Data Story on Presidents' speeches

Xin Luo

2017/9/7

This report is consist of 4 parts and the last part could be read as a appendix

1. Inaugaration Speeches and Nomination Speeches
   This part explores the simialrities and differences between inaugration speeches and nomination speeches thoroughly. Except for some general features, here we also discuss 2 example in details, which are both provided by Trump.
2. Similarities between 2 Successive Inaugrations
   By studying 2 succussive terms' inaugration speeches, we could conclude the issues really matter to presidents of these 2 terms, having a insight about these 8 years. Besides, for those 2-term presidents, we could be informed about their main contributions and focus, because they love to mention their contributions in the second term inaugration speeches.
3. Some Interesting Details
   Analysis on the general or personal speaking habits of the presidents
4. Topics speeches cover
   This part is to have a look about the topics presidents are attached to, but have nothing suprised. It's easy to guess the themes presidents would like to show their interests in the inaugration speeches.

(This report uses nomination, inaugration and farewell speeches of presidents and candidates)

Before starting the analysis, some data processing work have to be done.

1. Load the needed packages

```
setwd("~/Desktop/[ADS]Advanced Data Science/fall2017-project1-XinLuoCU")

library("rvest")
library("tibble")
library("qdap")
library("sentimentr")
library("gplots")
library("plyr")
library("dplyr")
library("tm")
library("syuzhet")
library("factoextra")
library("beeswarm")
library("scales")
library("RColorBrewer")
library("RANN")
library("tm")
library("topicmodels")
library("lda")
library('readr')
library('gdata')
library('wordcloud')
library('tidytext')
library("ggplot2")
```

```
library("scales")
library("gridExtra")

source("~/Desktop/[ADS]Advanced Data Science/fall2017-project1-XinLuoCU/lib/plotstacked.R")
source("~/Desktop/[ADS]Advanced Data Science/fall2017-project1-XinLuoCU/lib/speechFuncs.R")
```

2. Make the speeches files a dataframe

```
speech_list<-read_csv("../data/speech.list.csv")
speech_list$text<-NA
for(i in 1:nrow(speech_list)){
  text <- read_html(speech_list$urls[i]) %>% # load the page
    html_nodes(".displaytext") %>% # isloate the text
    html_text() # get the text
  speech_list$text[i]<-text
}
speech_list$Words<-c()
```

3. Generate the list of sentences

```
sentence_list<-NULL
for(i in 1:nrow(speech_list)){
  sentences<-sent_detect(speech_list$text[i],endmarks = c("?","!",".","|",";"))
  if(length(sentences)>0){
    emotion<-get_nrc_sentiment(sentences)
    word.count<-word_count(sentences)

    Word=sum(word.count,na.rm = T)
    emotion<-diag(1/(word.count+0.1))%*%as.matrix(emotion)
    sentence_list<-rbind(sentence_list,
                         cbind(speech_list[i,-ncol(speech_list)],
                               word.count,
                               Word=Word,
                               emotion,
                               sentences=as.character(sentences),
                               sent.id=1:length(sentences)
                               ))
  }

}

sentence_list=sentence_list[!is.na(sentence_list$word.count),]
sentence_list$sentences<-as.character(sentence_list$sentences)
sentence_list$Win<-tolower(sentence_list$Win)
#sentence_list$sentences<-tolower(sentence_list$sentences)
# sentence_list<-sentence_list%>%
#   filter(sentences%in%c("four more years!","audience","boo-o-o","the president."))
```

PART I: Inaugaration Speeches and Nomination Speeches

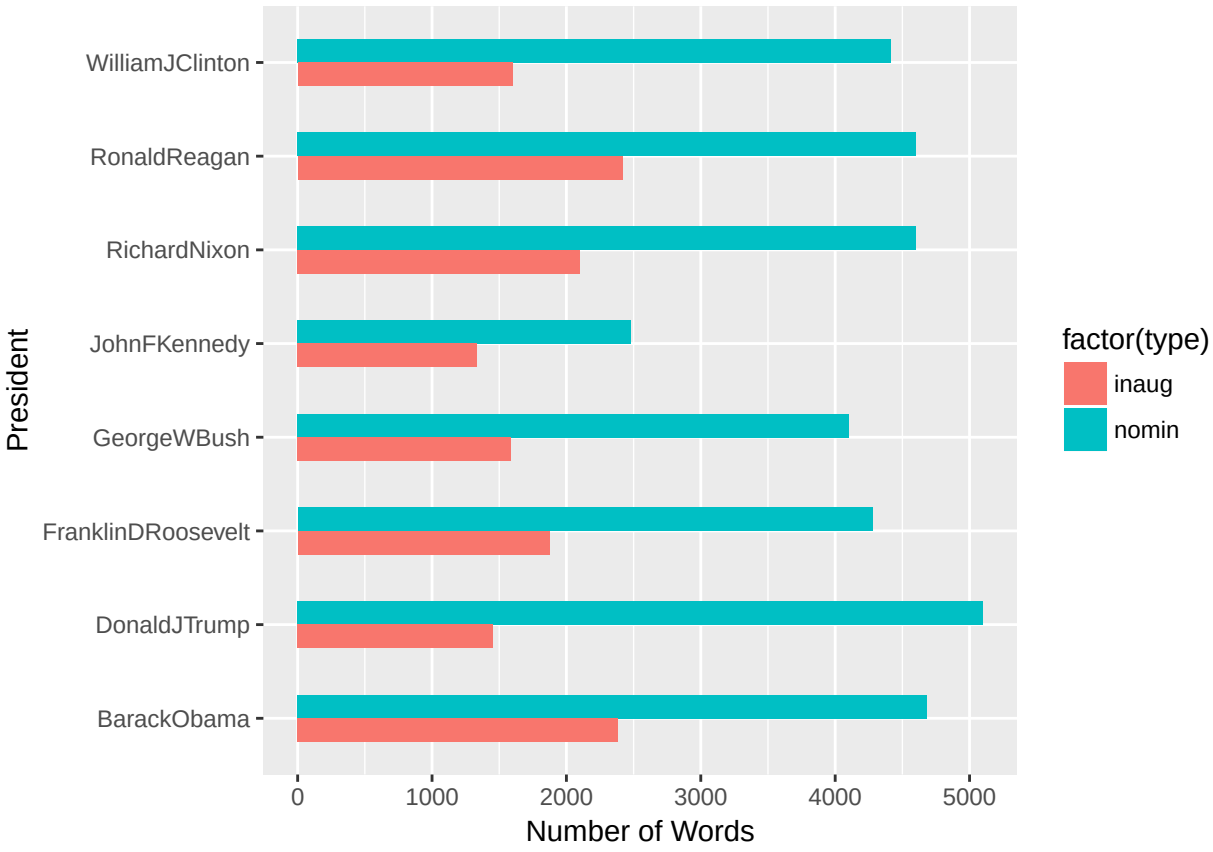
I select 8 well-known presidents here as examples of this part to explore the similarities and differences between nomination speeches and inaugration speeches.

```
sel<-c( "BarackObama", "DonaldJTrump", "FranklinDRoosevelt", "GeorgeWBush", "JohnFKennedy",   "RichardN
```

a. sentences analysis

First, consider the length of speeches. Here the speeches all comes from the term 1 of the presidents

