General analysis of inaugurations

Zishuo Li zl2528

January 23, 2017

Step1:

check and install needed packages. Load the libraries.

In the first part, require all package used below.

Step2:

Data harvest and process

In this part, I read all inauguration, process data and remove all unnecessary words including setting stop words, punctuation and numbers.

```
folder.path="../data/InauguralSpeeches/"
speech.list=list.files(path=folder.path,pattern = "*.txt")
prez.out=substr(speech.list, 6, nchar(speech.list)-4)


ff.all <- Corpus(DirSource(folder.path))
ff.all <- tm_map(ff.all,stripWhitespace)
ff.all <- tm_map(ff.all,content_transformer(tolower))
#set up stop words
myStopwords <-
c("can","say","one","way","use","also","howev","tell","will","much","need","take","tend","even","l
ke","particular","rather","said","get","well","make","ask","come","end","first","two","help","of
ten","may","might","see","someth","thing","point","post","look","right","now","think","'ve","'r
e","anoth","put","set","new","good","want","sure","kind","larg","yes","day","quit","sinc","attem
pt","lack","seen","awar","littl","ever","moreov","though","found","abl","enough","far","earli","aw
y","achiev","last","never","brief","bit","entir","lot","must","shall")
ff.all <- tm_map(ff.all,removeWords,stopwords("english"))
ff.all <- tm_map(ff.all,removeWords,myStopwords)
ff.all <- tm_map(ff.all,stemDocument)
ff.all <- tm_map(ff.all,removeWords,character(0))###
ff.all <- tm_map(ff.all,removePunctuation)
ff.all <- tm_map(ff.all,removeNumbers)
ff.all <- tm_map(ff.all,PlainTextDocument)
```

Constructing dates matrix

Since I would like to see the change of inauguration with time passing by, this chunk I mainly deal with dates data of inauguration and make it a tidy data frame, so that It could be merge with the word matrix and sentence matrix.

```
dates<- read.table("../data/InauguationDates.txt",header = T,sep="\t")
for( i in 1:ncol(dates)){dates[,i]<- dates[,i] %>% as.character()}
choose_year <- function(x){substr(x,nchar(x)-3,nchar(x))} #set a function to choose year rather
 than day/month/year
year_data<- sapply(dates[,2:5],choose_year)


# select white space and . and replace with"" to have the same format with prez.out
test<- gregexpr("\\s|[[:punct:]]",dates$PRESIDENT)
regmatches(dates$PRESIDENT,test) <- ""
dates <- cbind(dates[,1],year_data) %>%data.frame()
for(i in 2:5){dates[,i] <- as.numeric(as.character(dates[,i]))}
dates[,1] <- dates[,1]%>% as.character()
dates[dates[,1]=="GroverCleveland",1]<-c("GroverCleveland-I","GroverCleveland-II")# Since presid
ent GroverCleveland is special, I set it manually here.

# set a function to select president, corresponding year, and term.
dates_function <- function(x){
  term_value<- sum(!is.na(x[,2:5]))
  answer <- NULL
  for(i in 1:term_value){
    answer1 <- c(x[,1],term=i,x[,1+i])
    answer <- rbind(answer,answer1)
  }
  return(answer)
}
# use the function above to selecet data
answer <- NULL
for(i in 1:dim(dates)[1]){
  answer <- rbind(answer,dates_function(dates[i,]))
}
colnames(answer) <- c("File","term","year")# assign colnames to the data frame.
answer <- data.frame(answer)

for(i in 2:3){
  answer[,i] <- as.numeric(as.character(answer[,i]))}
index<- which(is.na(answer[,3])) #since president GroverCleverland is special, I deal with it ma
nually.
answer[index,3] <- dates[dates[,1]=="GroverCleveland-II",3]
answer[index,2] <- 2
colnames(answer) <- c("File",colnames(answer)[2:3])
dates_prez <- answer
head(dates_prez)
```

```
##               File term year
## 1 GeorgeWashington    1 1789
## 2 GeorgeWashington    2 1793
## 3        JohnAdams    1 1797
## 4  ThomasJefferson    1 1801
## 5  ThomasJefferson    2 1805
## 6     JamesMadison    1 1809
```

Step3: Term analysis/Topic analysis

topic modeling

In this chunk, I generate document term matrix and use LDA method to do topic modeling. Illustrate 15 topics and topic terms.In this case, k=15, means 15 topics.

```
dtm <- DocumentTermMatrix(ff.all)

freq <- colSums(as.matrix(dtm))
ord <- order(freq,decreasing = T)
row.names(dtm) <- speech.list
rowTotals <- apply(dtm,1,sum)


burnin <- 4000
iter <- 2000
thin <- 500
seed <- 2003
nstart <- 5
best <- TRUE

k <- 15

ldaout <- LDA(dtm,k,method = "Gibbs",control=list(seed=seed,best=best,burnin=burnin,iter=iter,th
in=thin))

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
##  1  1  1 31  2  1 11  2  1  1 16  1  1  1  2
```

```
topicProbabilities <- as.data.frame(ldaout@gamma)

terms.beta=ldaout@beta #select term out
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,] "dreams"    "counter"      "beginning" "speaker"      "front"
##  [2,] "republics" "preserved"    "emin"      "gratitud"     "confeder"
##  [3,] "abandon"   "departur"     "protector" "seaboard"     "launch"
##  [4,] "general"   "states"       "regard"    "result"       "exercis"
##  [5,] "oppos"     "amend"        "prohibit"  "negro"        "fix"
##  [6,] "science"   "stood"        "tide"      "facilit"      "heedless"
##  [7,] "standard"  "econom"       "poor"      "growth"       "rememb"
##  [8,] "republic"  "civilization" "gratitude" "suffering"    "aggrandiz"
##  [9,] "design"    "happy"        "appar"     "degrad"       "represent"
## [10,] "despot"    "contrary"     "although"  "commonwealth" "reckless"
## [11,] "together"  "fellow"       "learn"     "americans"    "dream"
## [12,] "minist"    "intellectu"   "durat"     "plenty"       "savag"
## [13,] "slaveri"   "thirteen"     "claus"     "discret"      "territories"
## [14,] "neutral"   "combat"       "repeal"    "correspond"   "repel"
## [15,] "stabil"    "rigid"        "insist"    "expans"       "humanity"
##       [,6]            [,7]
##  [1,] "inflat"        "kill"
##  [2,] "alik"          "indulg"
##  [3,] "removed"       "air"
##  [4,] "territori"     "adopt"
##  [5,] "financi"       "canal"
##  [6,] "decenc"        "eyes"
##  [7,] "belief"        "expand"
##  [8,] "amid"          "amiti"
##  [9,] "sinist"        "tragic"
## [10,] "district"      "departments"
## [11,] "quest"         "lives"
## [12,] "uncertain"     "vigil"
## [13,] "circumstances" "pretext"
## [14,] "emphat"        "mississippi"
## [15,] "transport"     "develop"
```

```
ldaOut.terms <- as.matrix(terms(ldaout,20))
ldaOut.terms
```

