauti" "sink" "decay"
## [12,] "treasuri" "sound" "surplus" "taxat" "loan"
## [13,] "depart" "appear" "exercis" "branch" "mode"
## [14,] "hope" "face" "children" "endur" "leadership"
## [15,] "import" "appoint" "desir" "behalf" "least"
## [,6] [,7]
## [1,] "theori" "proceed"
## [2,] "group" "run"
## [3,] "allianc" "field"
## [4,] "consol" "one"
## [5,] "inconsist" "settler"
## [6,] "undertaken" "pride"
## [7,] "revis" "geograph"
## [8,] "sphere" "impress"
## [9,] "slaveri" "admiss"
## [10,] "suppos" "veto"
## [11,] "name" "bold"
## [12,] "balanc" "specul"
## [13,] "latter" "defect"
## [14,] "met" "hour"
## [15,] "bank" "reward"
```

ldaOut.terms

```
## Topic 1 Topic 2 Topic 3 Topic 4 Topic 5
## [1,] "confid" "govern" "nation" "citizen" "peopl"
## [2,] "upon" "good" "great" "institut" "american"
## [3,] "measur" "believ" "peac" "subject" "public"
## [4,] "made" "men" "world" "american" "govern"
## [5,] "mani" "without" "respect" "never" "interest"
## [6,] "opinion" "chang" "polici" "free" "demand"
## [7,] "still" "individu" "foreign" "right" "civil"
## [8,] "carri" "life" "relat" "form" "busi"
## [9,] "understand" "econom" "histori" "principl" "labor"
## [10,] "principl" "democraci" "justic" "member" "economi"
## [11,] "far" "problem" "honor" "enjoy" "everi"
## [12,] "adopt" "alway" "fellowcitizen" "domest" "care"
## [13,] "greatest" "among" "sinc" "equal" "use"
## [14,] "exist" "order" "cours" "possess" "industri"
## [15,] "aid" "human" "exempl" "benefit" "enterpris"
## [16,] "former" "truth" "mutual" "realiz" "support"
## [17,] "found" "moral" "disput" "perfect" "fair"
## [18,] "fact" "forc" "adjust" "person" "among"
## [19,] "observ" "master" "emerg" "concern" "scheme"
## [20,] "motiv" "constant" "war" "countrymen" "lead"
## Topic 6 Topic 7 Topic 8 Topic 9 Topic 10
## [1,] "work" "countri" "everi" "state" "upon"
## [2,] "call" "year" "peopl" "unit" "institut"
## [3,] "live" "peopl" "duti" "union" "congress"
## [4,] "help" "great" "protect" "necessari" "legisl"
## [5,] "men" "progress" "just" "territori" "interest"
## [6,] "america" "law" "less" "war" "presid"
## [7,] "today" "new" "preserv" "power" "act"
```

```
## [8,] "hand"      "restor"      "noth"        "question"    "part"
## [9,] "begin"    "line"        "high"        "requir"     "duti"
## [10,] "new"      "suffer"      "land"        "yet"         "action"
## [11,] "old"      "trade"       "bless"       "mean"        "execut"
## [12,] "greater" "given"       "unit"        "practic"     "howev"
## [13,] "whose"   "product"     "constitut"   "consid"     "differ"
## [14,] "product" "past"        "oath"        "accomplish" "author"
## [15,] "women"   "true"        "wise"        "clear"       "local"
## [16,] "place"   "attent"      "respons"     "ever"        "bodi"
## [17,] "famili"  "throughout" "trust"       "settl"       "want"
## [18,] "lie"     "consider"    "faith"       "congress"    "given"
## [19,] "million" "evil"        "best"        "vast"        "case"
## [20,] "small"   "extens"      "race"        "road"        "immedi"
##      Topic 11 Topic 12 Topic 13 Topic 14 Topic 15
## [1,] "spirit"  "time"       "power"       "let"         "public"
## [2,] "peopl"   "present"    "govern"      "hope"        "countri"
## [3,] "liberti" "revenu"     "constitut"   "nation"      "polit"
## [4,] "time"    "system"     "grant"       "know"        "import"
## [5,] "free"    "upon"       "execut"      "prosper"     "parti"
## [6,] "charact" "prosper"    "effect"      "face"        "offic"
## [7,] "continu" "secur"      "instrument"   "seek"        "purpos"
## [8,] "republ"  "currenc"   "patriot"     "freedom"     "best"
## [9,] "upon"    "money"     "depart"      "futur"       "servic"
## [10,] "produc" "treasuri"   "general"     "children"    "law"
## [11,] "long"   "credit"     "great"       "endur"       "princip"
## [12,] "love"   "judgment"   "feder"       "common"      "administr"
## [13,] "result" "maintain"   "control"     "father"      "give"
## [14,] "whole"  "prompt"     "appear"      "courag"     "support"
## [15,] "thing"  "howev"      "becom"       "world"      "regard"
## [16,] "devot"  "increas"    "exercis"     "generat"     "necess"
## [17,] "feel"   "valu"       "danger"      "today"      "elect"
## [18,] "secur"  "constant"   "reserv"      "word"       "presid"
## [19,] "remain" "effort"     "influenc"    "achiev"     "place"
## [20,] "passion" "failur"    "proper"      "depend"     "partisan"
```