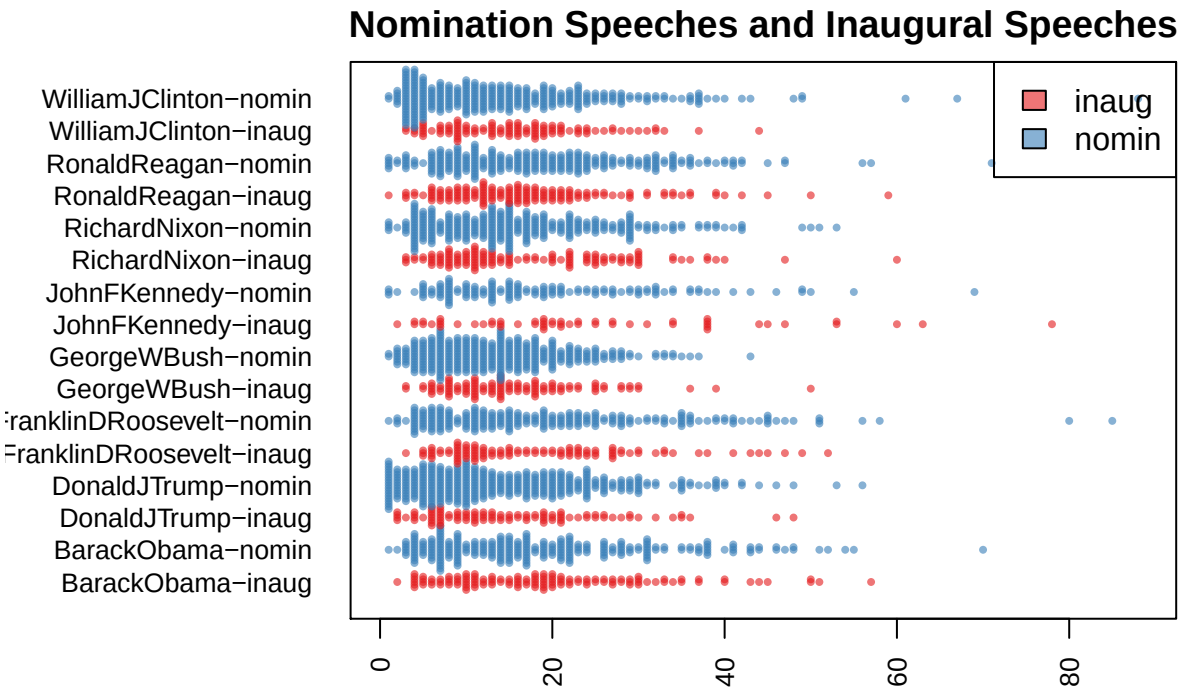
```
nomin_inaug_list<-sentence_list%>%
  filter(Win=="Yes"|Win=="yes",type!="farewell",File%in%sel,Term==1)%>%
  arrange(desc(File))

nomin_inaug_list$x<-apply(nomin_inaug_list[,c("File","type")],1,paste,collapse="-")
nomin_inaug_list$FileOrdered<-reorder(nomin_inaug_list$x,nomin_inaug_list$word.count,mean,order=T)
nomin_inaug_list<-nomin_inaug_list%>%
  arrange(desc(FileOrdered))

Speech_length<-nomin_inaug_list%>%
  group_by(File,type)%>%
  dplyr::summarise(
  words=mean(Word),
  mean_words=mean(word.count,na.rm=T)
 )

par(oma=c(2,2,2,2),mar=c(2,2,2,2))
ggplot(Speech_length,aes(x=File,y=words,fill=factor(type)))+
  geom_bar(stat="identity",position="dodge",width = 0.5)+
  coord_flip() +
  xlab("President")+ylab("Number of Words")
```



Comapre to nomination speeches, most inaugration speeches have much less words. If we take the length of sentences into consideration, is this phenomenon because of the short sentences or long sentences?

Let's make a bee swarm plot for the length of sentences

```
par(oma=c(2,2,2,2),mar=c(2,7,2,0),bty="o")
color=alpha(brewer.pal(3, "Set1"), 0.6)[1:2]
beeswarm(word.count~x,
         data=nomin_inaug_list,
         horizontal = TRUE,
         pch=16, col=rep(color,19),
         cex=0.55, cex.axis=0.8, cex.lab=0.8,
         spacing=5/nlevels(nomin_inaug_list$FileOrdered),
         las=2, ylab="", xlab="Number of words in a sentence.",
         main="Nomination Speeches and Inaugural Speeches")
legend("topright",legend = c("inaug","nomin"),fill=color)
```



In the bee swarm plot, the part of short sentences of nominations are much thicker than inauguration speeches while the number of long sentences of them are relatively close, indicating presidents use more short sentences in their nomination speeches

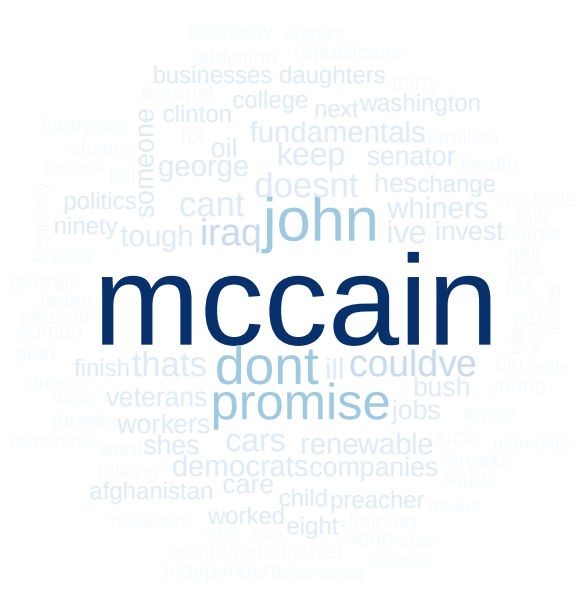
b. Keywords

To find out the keywords of each file, here we use all nomination, inaugration and farewell speeches as base, calcuates the Tf-idf of words to decide if their are playing key roles in their own documents. Then make wordcloud plots for nomination and inaugration speeches respectively with Tf-idf as weights.

```
#compare their wordcloud
text_all<-VCorpus(DirSource("../data/fulltext/"))

text_all<-tm_map(text_all, stripWhitespace)
text_all<-tm_map(text_all, content_transformer(tolower))
text_all<-tm_map(text_all, removeWords, character(0))
text_all<-tm_map(text_all, removePunctuation)
text_all<-tm_map(text_all, removeNumbers)
text_all<-tm_map(text_all, removeWords, stopwords("english"))
tdm.all<-TermDocumentMatrix(text_all)
tdm.tidy=tidy(tdm.all)

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

dtm <- DocumentTermMatrix(text_all,
                          control = list(weighting = function(x)
                                             weightTfIdf(x,
                                                         normalize =FALSE),
                                         stopwords = TRUE))
text.dtm=tidy(dtm)

split_filename<-function(vec){
```

```
  x<-strsplit(vec,split = "")[[1]]
  matches<-gregexpr(pattern ="[A-Z][a-zA-Z]+-" , text = vec)
  p.name<-regmatches(vec, matches)
  p.name<-strsplit(p.name[[1]],split = "")[[1]]
  p.name<-paste(p.name[-length(p.name)],collapse = "")
  term<-x[length(x)-4]

  return(c(as.character(x[1]),as.character(p.name),as.character(term)))
}
text.dtm[,4:6]<-laply(text.dtm$document,split_filename)
colnames(text.dtm)[4:6]<-c("type","president","Term")

i=1
t=1
plot_word_cloud<-function(i,t){
text_inaug<-text.dtm%>%
  filter(president==sel[i],type=="i",Term==t)
text_nomin<-text.dtm%>%
  filter(president==sel[i],type=="n",Term==t)

par(mfrow=c(1,2),mar=c(1,1,1,1))
wordcloud(text_nomin$term, text_nomin$count/sum(text_nomin$count),
          scale=c(4,0.3),
          max.words=100,
          min.freq=1,
          random.order=FALSE,
          rot.per=0.1,
          use.r.layout=T,
          random.color=FALSE,
          colors=brewer.pal(9,"Blues"),main="nomin")
text(0.5,-0.1,"\"Nomin\"",col="black",cex=1.2)

wordcloud(text_inaug$term, text_inaug$count/sum(text_inaug$count),
          scale=c(4,0.3),
          max.words=100,
          min.freq=1,
          random.order=FALSE,
          rot.per=0.1,
          use.r.layout=T,
          random.color=FALSE,
          colors=brewer.pal(9,"Oranges"),main="inagu")
text(0.5,-0.1,"\"Inaug\"",col="black",cex=1.2)
title(paste(sel[i],"-",t), outer = TRUE,line=-1.5,col=brewer.pal(9,"Greys"),cex=2)
}
plot_word_cloud(1,1)#Obama financial crisis
```

BarackObama − 1

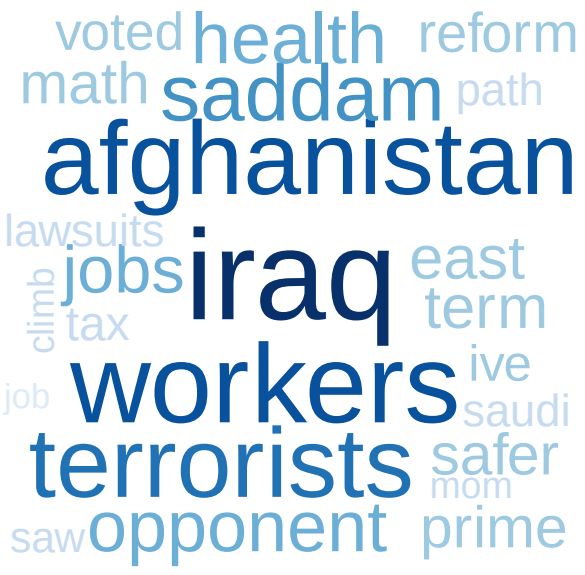


"Nomin"

"Inaug"

```
plot_word_cloud(4,2)#Bush iraq war
```

GeorgeWBush − 2



"Nomin"

"Inaug"

The keywords selected for each speech is very accurate. For example, the keyword of Obama's nomination speech is his competitor in the election, "Mccain" and the "icy", "stroms" showed in his inaugration speech reflect the financial crisis just happened. In Bush's inaugration wordcloud, there are big "Iraq" and