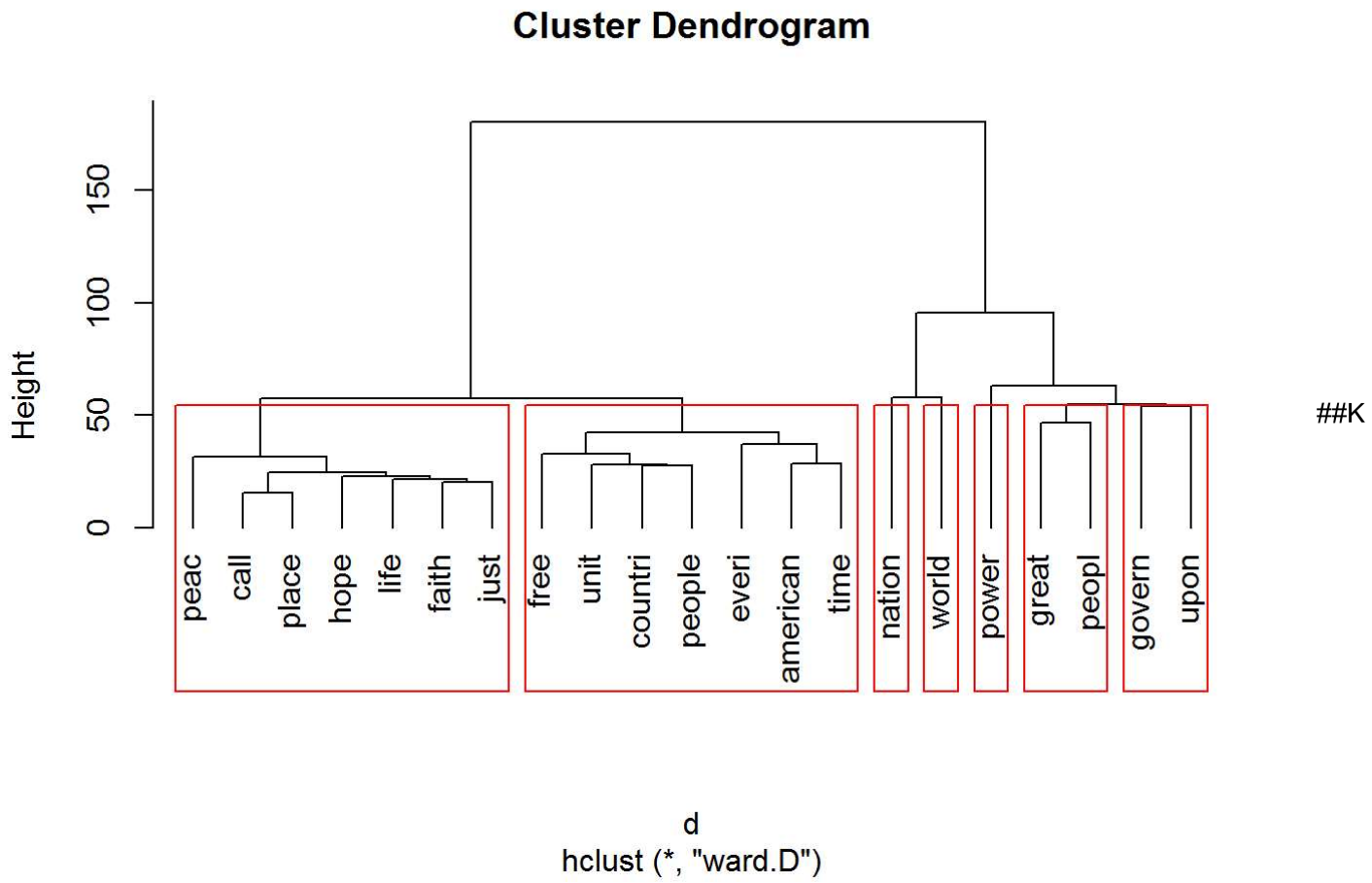
```
##       Topic 1     Topic 2         Topic 3     Topic 4      Topic 5
##  [1,] "hand"      "union"         "knowledg"  "govern"     "congress"
##  [2,] "among"     "happi"         "demand"    "state"      "law"
##  [3,] "feder"     "equal"         "bodi"      "upon"       "busi"
##  [4,] "left"      "extend"        "abandon"   "great"      "legisl"
##  [5,] "senat"     "opinion"       "elect"     "nation"     "race"
##  [6,] "tax"       "bless"         "abl"       "power"      "protect"
##  [7,] "dreams"    "fellowcitizen" "courag"    "peopl"      "upon"
##  [8,] "confront"  "foreign"       "solemn"    "public"     "south"
##  [9,] "creat"     "nations"       "valu"      "everi"      "tariff"
## [10,] "fact"      "mind"          "better"    "duti"       "control"
## [11,] "look"      "period"        "brought"   "constitut"  "trade"
## [12,] "weapon"    "harmoni"       "friend"    "unit"       "taken"
## [13,] "beginning" "debt"          "hous"      "countri"    "hope"
## [14,] "price"     "honor"         "stop"      "interest"   "prevent"
## [15,] "speaker"   "common"        "comfort"   "government" "secur"
## [16,] "thousand"  "whole"         "enterpris" "people"     "amend"
## [17,] "carri"     "object"        "european"  "execut"     "industri"
## [18,] "deficit"   "error"         "public"    "right"      "chang"
## [19,] "fight"     "just"          "wealth"    "principl"   "oppos"
## [20,] "strongest" "rights"        "administr" "states"     "avoid"
##       Topic 6   Topic 7   Topic 8        Topic 9      Topic 10
##  [1,] "excit"   "nation"  "republic"     "interest"   "power"
##  [2,] "solv"    "world"   "order"        "energi"     "charact"
##  [3,] "thing"   "peopl"   "call"         "sacr"       "control"
##  [4,] "stood"   "life"    "civilization" "evid"       "member"
##  [5,] "wast"    "peac"    "democracy"    "light"      "instrument"
##  [6,] "cost"    "hope"    "reflect"      "agit"       "grant"
##  [7,] "decis"   "free"    "add"          "design"     "spirit"
##  [8,] "rest"    "men"     "sacrific"     "voic"       "appear"
##  [9,] "alter"   "freedom" "today"        "trust"      "republ"
## [10,] "firm"    "human"   "particip"     "confidence" "indeed"
## [11,] "follow"  "seek"    "road"         "conflict"   "passion"
## [12,] "mani"    "peace"   "trade"        "nation"     "suppos"
## [13,] "process" "believ"  "relationship" "american"   "bodi"
## [14,] "proud"   "great"   "service"      "case"       "want"
## [15,] "repos"   "stand"   "caus"         "mark"       "confederaci"
## [16,] "struggl" "man"     "compet"       "pray"       "exclus"
## [17,] "tide"    "know"    "express"      "affair"     "intend"
## [18,] "allow"   "common"  "eye"          "gave"       "legisl"
## [19,] "bad"     "war"     "hate"         "judg"       "origin"
## [20,] "deal"    "live"    "precious"     "providence" "possess"
##       Topic 11   Topic 12    Topic 13    Topic 14    Topic 15
##  [1,] "america"  "long"      "law"       "war"       "law"
##  [2,] "let"      "found"     "union"     "forc"      "american"
##  [3,] "time"     "requir"    "constitut" "equal"     "respons"
##  [4,] "american" "fals"      "case"      "meet"      "advanc"
##  [5,] "nation"   "littl"     "parti"     "commerc"   "progress"
##  [6,] "everi"    "certain"   "perfect"   "afford"    "servic"
##  [7,] "world"    "sacrific"  "except"    "improv"    "method"
##  [8,] "today"    "defin"     "slaveri"   "soon"      "find"
##  [9,] "freedom"  "engag"     "almighti"  "proof"     "citizenship"
## [10,] "govern"   "oath"      "provis"    "resourc"   "follow"
```

```
## [11,] "great"    "alon"      "suprem"    "whose"     "govern"
## [12,] "work"     "suffer"    "surrend"   "commenc"   "accept"
## [13,] "year"     "blood"     "entitl"    "naval"     "develop"
## [14,] "god"      "disciplin" "north"     "enabl"     "perman"
## [15,] "know"     "hour"      "destroy"   "remain"    "promot"
## [16,] "fellow"   "seiz"      "either"    "chang"     "concern"
## [17,] "just"     "vigil"     "oath"      "late"      "enforc"
## [18,] "peopl"    "welcom"    "univers"   "manner"    "ideal"
## [19,] "presid"   "anoth"     "littl"     "neutral"   "people"
## [20,] "children" "brave"     "major"     "necessari" "justic"
```

Hierarchal Term Clustering

In this part, I continue to process data and cluster the most used terms into 7 group