```
freq <- sort(colSums(as.matrix(dtm)))
```

```
wf <- data.frame(word=names(freq),freq=freq)
```

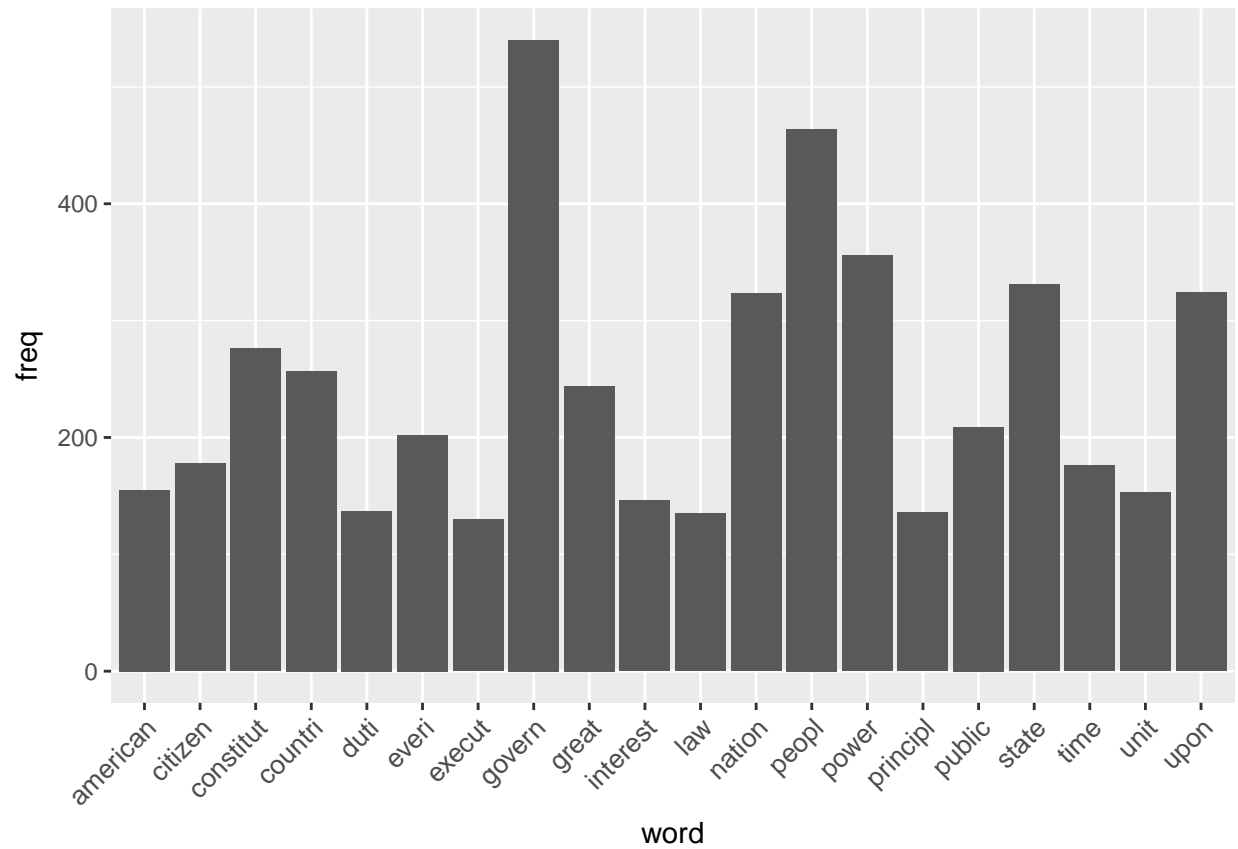
```
freq2 <- colSums(as.matrix(dtm))
```

```
p_e <- ggplot(subset(tail(wf,20),freq>50),aes(word,freq))
```

```
p_e <- p_e+geom_bar(stat = "identity")
```

```
p_e <- p_e+theme(axis.text.x = element_text(angle = 45,hjust = 1,size = 10))
```

```
p_e
```



The most common words that extroverted president used during an economic recession.