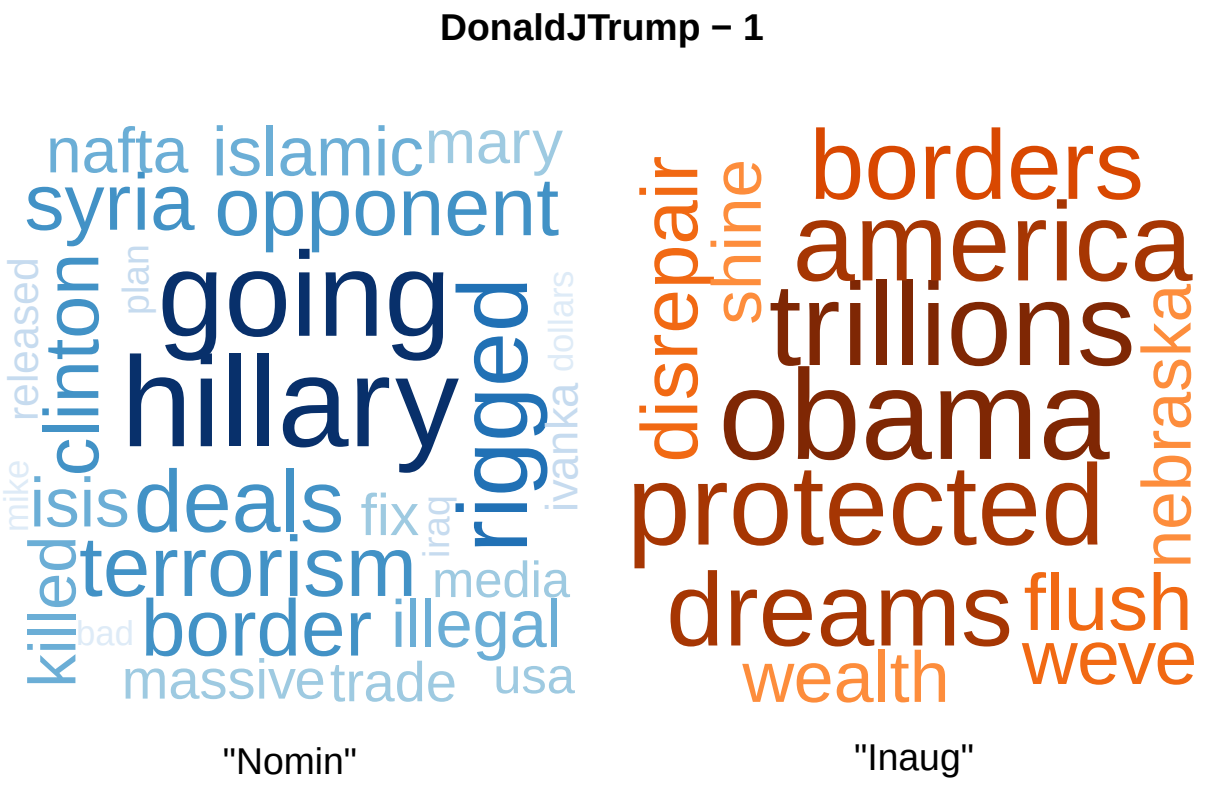
"Afghanistan", indicating the Iraq war began in 2003.

From the word clouds, we can see the nomination speeches always have more words and these words are very specific, like the country names and people names, while Inauguration speeches intend to contain the "bigger" words. For example, Bush's "iraq" and "terrorist" are replaced by "tyranny" and "defended".

Generally speaking, in nomination speeches, speakers provide people detailed plans and promises in order to receive supports, while the inaugration speech do not have that strong purpose. While making inaugration speeches, speakers are more likely to summarize the situation of the country and the his main claims, they would not give that much details to people under the limitation of time.

Use Trump's two speeches as an example.

```
plot_word_cloud(2,1)#Trump
```



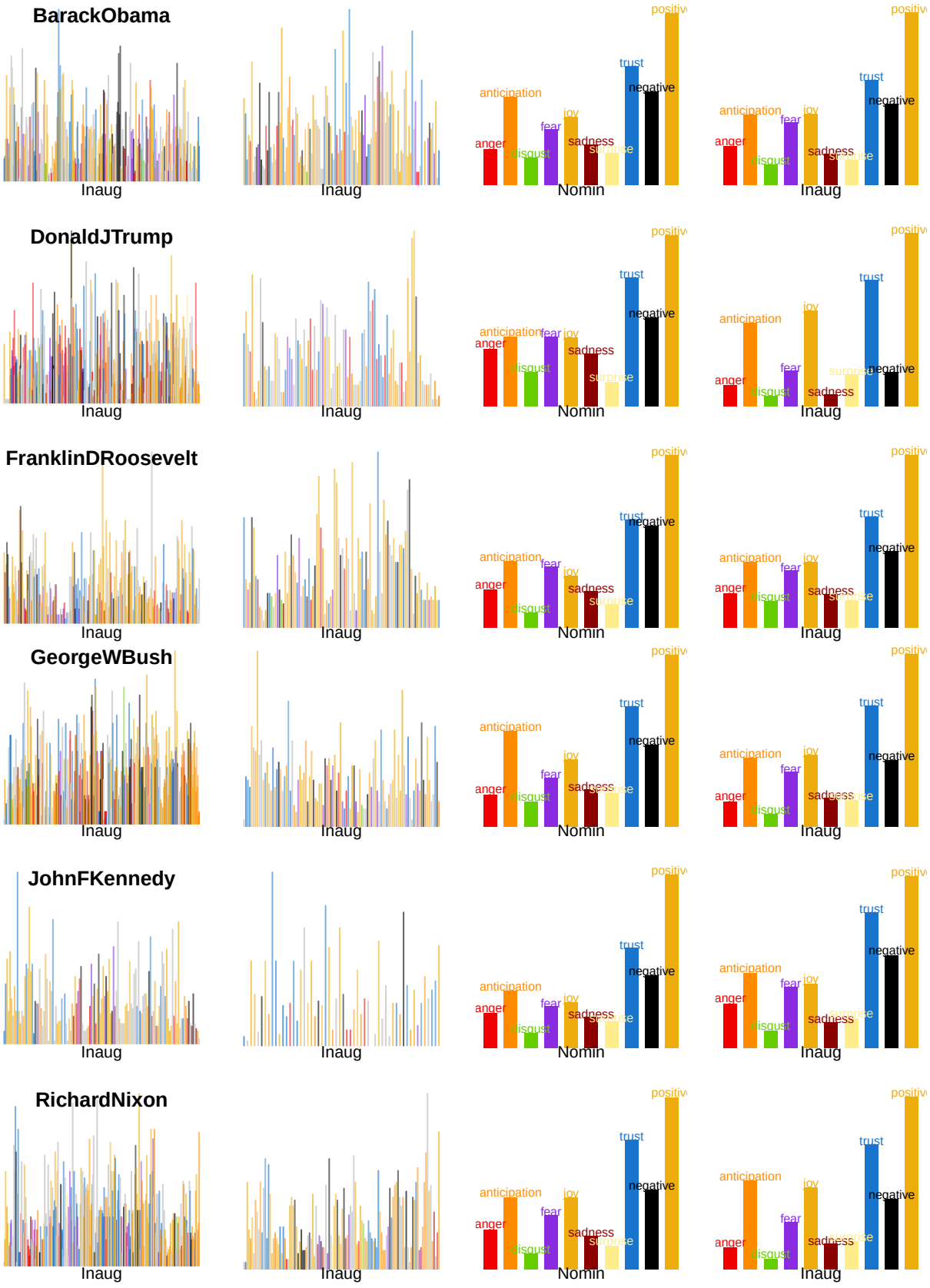
In his nomination speech, he mentioned many people like Obama and Hillary, "Hillary" is even the key word of this speech, reflecting that the intense competition between them. Besides, his families, Melania and Ivanka also shown in order to build his personal image. Except for these people, there are many countries and organizations names and most of them are obviously asscociated with "immigration", "terrorism","border" words. As we all know, the key ideas of Trump during his presidential campaign are dealing with these issues with some strong approaches.

In contrast, names and sharp opinions dissapear in his inaugration speech and most of them are replaced by some "big" words like "dreams", "protect" and "transferring". Though we could still conclude the main ideas of Trump from the inaugration speech, compare to nomination speech his distinctive characteristics become vague.

c. Sentimental Analysis

```
col.use=c("lightgray", "red2", "darkorange",
            "chartreuse3", "blueviolet",
            "darkgoldenrod2", "darkred",
```

```
            "lightgoldenrod1", "dodgerblue3",
            "black", "darkgoldenrod2")

par(mfrow=c(3,4), mar=c(1,1,1,0.5), bty="n", xaxt="n", yaxt="n", font.main=1)
for(i in c(1:6)){
  p1=f.plotsent.len(In.list=sentence_list, InFile=sel[i],
                InType="nomin", InTerm=1,President=NULL)
  title(sel[i],line=-1,col=brewer.pal(9,"Greys"),cex=2,font.main=2)
  title(sub = "Inaug",line=-0.3,col=brewer.pal(9,"Greys"),cex=2)

  p3=f.plotsent.len(In.list=sentence_list, InFile=sel[i],
                InType="inaug", InTerm=1,President=NULL)
  title(sub = "Inaug",line=-0.3,col=brewer.pal(9,"Greys"),cex=2)

  nomin_sent<-colMeans(sentence_list%>%
                     filter(File==sel[i],type=="nomin")%>%
                     select(anger:positive))
  p2=barplot(nomin_sent/sum(nomin_sent),col=col.use[2:11],
             border = NA,width=1,space=0.5,ylim=c(0,max(nomin_sent)/sum(nomin_sent)+0.015))
  text(x=p2,y=nomin_sent/sum(nomin_sent)+0.005,labels = colnames(sentence_list)[12:21],cex=0.8,col=col.u
  title(sub = "Nomin",line=-0.3,col=brewer.pal(9,"Greys"),cex=2)

  inaug_sent<-colMeans(sentence_list%>%
                     filter(File==sel[i],type=="inaug")%>%
                     select(anger:positive))

  p4=barplot(inaug_sent/sum(inaug_sent),col=col.use[2:11],
             border = NA,width=1,space=0.5,ylim=c(0,max(inaug_sent)/sum(inaug_sent)+0.015))
  text(x=p4,y=inaug_sent/sum(inaug_sent)+0.005,labels = colnames(sentence_list)[12:21],cex=0.8,col=col.u
  title(sub = "Inaug",line=-0.3,col=brewer.pal(9,"Greys"),cex=2)

  #title(sel[i], outer = TRUE,line=-1,col=brewer.pal(9,"Greys"),cex=2)
}
```