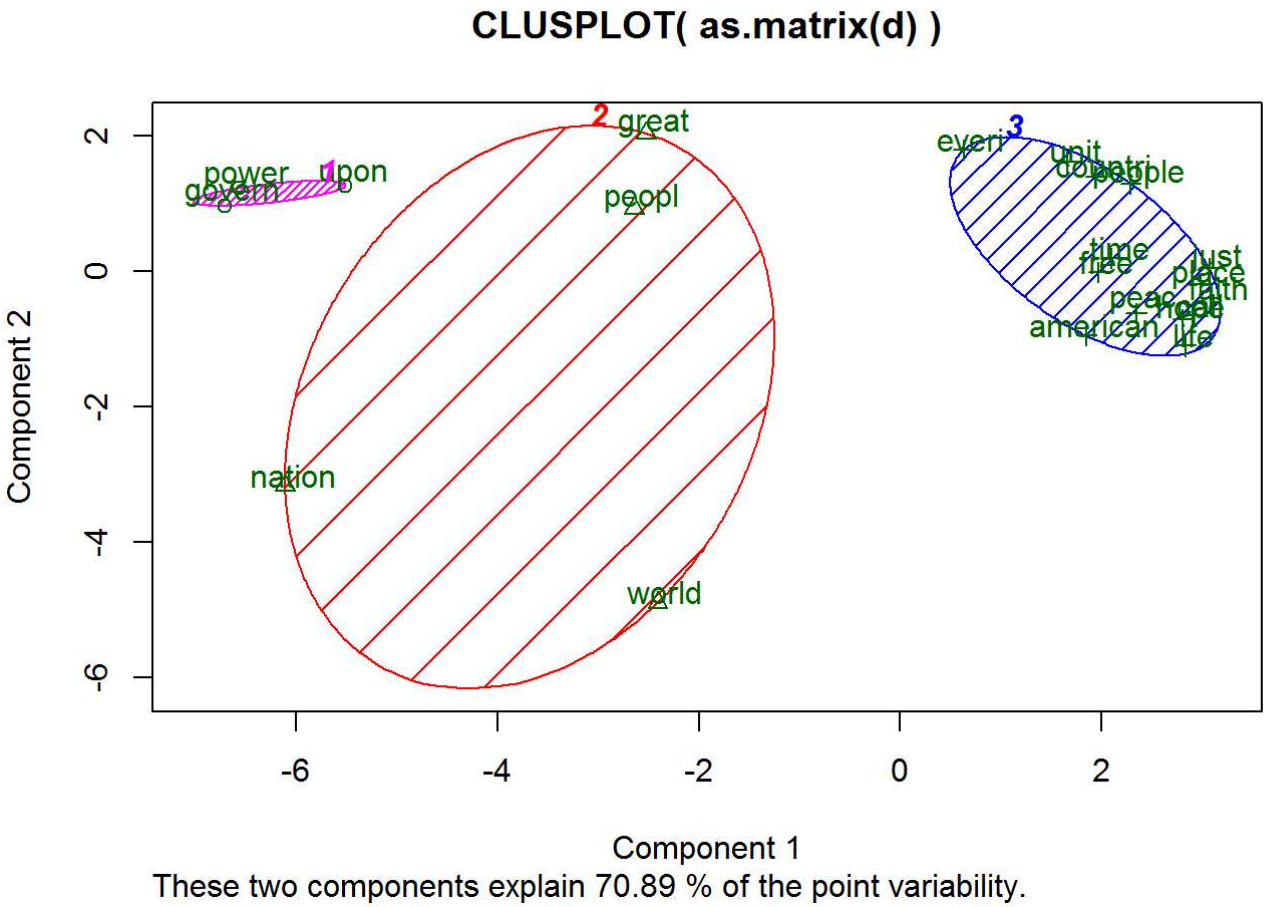
```
dtmss <- removeSparseTerms(dtm, 0.2)
d <- dist(t(dtmss), method="euclidian")
fit <- hclust(d=d, method="ward.D")
plot.new()
plot(fit, hang=-1)
groups <- cutree(fit, k=7)  # "k=" defines the number of clusters you are using
rect.hclust(fit, k=7, border="red")
```



means cluster

In this part, I use k means clusters to cluster the terms into 3 groups and illustrate it clearly.

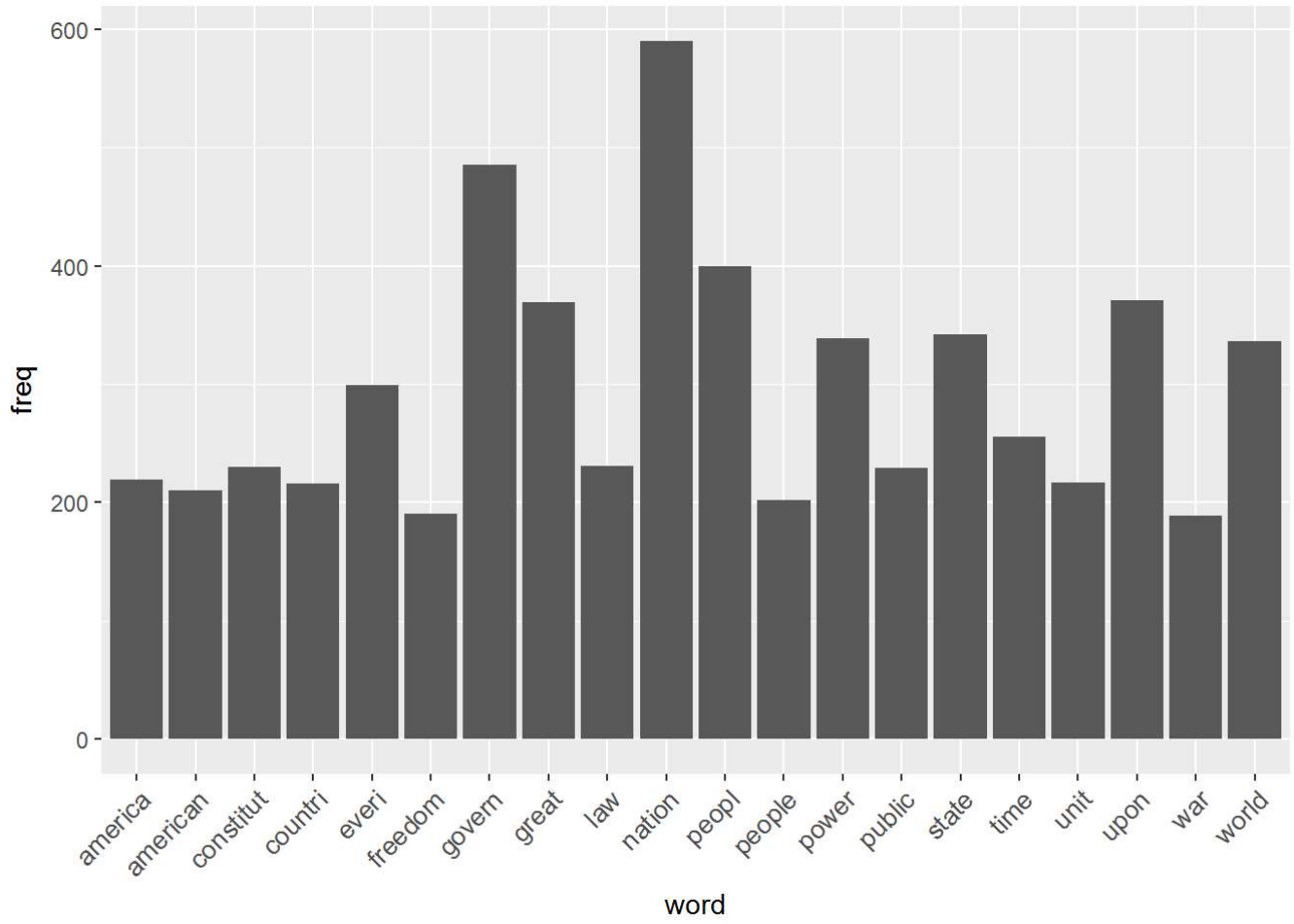
```
d <- dist(t(dtmss), method="euclidian")
kfit <- kmeans(d, 3)
clusplot(as.matrix(d), kfit$cluster, color=T, shade=T, labels=2, lines=0)
```



##term frequence

In this part, I would like to show the frequence of the most used term in all 58 inauguration. Since I have already set stop words above, terms such like “must”, “shall” are not listed, this could show the more meaningful terms.

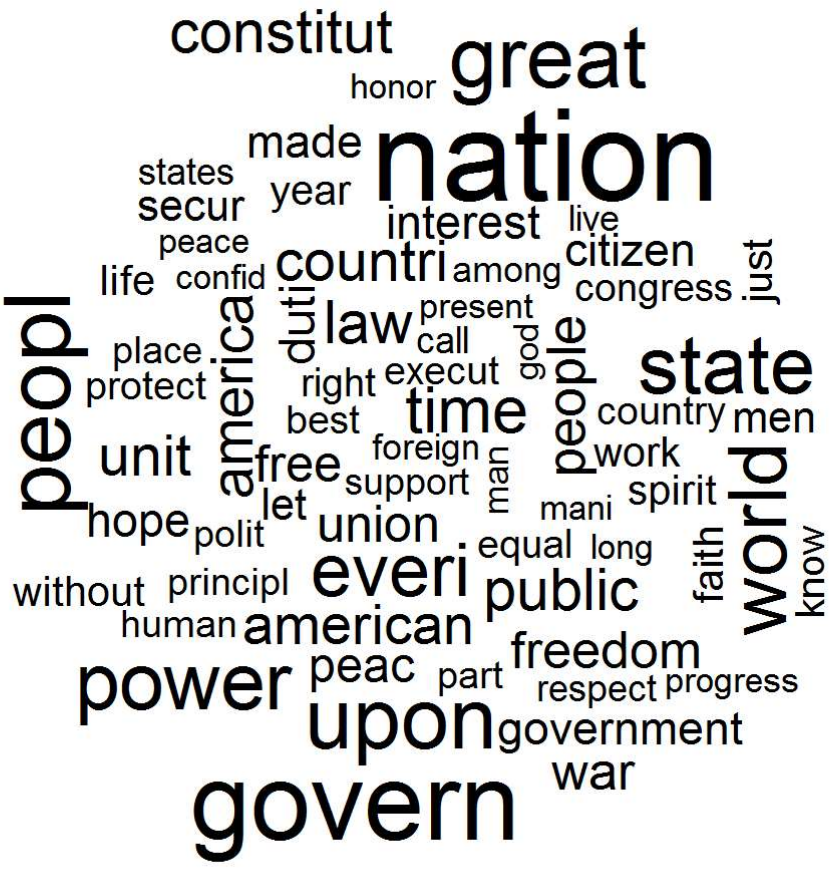
```
freq <- sort(colSums(as.matrix(dtm)))

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

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

p <- ggplot(subset(tail(wf,20),freq>50),aes(word,freq))
p <- p+geom_bar(stat = "identity")
p <- p+theme(axis.text.x = element_text(angle = 45,hjust = 1,size = 10))
p
```



##word cloud

In this part, I mainly use two method to show the word cloud of the most frequently used terms.

```
wordcloud(names(freq),freq,min.freq = 100)
```




```
wordcloud(names(freq),freq,min.freq = 100,scale = c(5,.1),colors=brewer.pal(6,"Dark2"))
```

Step4: sentence analysis

In this part, I load all the sentences and construct emotions matrix with function to prepare for sentence analysis

```
speech.sentance<- lapply(speech.list,
function(x)readLines(paste("../data/InauguralSpeeches/",x,sep = "")))

sentence.list=NULL
for(i in 1:length(speech.sentance)){
  sentences=sent_detect(speech.sentance[[i]],
                        endmarks = c("?", ".", "!", "|",";"))
  if(length(sentences)>0){
    emotions=get_nrc_sentiment(sentences)
    word.count=word_count(sentences)
    #Count the word in each sentences
    emotions=diag(1/(word.count+0.01))%*%as.matrix(emotions)
    term =as.numeric(substr(prez.out[i],nchar(prez.out[i]),nchar(prez.out[i])))
    File=as.character(substr(prez.out,1,nchar(prez.out)-2)[i])
    sentence.list=rbind(sentence.list,
            cbind(sentences=as.character(sentences),
                  File,
                  term,
                  word.count,
                  emotions,
                  sent.id=1:length(sentences)
                  )
    )
  }
}

sentence.list<- data.frame(sentence.list)
sentence.list=sentence.list%>%filter(!is.na(word.count))
sentence.list <- data.frame(sentence.list)
name_inaug<- colnames(sentence.list)

sentence.list_part1 <- sentence.list[,1:2]
sentence.list_part2 <- NULL
for (k in 3:15){
  sentence.list_part2 <- cbind(sentence.list_part2,as.numeric(as.character(sentence.list[,k])))
}
sentence.list <-cbind(sentence.list_part1,sentence.list_part2)
colnames(sentence.list) <- name_inaug
```

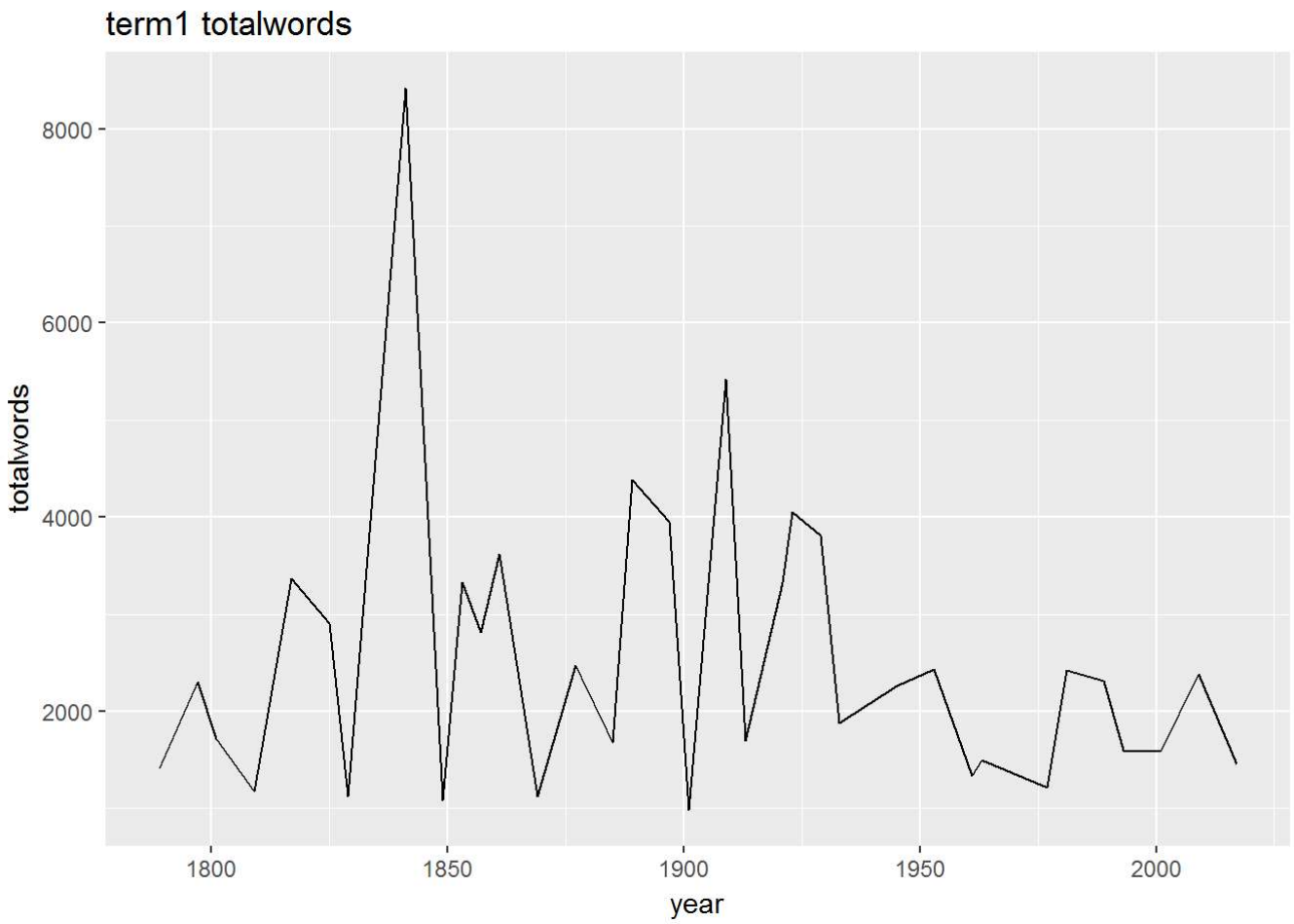
In this chunk, I merge the sentence matrix with the date matrix constructed above through names of president and terms of each president.

```
sentence.list<- merge(sentence.list,dates_prez,by.x = c("File","term"),by.y = c("File","term"))
colnames(sentence.list)<-c(colnames(sentence.list)[1:length(colnames(sentence.list))-1],"year")
head(sentence.list,2)
```