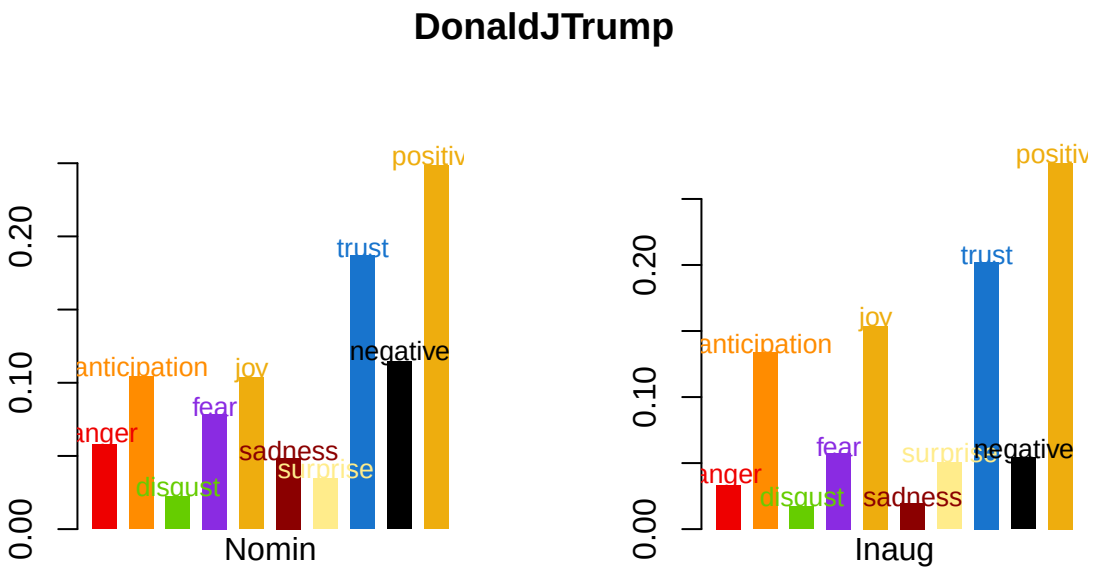
The sentiments presidents show in nomination speeches are more diverse and complicate than their inauguration

speeches, which is consistent with the features of nomination and inaugration speeches. Most of the time, nomination speech gives the speaker more freedom and longer time while the inaugration is much more normal and rigid.

The general speard of sentiments are similar in all these speeches. As we can see, though postive sentiment is presidents' favorite, they would not absolutely ignore or hide the other bad feelings. On the contrary, they utilize bad feelings to incite people.

Besides, most presidents' nomination and inaugration speeches have almost the same distribution of sentiments, indicating that presidents has fixed personal style in speeches, which should be consistent with the personal image he built. But Trump is an exceptation.

```
par(mfrow=c(1,2),par(pty="s"))
i=2
p2=barplot(nomin_sent/sum(nomin_sent),col=col.use[2:11],
           border = NA,width=1,space=0.5,ylim=c(0,max(nomin_sent)/sum(nomin_sent)+0.015),
           xaxt="n")

text(x=p2,y=nomin_sent/sum(nomin_sent)+0.005,
     labels = colnames(sentence_list)[12:21],
     cex=0.8,col=col.use[2:11])
title(sub = "Nomin",line=0,col=brewer.pal(9,"Greys"),cex=2)

inaug_sent<-colMeans(sentence_list%>%
                       filter(File==sel[i],type=="inaug")%>%
                       select(anger:positive))

p4=barplot(inaug_sent/sum(inaug_sent),col=col.use[2:11],
           border = NA,width=1,space=0.5,ylim=c(0,max(inaug_sent)/sum(inaug_sent)+0.015),
           xaxt="n")
text(x=p4,y=inaug_sent/sum(inaug_sent)+0.005,
     labels =colnames(sentence_list)[12:21],
     cex=0.8,col=col.use[2:11])

title(sub = "Inaug",line=0,col=brewer.pal(9,"Greys"),cex=2)
title(sel[i],outer = T,line=-3)
```



Compare to the other presidents, distribution of sentiments of Trump's inaugration speech is obviously

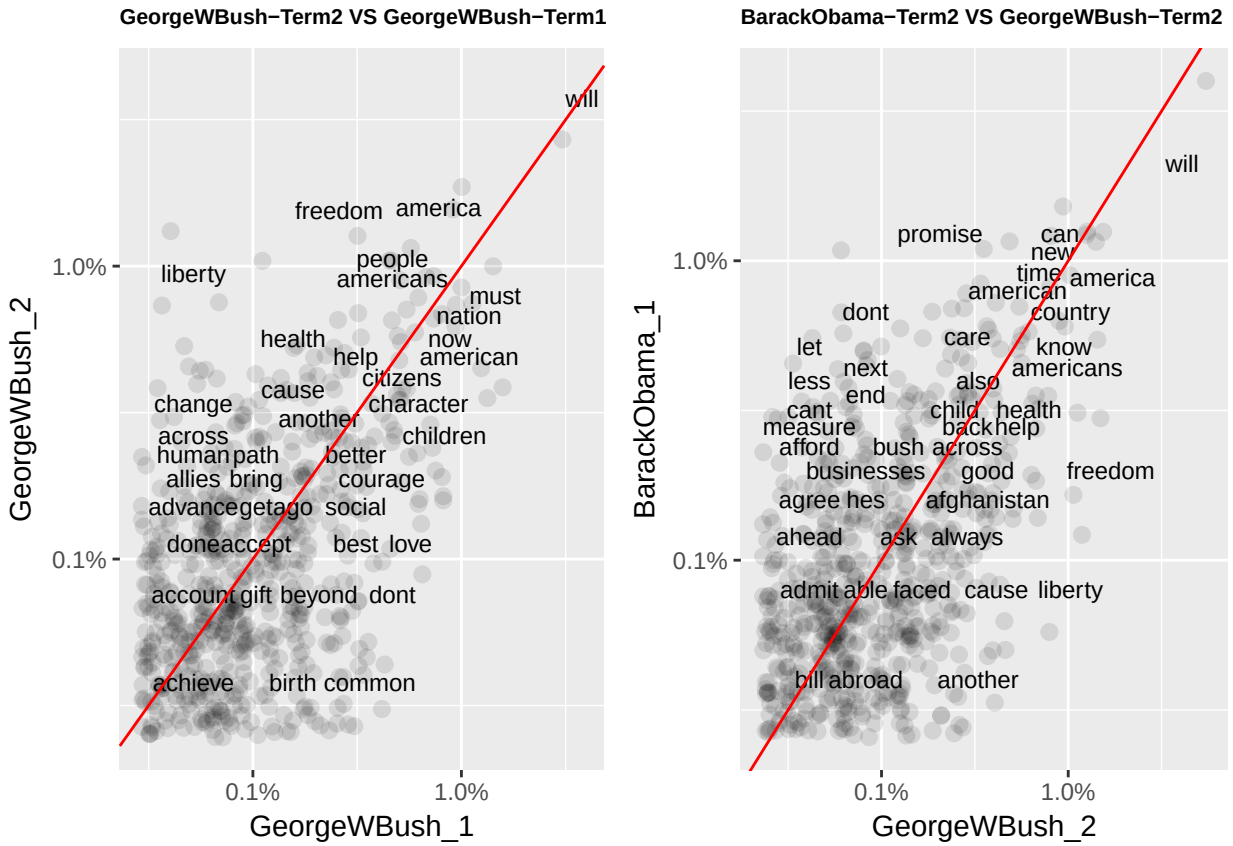
different from his nomination speech and looks distinctive among all these speeches.
His inaugration emphasizes the opstimistic sentiments like “anticipation”, “joy” and “trust” while reduce the percentages of “anger”, “disgust”, “sadness” and “negative” to a very low level. Personally speaking, it has strong desire to encourage people by avoiding the negative feelings.

PART II: Similarities between 2 Successive Inaugrations

By exploring the similar words of two successive inaugration speeches , we can have a insight about the issues which both 2 presidents care about. They might be problems have not yet been resolved or aspects both two presidents focus on. Here I use the latest 5 inauguation speeches as an example.

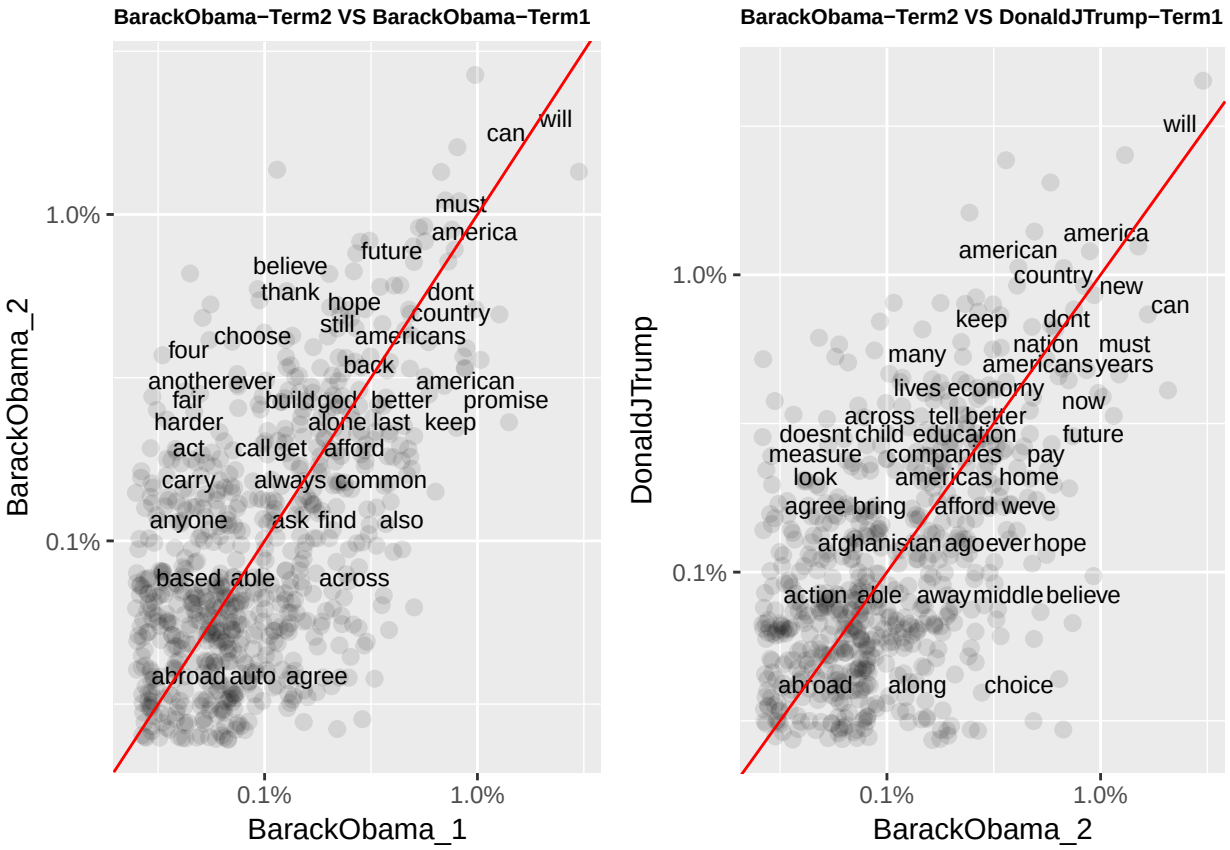
```
Obama1<-tdm.tidy%>%
  filter(document==paste("inaug","BarackObama","-",1,".txt",sep="")|document==paste("nomin","BarackObama
  group_by(term)%>%
  summarise(
    count=sum(count)
  )
Obama2<-tdm.tidy%>%
  filter(document==paste("inaug","BarackObama","-",2,".txt",sep="")|document==paste("nomin","BarackObama
  group_by(term)%>%
  summarise(
    count=sum(count)
  )

Trump<-tdm.tidy%>%
  filter(document==paste("inaug","DonaldJTrump","-",1,".txt",sep="")|document==paste("nomin","BarackOba
  group_by(term)%>%
  summarise(
    count=sum(count)
  )

Bush1<-tdm.tidy%>%
  filter(document==paste("inaug","GeorgeWBush","-",1,".txt",sep="")|document==paste("nomin","GeorgeWBus
  group_by(term)%>%
  summarise(
    count=sum(count)
  )

Bush2<-tdm.tidy%>%
  filter(document==paste("inaug","GeorgeWBush","-",2,".txt",sep="")|document==paste("nomin","GeorgeWBus
  group_by(term)%>%
  summarise(
    count=sum(count)
  )

join_list1<-merge(Bush1,Bush2,by="term",all.x=F,all.y = F)
join_list1[,2]<-join_list1[,2]/sum(join_list1[,2],na.rm=T)
join_list1[,3]<-join_list1[,3]/sum(join_list1[,3],na.rm=T)
colnames(join_list1)=c("Term","GeorgeWBush_1","GeorgeWBush_2")


p1=ggplot(join_list1, aes(GeorgeWBush_1, GeorgeWBush_2)) +
  geom_jitter(alpha=0.1, size = 2.5, width = 0.25, height = 0.25) +
  geom_text(aes(label = Term), check_overlap = TRUE, vjust = 1.5,cex=3) +
```

```
  scale_x_log10(labels = percent_format()) +
  scale_y_log10(labels = percent_format()) +
  geom_abline(color = "red")

join_list2<-merge(Bush2,Obama1,by="term",all.x=F,all.y = F)
join_list2[,2]<-join_list2[,2]/sum(join_list2[,2],na.rm=T)
join_list2[,3]<-join_list2[,3]/sum(join_list2[,3],na.rm=T)
colnames(join_list2)=c("Term","GeorgeWBush_2","BarackObama_1")


p2=ggplot(join_list2, aes( GeorgeWBush_2,BarackObama_1)) +
  geom_jitter(alpha=0.1, size = 2.5, width = 0.25, height = 0.25) +
  geom_text(aes(label = Term), check_overlap = TRUE, vjust = 1.5,cex=3) +
  scale_x_log10(labels = percent_format()) +
  scale_y_log10(labels = percent_format()) +
  geom_abline(color = "red")

join_list3<-merge(Obama1,Obama2,by="term",all.x=F,all.y = F)
join_list3[,2]<-join_list3[,2]/sum(join_list3[,2],na.rm=T)
join_list3[,3]<-join_list3[,3]/sum(join_list3[,3],na.rm=T)
colnames(join_list3)=c("Term","BarackObama_1","BarackObama_2")
library(scales)

p3=ggplot(join_list3, aes(BarackObama_1,BarackObama_2)) +
  geom_jitter(alpha=0.1, size = 2.5, width = 0.25, height = 0.25) +
  geom_text(aes(label = Term), check_overlap = TRUE, vjust = 1.5,cex=3) +
  scale_x_log10(labels = percent_format()) +
  scale_y_log10(labels = percent_format()) +
  geom_abline(color = "red")


join_list4<-merge(Obama2,Trump,by="term",all.x=F,all.y = F)
join_list4[,2]<-join_list4[,2]/sum(join_list4[,2],na.rm=T)
join_list4[,3]<-join_list4[,3]/sum(join_list4[,3],na.rm=T)
colnames(join_list4)=c("Term","BarackObama_2","DonaldJTrump")


p4=ggplot(join_list4, aes(BarackObama_2, DonaldJTrump)) +
  geom_jitter(alpha=0.1, size = 2.5, width = 0.25, height = 0.25) +
  geom_text(aes(label = Term), check_overlap = TRUE, vjust = 1.5,cex=3) +
  scale_x_log10(labels = percent_format()) +
  scale_y_log10(labels = percent_format()) +
  geom_abline(color = "red")


grid.arrange(
  p1+ ggtitle ("GeorgeWBush-Term2 VS GeorgeWBush-Term1")+
    theme(plot.title = element_text(size = 8, face = "bold")),
  p2+ ggtitle ("BarackObama-Term2 VS GeorgeWBush-Term2")+
    theme(plot.title = element_text(size = 8, face = "bold")),
  ncol = 2, nrow = 1)
```



In 2 consecutive terms of George W Bush, expect for those common words like "america", "will", "nation" and "people", we can notice that there are also "freedom" ,"help" ,"health" and "children". These words remind us the "No child left behind Act" and "Health Savings Account" proposed and signed by Bush. From the second plot, both Obama and Bush show interest in "health". As we all know, while Bush got the "Health Saving Account", Obama signed the "Obamacare", which is one of his major contributions.

```
grid.arrange(
  p3+ ggtitle ("BarackObama-Term2 VS BarackObama-Term1")+
    theme(plot.title = element_text(size = 8, face = "bold")),
  p4+ ggtitle ("BarackObama-Term2 VS DonaldJTrump-Term1")+
    theme(plot.title = element_text(size = 8, face = "bold")),
  ncol = 2, nrow = 1)
```



The shared key words of two inaugration speeches of Obama contain "jobs", "pay", "afford" and "crisis", indicating the 2007-2008 financial crisis that Obama encountered. In the right plot, "economy" and "companies" are mentioned by both Trump and Obama, so the Business is not only Obama's big deal, but also Trump's focus.