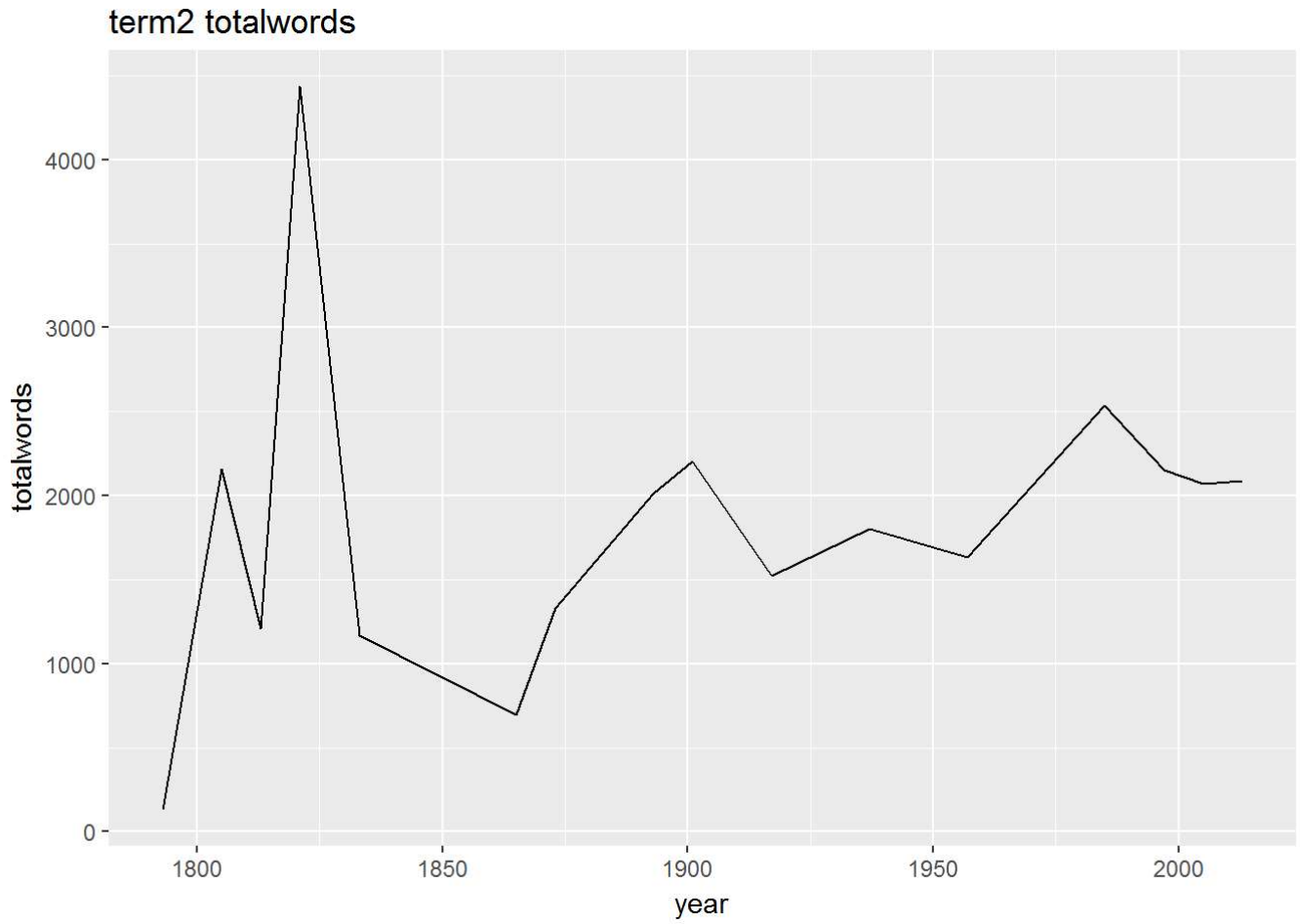
```
##             File term
## 1 AbrahamLincoln    1
## 2 AbrahamLincoln    1
##

    sentences
## 1 While I make no recommendation of amendments, I fully recognize the rightful authority of t
he people over the whole subject, to be exercised in either of the modes prescribed in the instr
ument itself;
## 2                                                                                and I should, under
 existing circumstances, favor rather than oppose a fair opportunity being afforded the people t
o act upon it.
##   word.count anger anticipation disgust fear joy sadness surprise
## 1         33     0   0.00000000       0    0   0       0        0
## 2         21     0   0.04759638       0    0   0       0        0
##       trust   negative   positive sent.id year
## 1 0.0605877 0.03029385 0.09088155     118 1861
## 2 0.0000000 0.04759638 0.09519277     119 1861
```

In this chunk, I plot the total length of sentences of every inauguration to see if there is any trend existing.

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

sentence.list_term1 <- filter(sentence.list,term==1)

sentence.list_term2 <- filter(sentence.list,term==2)

term1_data<- group_by(sentence.list_term1,year)
timeline1<- summarise(term1_data,
          totalwords=sum(word.count),
          totalsentenca=length(sent.id)
          )

ggplot(data =timeline1 )+
  geom_line(mapping = aes(x=year,y=totalwords))+
  labs(title="term1 totalwords")
```

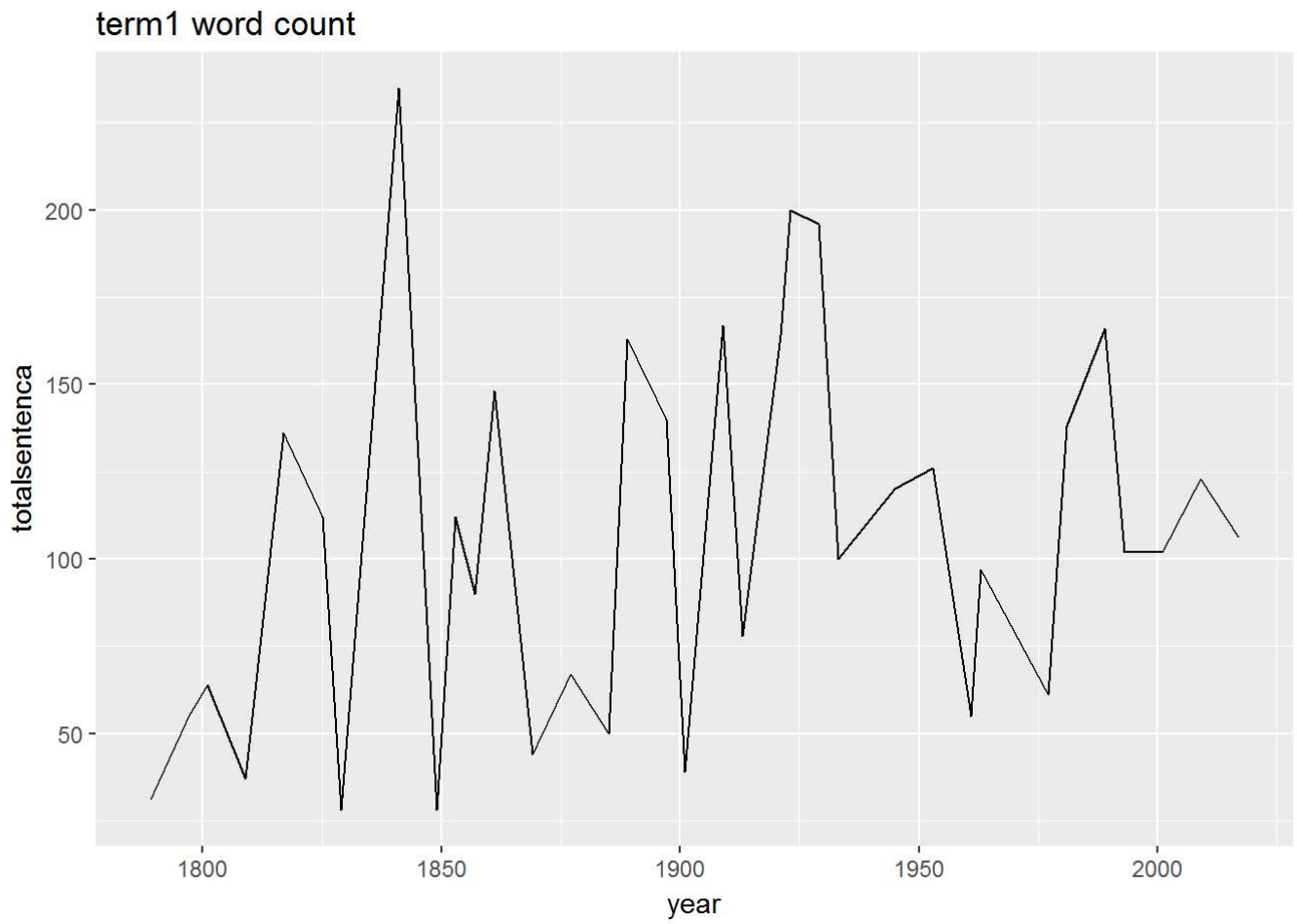



```
term2_data<- group_by(sentence.list_term2,year)
timeline2<- summarise(term2_data,
          totalwords=sum(word.count),
          totalsentenca=length(sent.id)
          )

ggplot(data =timeline2 )+
  geom_line(mapping = aes(x=year,y=totalwords))+
  labs(title="term2 totalwords")
```

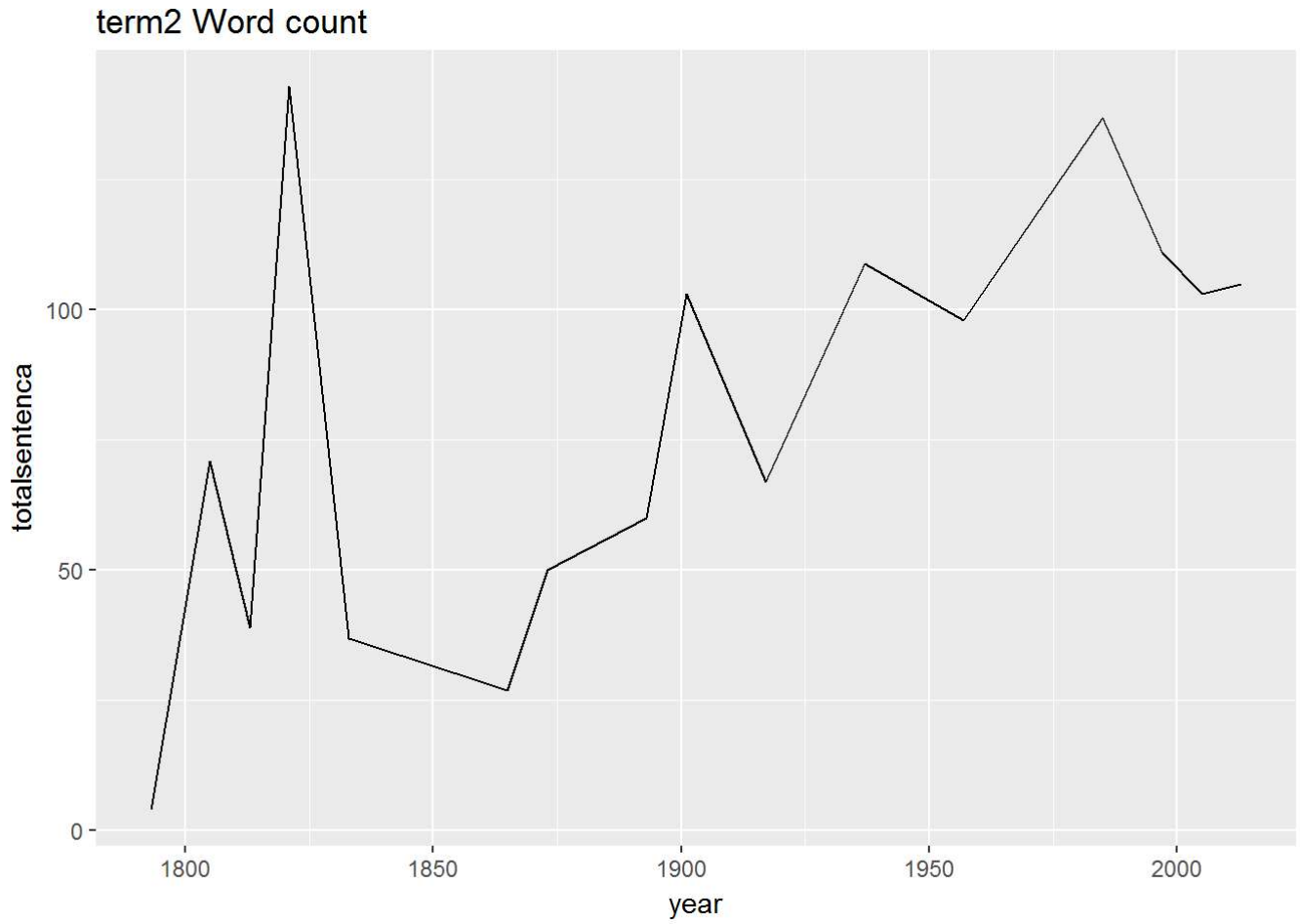


In this chunk, I plot the word count of every inauguration based on term.

```
ggplot(data = timeline1)+
  geom_line(mapping = aes(x=year,y=totalsentenca))+
  labs(title="term1 word count")
```




```
ggplot(data = timeline2)+
  geom_line(mapping = aes(x=year,y=totalsentenca))+
  labs(title="term2 Word count")
```



Since Donald Trump has been so popular recently, In this chunk, ICheck the shortest and longest sentences that donald trump said.

```
sentence_Donald<- filter(sentence.list,File=="DonaldJTrump")
arrange(sentence_Donald,desc(word.count))[1:3,"sentences"]
```