PART III: Some Interesting Details

a. Short Sentences

Short Sentences are useful in increasing the momentum of the speeches. Using short sentences in the middle of long sentences could make the speech structure not that boring and incite the listeners' mood.

Here 3 types of speeches are taken into consideration in order to explore the presidents' habits of speaking.

(1).What are the presidents' favorite short sentences?

```
short_sentences<-sentence_list%>%
  filter(word.count<=8,nchar(sentences)>3,Win=="yes")
short_sentences$sentences<-tolower(short_sentences$sentences)
short_sentences<-short_sentences%>%
  filter(!sentences%in%c("the president.","audience.",'boo-o-o!','no!','no-o-o!',"four more years!","4 m

short_sentences1<-short_sentences%>%
  group_by(sentences)%>%
  summarise(
    times=length(word.count),
    max_times=max(table(President)),
```

```
    fav_president=paste(names(which.max(table(President))),collapse=";")
    #,presidents=paste(names(table(President)),table(President),collapse="; ",sep="-")
  )%>%

  arrange(desc(times))
#short_sentences1<-short_sentences1[,c(1,2,4)]
print(head(short_sentences1,10))
```

```
## # A tibble: 10 x 4
##                                sentences times max_times
##                                    <chr> <int>     <int>
##  1                             thank you.    39        15
##  2                                reagan!     6         6
##  3                             all right.     5         5
##  4                     thank you so much.     5         5
##  5                   thank you very much.     5         2
##  6                          we can do it.     5         5
##  7                                  amen.     4         1
##  8 god bless you, and god bless america.     4         2
##  9   my opponent says no, but i say yes.     4         4
## 10                               we will.     4         4
## # ... with 1 more variables: fav_president <chr>
```

It is not surprise that the favorite sentence of presidents is "thank you" and Obama is its biggest fan. I also found out Reagan repeat "all right" 5 times, it might be his mantra (After checking the txt file, the 6 "reagan!" shown here is the recorded audience members' voice).

And we also see some slogans of presidents, like Clinton's "we can do it", George Bush's "my opponent says no, but i say yes", George W Bush's "we will". There is no doubt that many of them use the short sentences as parallelism in their speeches.

(2). Who is the president love short sentences most?

```
short_sentences2<-short_sentences%>%
  filter(nchar(sentences)>2)%>%
  group_by(File)%>%
  summarise(
    times=round(length(word.count)/length(unique(Term))/length(unique(type))))%>%
    # fav_sentence=names(which.max(table(sentences))),
    # max_times=max(table(sentences))
    #,sentences=paste(names(table(sentences)),table(sentences),collapse="; ",sep="-")
    arrange(desc(times))
head(short_sentences2,6)
```

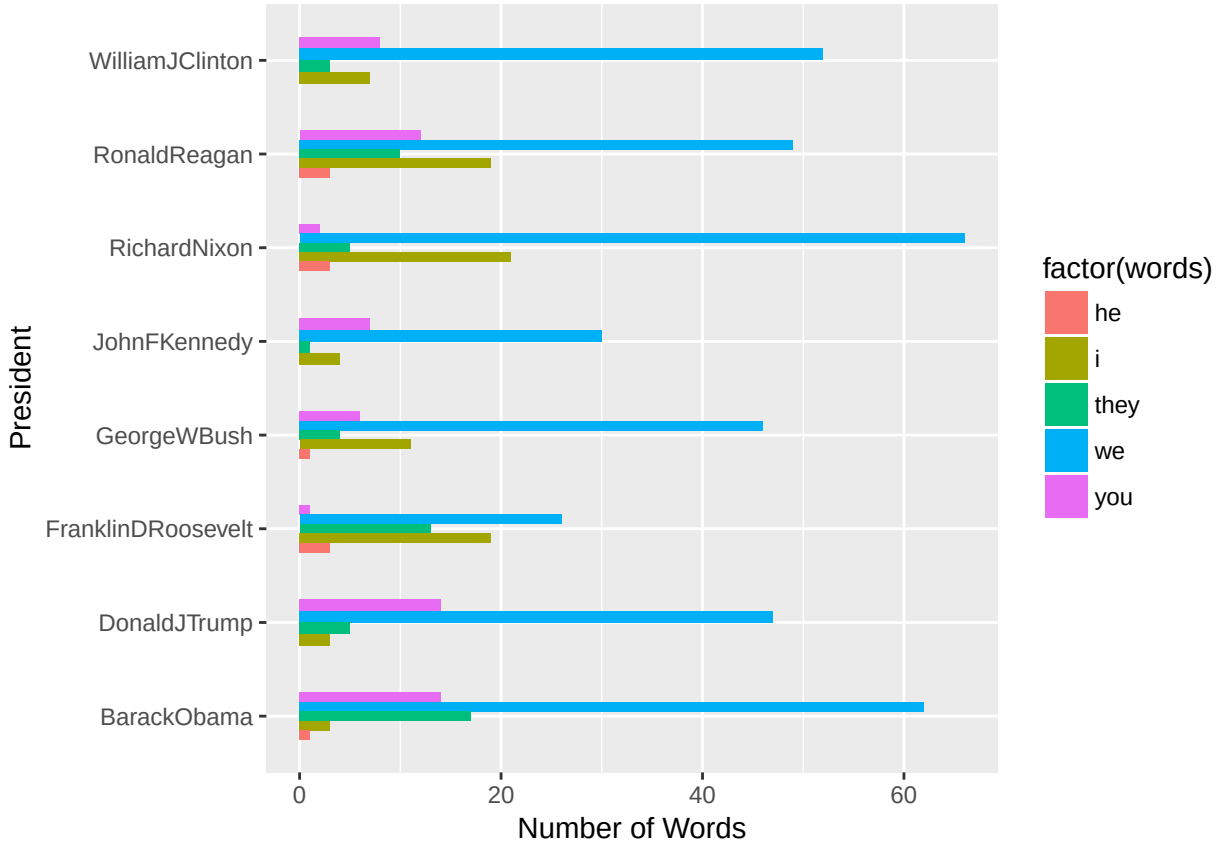
```
## # A tibble: 6 x 2
##             File times
##            <chr> <dbl>
## 1   DonaldJTrump    64
## 2  HerbertHoover    40
## 3     GeorgeBush    33
## 4    BarackObama    28
## 5 WarrenGHarding    27
## 6    GeraldRFord    25
```

The champion is Trump, his each speech has 64 short sentences in average. The presidents we are familar

with like Obama and George Bush love using short sentences in their speeches too. I guess that the recent presidents have more short sentences than before because people today prefer simple expression rather than the speaking in long-winded sentences.

b. Personal pronoun of Presidents

```
tidy.subject_word <- sentence_list %>%
  select(File,type,Term,sentences)%>%
  unnest_tokens(words, sentences)%>%
  filter(words%in%c("we","i",'you',"he","she","they"),Term==1,type=="inaug")%>%
  group_by(File,type,Term,words)%>%
  summarise(
    # File=File[1],
    # Term=Term[1],
    # type=type[1],
    count=length(File)
  )%>%
  arrange(desc(count))

ggplot(tidy.subject_word%>%
         filter(File%in%sel)
         ,aes(x=File,y=count,fill=factor(words)))+
  geom_bar(stat="identity",position="dodge",width = 0.5)+
  coord_flip() +
  xlab("President")+ylab("Number of Words")
```



About the first-person pronouns, most of the time presidents use “we” instead of “I” and even though second

and third-person pronouns have much lower frequency than "we", some of them are higher than the frequency of "I". It is obvious that presidents lower the importance of themselves in inaugration speeches. Instead, they empahsize they are part of citizens. I'm impressed that even a conficent person like Trump decreases the frequency of "I" to such a lower level.

It's also noticeable that none of the president I selected mention "she" in their inaugration speeches.