```
## [1] And whether a child is born in the urban sprawl of Detroit or the wind-swept plains of Ne
braska, they look up at the same night sky, they will their heart with the same dreams, and they
 are infused with the breath of life by the same almighty creator.
## [2] Every four years, we gather on these steps to carry out the orderly and peaceful transfer
 of power, and we are grateful to President Obama and First Lady Michelle Obama for their gracio
us aid throughout this transition.
## [3] We stand at the birth of a new millennium, ready to unlock the mysteries of space, to fre
e the earth from the miseries of disease, and to harness the energies, industries and technologi
es of tomorrow.
## 5643 Levels: 'May' Congress prohibit slavery in the Territories? ...
```

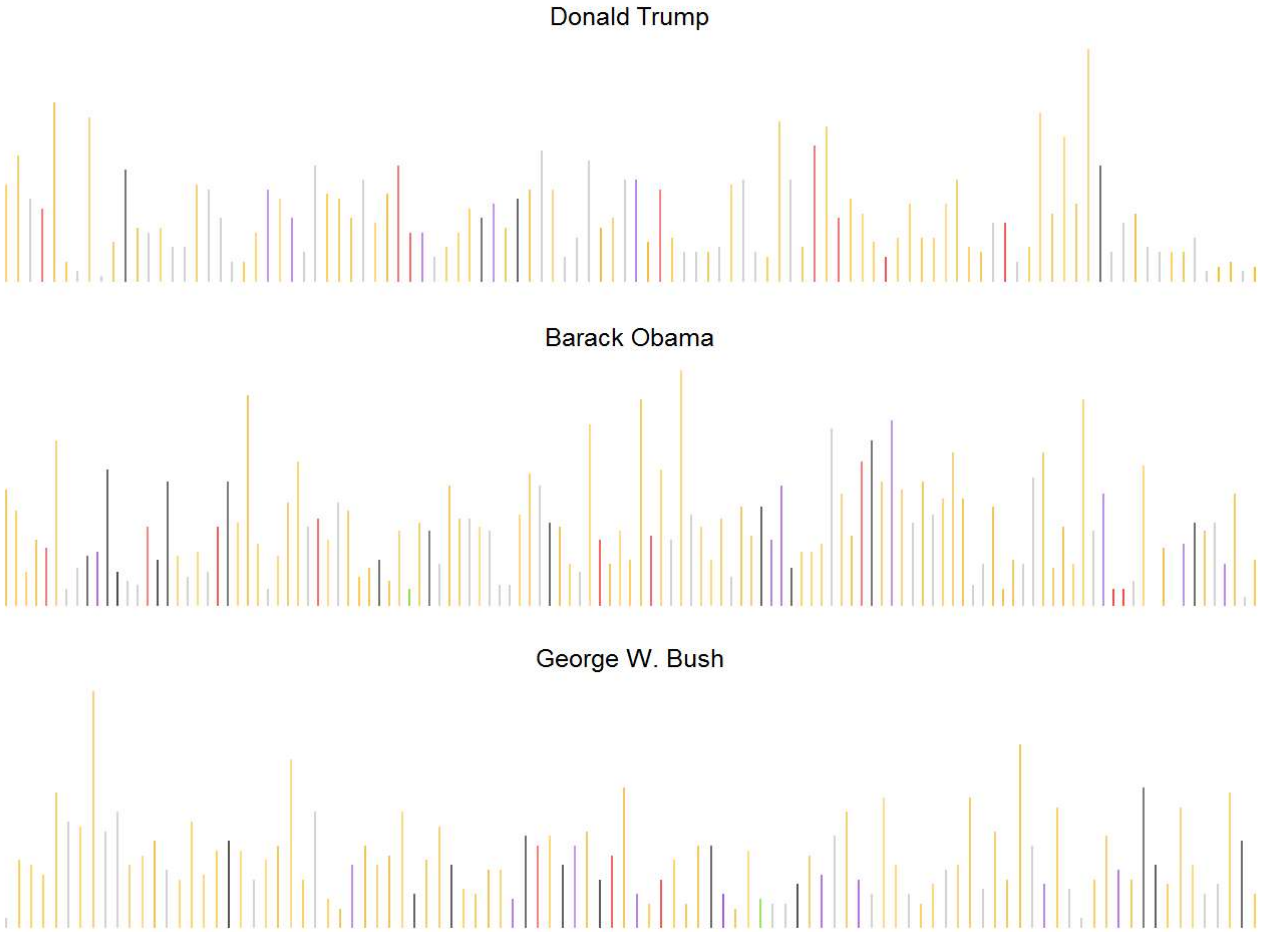
```
arrange(sentence_Donald,word.count)[1:10,"sentences"]
```

```
##  [1] C.                          Thank you.
##  [3] Thank you.                  Thank you.
##  [5] God bless America.          God bless you.
##  [7] They have been magnificent. This is your day.
##  [9] This is your celebration.   We will not fail.
## 5643 Levels: 'May' Congress prohibit slavery in the Territories? ...
```

Relationship between Sentence length and emotions

In this chunk, f.plotsent.len is a function defined above and it mainly choose the most strong emotion in each sentence and assign specific color to that sentence.

```
par(mfrow=c(3,1), mar=c(1,0,2,0), bty="n", xaxt="n", yaxt="n", font.main=1)

f.plotsent.len(In.list=sentence.list, InFile="DonaldJTrump",InTerm=1, President="Donald Trump")

f.plotsent.len(In.list=sentence.list, InFile="BarackObama",  InTerm=1, President="Barack Obama")

f.plotsent.len(In.list=sentence.list, InFile="GeorgeWBush", InTerm=1, President="George W.
Bush")
```



The most emotionally charged sentences of Barack Obama and Donald Trump

```
print("Barack Obama")
```

```
## [1] "Barack Obama"
```

```
speech.df=tbl_df(sentence.list)%>%
  filter(File=="BarackObama", term==1, word.count>=5)%>%
  select(sentences, anger:trust)
speech.df=as.data.frame(speech.df)
as.character(speech.df$sentences[apply(speech.df[,-1], 2, which.max)])
```

```
## [1] "On this day, we gather because we have chosen hope over fear, unity of purpose over conf
lict and discord."
## [2] "This is the journey we continue today."

## [3] "We remain the most prosperous, powerful nation on Earth."

## [4] "Our Nation is at war against a far-reaching network of violence and hatred."

## [5] "We remain the most prosperous, powerful nation on Earth."

## [6] "Our Nation is at war against a far-reaching network of violence and hatred."

## [7] "This is the meaning of our liberty and our creed;"

## [8] "God bless you, and God bless the United States of America."
```

```
print("Donald Trump")
```

```
## [1] "Donald Trump"
```

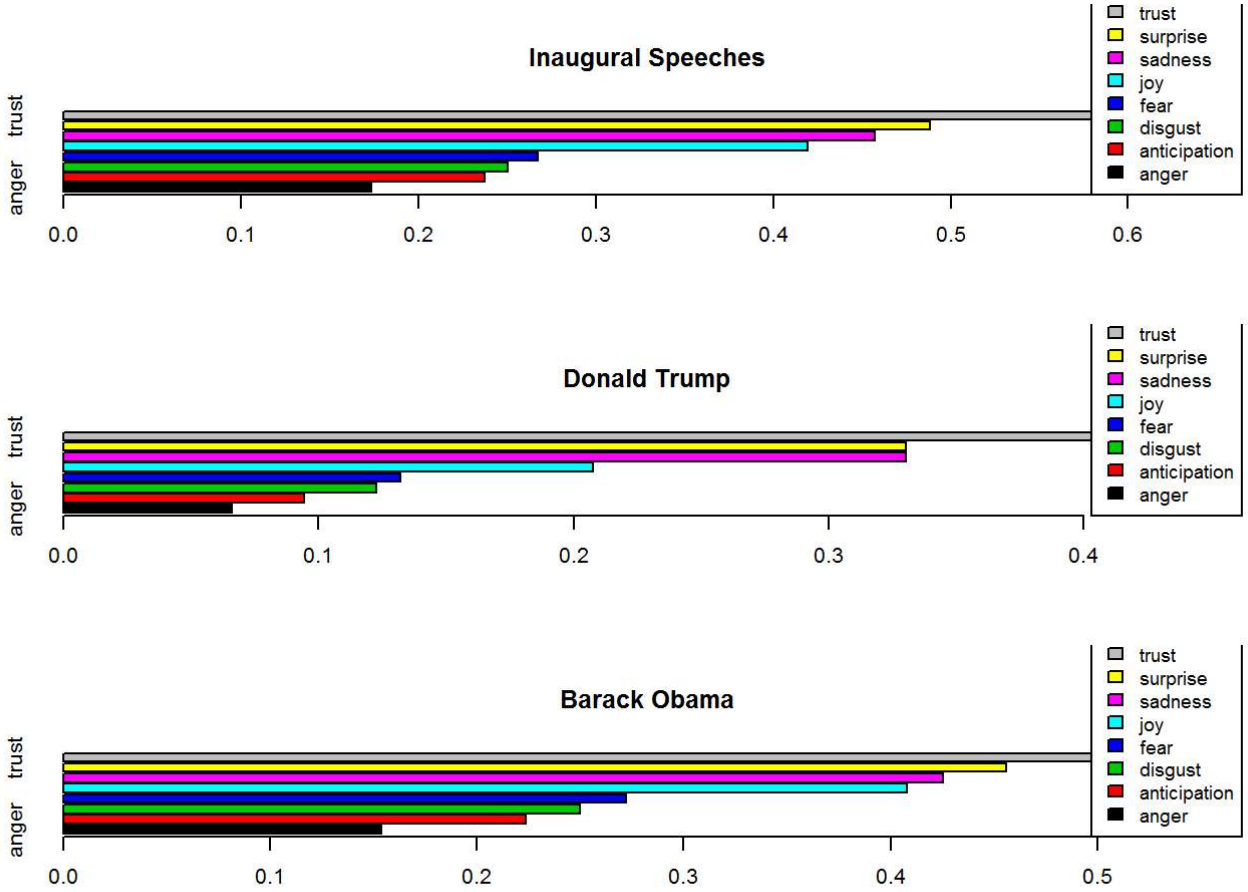
```
speech.df=tbl_df(sentence.list)%>%
  filter(File=="DonaldJTrump", term==1,word.count>=5)%>%
  select(sentences, anger:trust)
speech.df=as.data.frame(speech.df)
as.character(speech.df$sentences[apply(speech.df[,-1], 2, which.max)])
```

```
## [1] "There should be no fear."

## [2] "America will start winning again, winning like never before."

## [3] "America will start winning again, winning like never before."

## [4] "There should be no fear."

## [5] "buy American and hire American."

## [6] "America will start winning again, winning like never before."

## [7] "The bible tells us how good and pleasant it is when God's people live together in unit
y."
## [8] "At the center of this movement is a crucial conviction, that a nation exists to serve it
s citizens."
```

comparation of emotions

In this chunk, I compare the emotions between all inaugural speeches, speeche of Donald Trump and speech of Obama

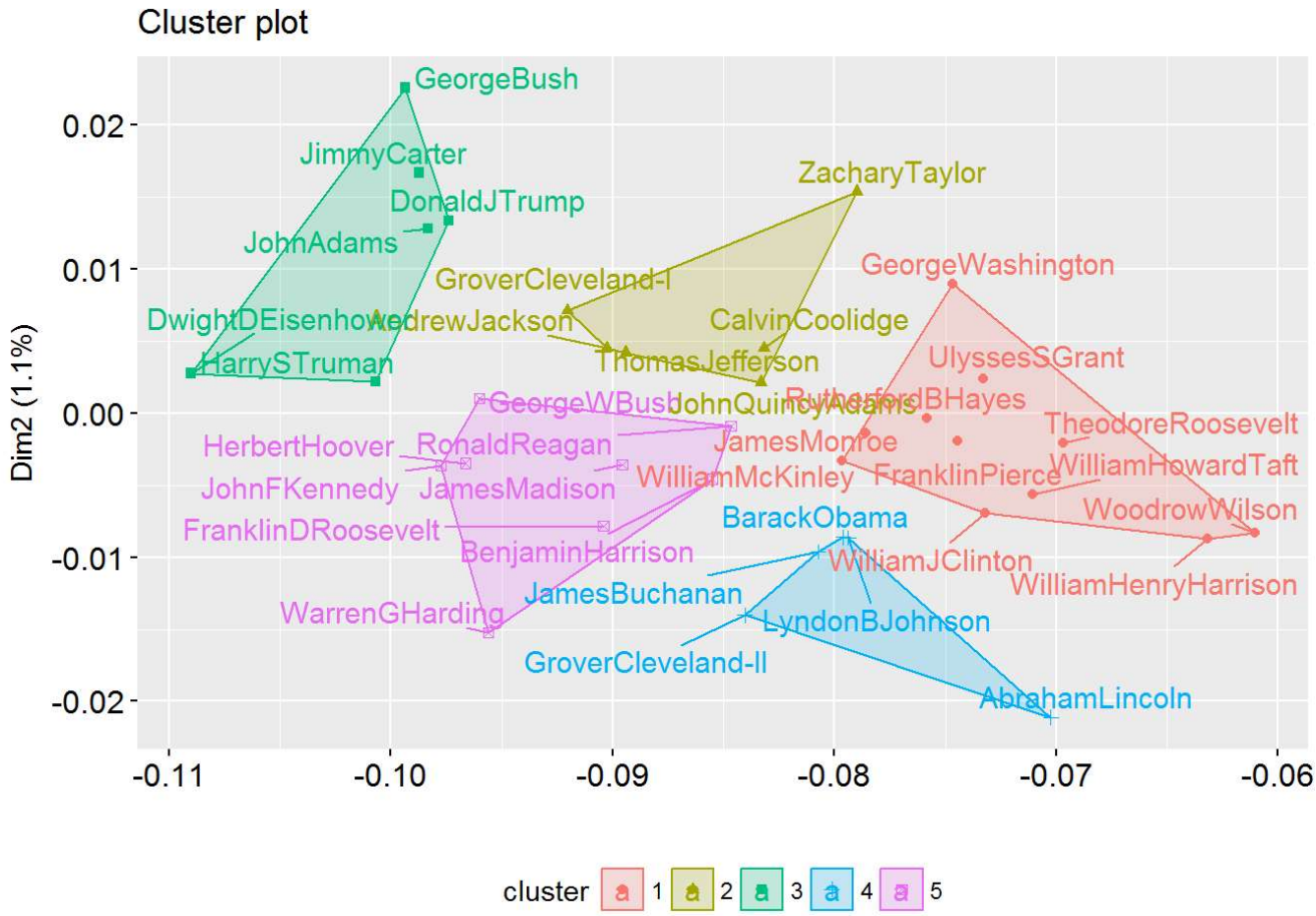
```
#The distribution of emotions in all sentences
par(mfrow=c(3,1))
emo.means=colMeans(select(sentence.list, anger:trust)>0.01)
barplot(height = emo.means[order(emo.means)],horiz = TRUE,col=1:8,main="Inaugural Speeches",name
s=names(emo.means),legend = names(emo.means),args.legend = list(x="bottomright",cex=0.9))

emo.means=colMeans(select(sentence_Donald, anger:trust)>0.01)
barplot(height = emo.means[order(emo.means)],horiz = TRUE,col=1:8,main="Donald Trump",names=name
s(emo.means),legend = names(emo.means),args.legend = list(x="bottomright",cex=0.9))

sentence_Obama<- filter(sentence.list,File=="BarackObama")
emo.means=colMeans(select(sentence_Obama, anger:trust)>0.01)
barplot(height = emo.means[order(emo.means)],horiz = TRUE,col=1:8,main="Barack Obama",names=name
s(emo.means),legend = names(emo.means),args.legend = list(x="bottomright",cex=0.9))
```



Emotion cluster

In this chunk, I cluster all presidents according to the emotions matrix of their inauguration using Kmeans method.

```
presid.summary=sentence.list%>%
  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))

presid.summary=as.data.frame(presid.summary)
rownames(presid.summary)=as.character((presid.summary[,1]))
km.res=kmeans(presid.summary[,-1],iter.max=200,5)
fviz_cluster(km.res,stand=F, repel= TRUE,data = presid.summary[,-1], xlab="", xaxt="n",show.clus
t.cent=FALSE)
```


Cluster plot
GeorgeBush
JimmyCarter
DonaldJTrump
JohnAdams
DwightDEisenhower
HarrySTruman
ZacharyTaylor
GroverCleveland-I
AndrewJackson
CalvinCoolidge
ThomasJefferson
GeorgeWashington
UlyssesSGrant
RutherfordBHayes
JohnQuincyAdams
TheodoreRoosevelt
JamesMonroe
WilliamHowardTaft
FranklinPierce
WoodrowWilson
WilliamJClinton
WilliamHenryHarrison
BarackObama
LyndonBJohnson
AbrahamLincoln
GeorgeWBush
HerbertHoover
RonaldReagan
JohnFKennedy
JamesMadison
FranklinDRoosevelt
WilliamMcKinley
BenjaminHarrison
JamesBuchanan
WarrenGHarding
GroverCleveland-II
Dim2 (1.1%)
0.02
0.01
0.00
-0.01
-0.02
-0.11
-0.10
-0.09
-0.08
-0.07
-0.06
cluster 1 2 3 4 5