Part IV: Topic Modeling

Let's have look about the popular topics of presidents' speeches. Topic Model will help us find the topics and conclude their keywords.

a. Build the Model

(1) make DocumentTermMatrix with the snipets of sentences

```
#ready for topic modeling
#make snipets
corpus.list=sentence_list[2:(nrow(sentence_list)-1), ]
sentence.pre=sentence_list$sentences[1:(nrow(sentence_list)-2)]
sentence.post=sentence_list$sentences[3:(nrow(sentence_list)-1)]
corpus.list$snipets=paste(sentence.pre, corpus.list$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, ]

docs <- Corpus(VectorSource(corpus.list$snipets))
#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"))
#remove whitespace
docs <- tm_map(docs, stripWhitespace)
#Stem document
docs <- tm_map(docs,stemDocument)

dtm <- DocumentTermMatrix(docs)
rownames(dtm) <- paste(corpus.list$type, corpus.list$File,
                       corpus.list$Term, corpus.list$sent.id, sep="_")
rowTotals <- apply(dtm , 1, sum)
dtm  <- dtm[rowTotals> 0, ]

corpus.list=corpus.list[rowTotals>0, ]
```

(2). Run 10 Topics LDA model using Gibbs Sampling

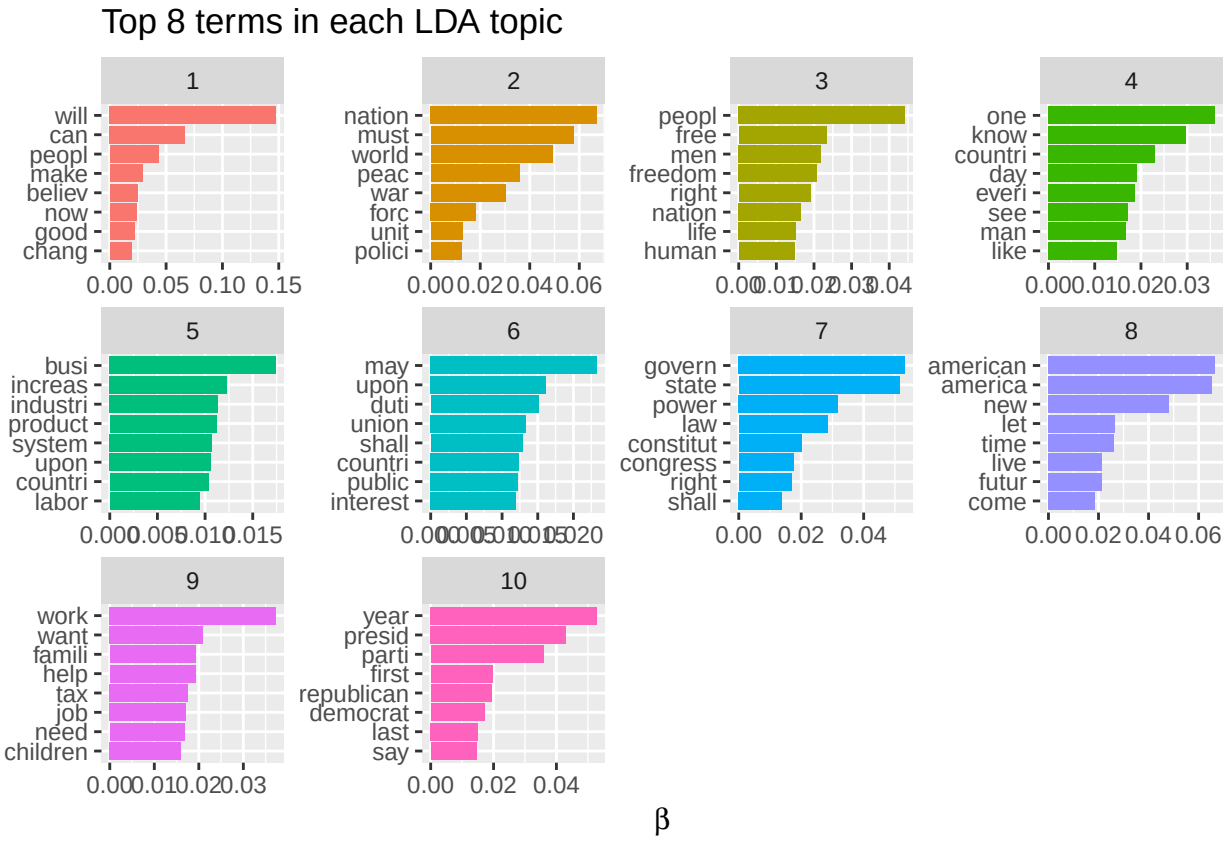
```
#Set parameters for Gibbs sampling
burnin <- 4000
iter <- 2000
```

```
thin <- 500
seed <-list(2003,5,63,100001,765)
nstart <- 5
best <- TRUE
#Number of topics
k <- 10
# ldaOut10_1 <-LDA(dtm, 10, method="Gibbs", control=list(nstart=nstart,
#                                                        seed = seed, best=best,
#                                                        burnin = burnin, iter = iter,
#                                                        thin=thin))

#load the lda model I trained, in case of rerunning the lda(time consuming) every time i knit the file
load("../output/total_R.RData")
tidy_lda<-tidy(ldaOut10_1)
```

b.Analysis

(1).What are the Top Terms of Each Topic?

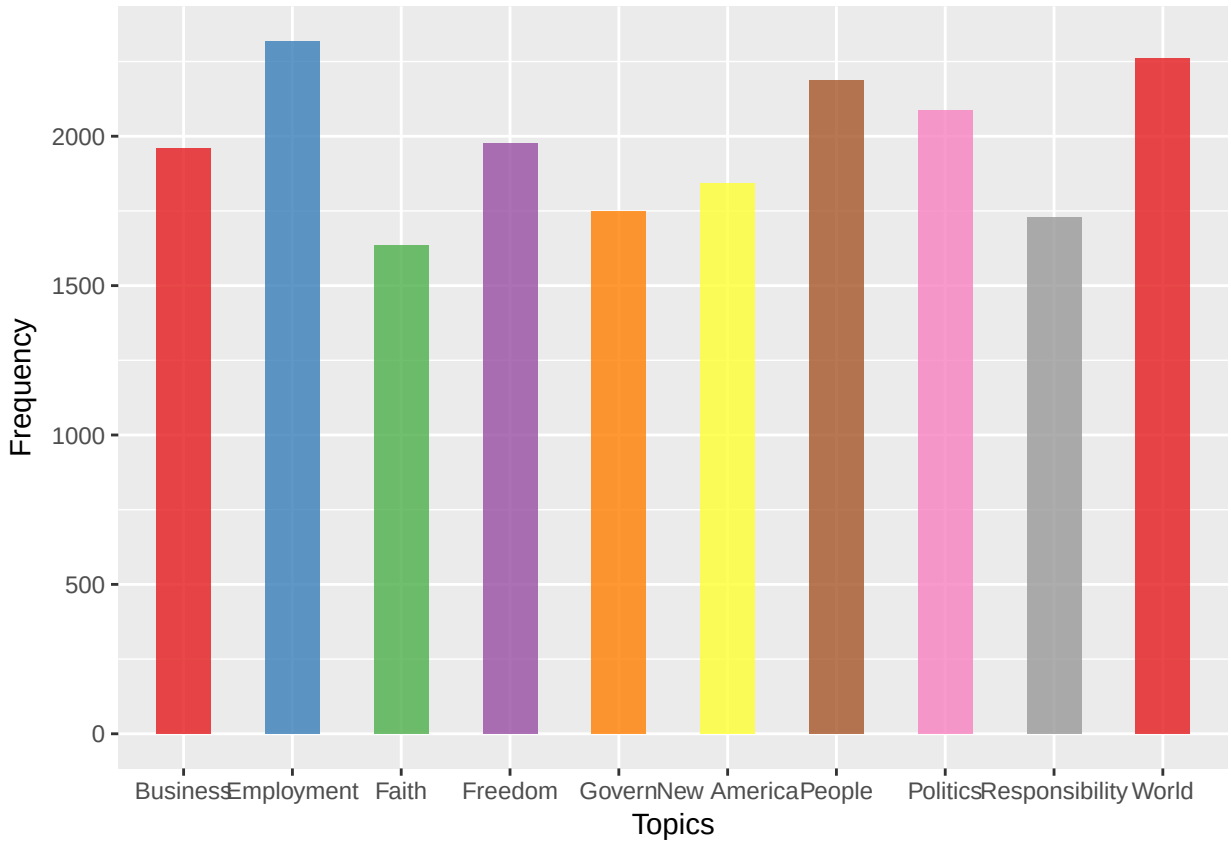
```
top_terms <- tidy_lda %>%
  group_by(topic) %>%
  top_n(8, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)


top_terms %>%
  mutate(term = reorder(term, beta)) %>%
  group_by(topic, term) %>%
  arrange(desc(beta)) %>%
  ungroup() %>%
  mutate(term = factor(paste(term, topic, sep = "__"),
                       levels = rev(paste(term, topic, sep = "__")))) %>%
  ggplot(aes(term, beta, fill = as.factor(topic))) +
  geom_col(show.legend = FALSE) +
  coord_flip() +
  scale_x_discrete(labels = function(x) gsub("__.+$", "", x)) +
  labs(title = "Top 8 terms in each LDA topic",
       x = NULL, y = expression(beta)) +
  facet_wrap(~ topic, ncol = 4, scales = "free")
```



It is noticeable that the values of top terms are not high enough, few of them has exceeded 0.6. Therefore, maybe these terms cannot represent the topics well as we thought. But I conclude them manually, they are "Faith", "World Peace", "Freedom","People", "Business" ,"Responsibility", "Govern", "New America", "Employment", "Politics". Generally speaking, these topics cover almost every aspect a inaugration would mention.

(2).How does the topics distribute?

```
topics.hash=c("Faith","World","Freedom","People", "Business" ,"Responsibility", "Govern", "New America"
ldaOut10.topics <- as.matrix(topics(ldaOut10))

#top 20 terms in each topic
ldaOut10.terms <- as.matrix(terms(ldaOut10,20))
#probabilities associated with each topic assignment
topicProbabilities <- as.data.frame(ldaOut10@gamma)

terms.beta=ldaOut10_1@beta
terms.beta=scale(terms.beta)#scale the columns
corpus.list$ldatopic=as.vector(ldaOut10.topics)
corpus.list$ldahash=topics.hash[ldaOut10.topics]
colnames(topicProbabilities)=topics.hash
corpus.list.df=cbind(corpus.list, topicProbabilities)

#as.data.frame(table(corpus.list$ldahash))
colors=c(alpha(brewer.pal(9, "Set1"), 0.8),alpha(brewer.pal(9, "Set1"), 0.8)[1])
ggplot(corpus.list,aes(ldahash))+geom_bar(width = 0.5,fill=colors)+labs(x="Topics",y="Frequency")
```



As the plot shown, the topics are evenly distributed. Therefore, presidents do not have preference in these topics.

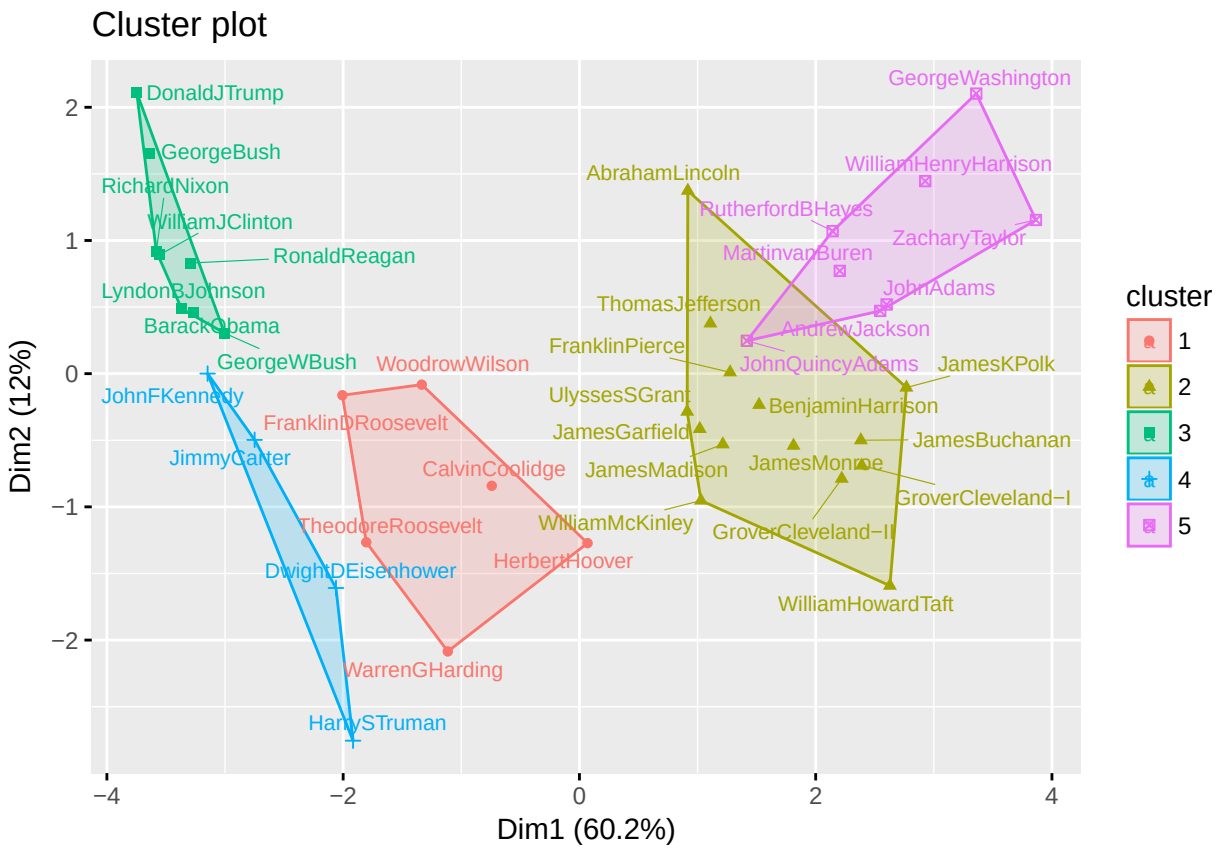
(3). Clustering

According to the mean beta values for the snipets of each topic, we give each inaugration speech 10 topic values representing its probability for belonging to each topic.

```
par(mar=c(1,1,1,1))
topic.summary=as_tibble(corpus.list.df)%>%
              filter(type%in%c("inaug"))%>%
              select(File, Faith:Politics)%>%
              group_by(File)%>%
              summarise_each(funs(mean))

#topic.summary$max.topic<-topics.hash[apply(topic.summary[,2:11],1,which.max)]
#normalization
# f1<-function(vec){
#   return(vec/sum(vec))
# }
# topic.summary[,2:11]<-apply(topic.summary[,2:11],1,f1)
topic.summary=as.data.frame(topic.summary)
rownames(topic.summary)=topic.summary[,1]

topic.summary=as.data.frame(topic.summary)
rownames(topic.summary)=as.character((topic.summary[,1]))
```

Use k-means algorithm to cluster the presidents according to their inaugration speeches

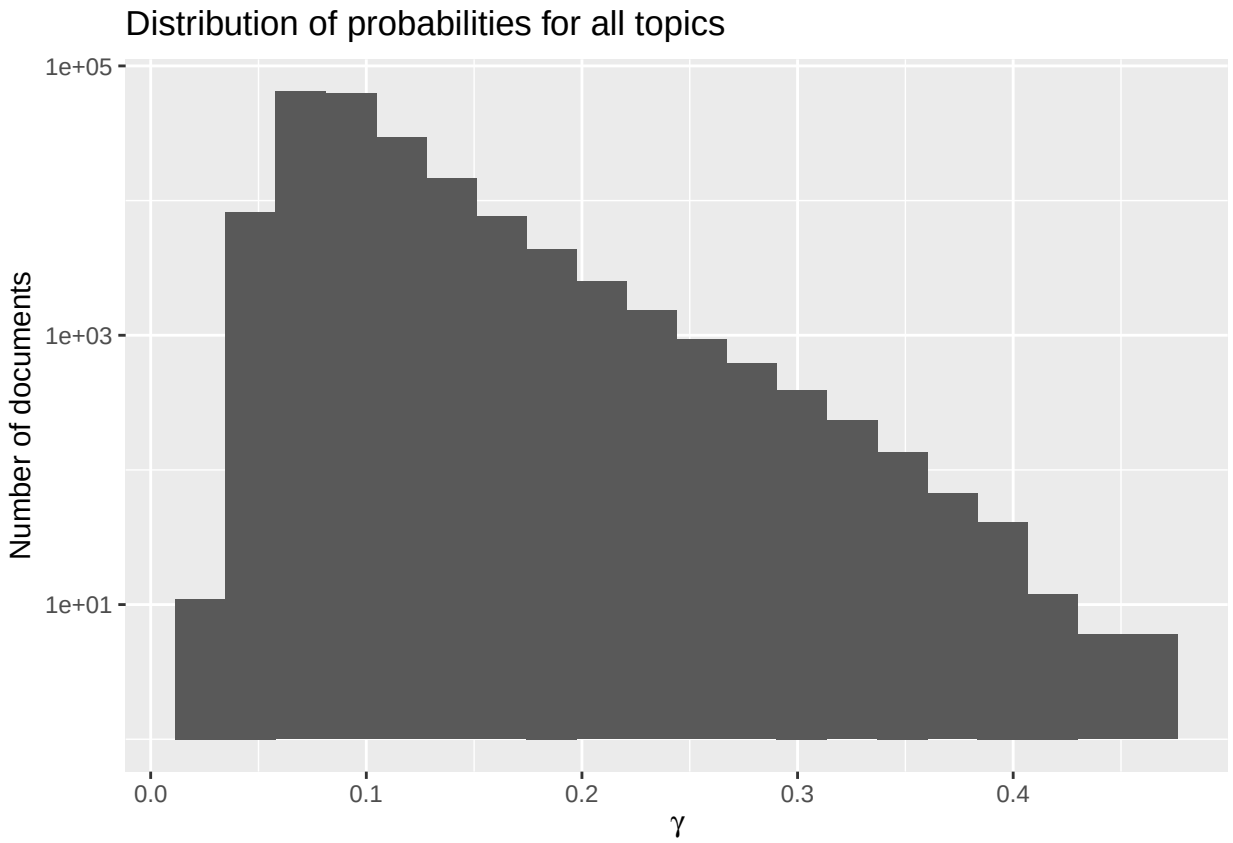
```
km.res=kmeans(scale(topic.summary[,-1]), iter.max=200,
              5)
fviz_cluster(km.res,
             stand=T, repel= TRUE,
             data = topic.summary[,-1],
             show.clust.cent=FALSE,labelsize = 8)
```



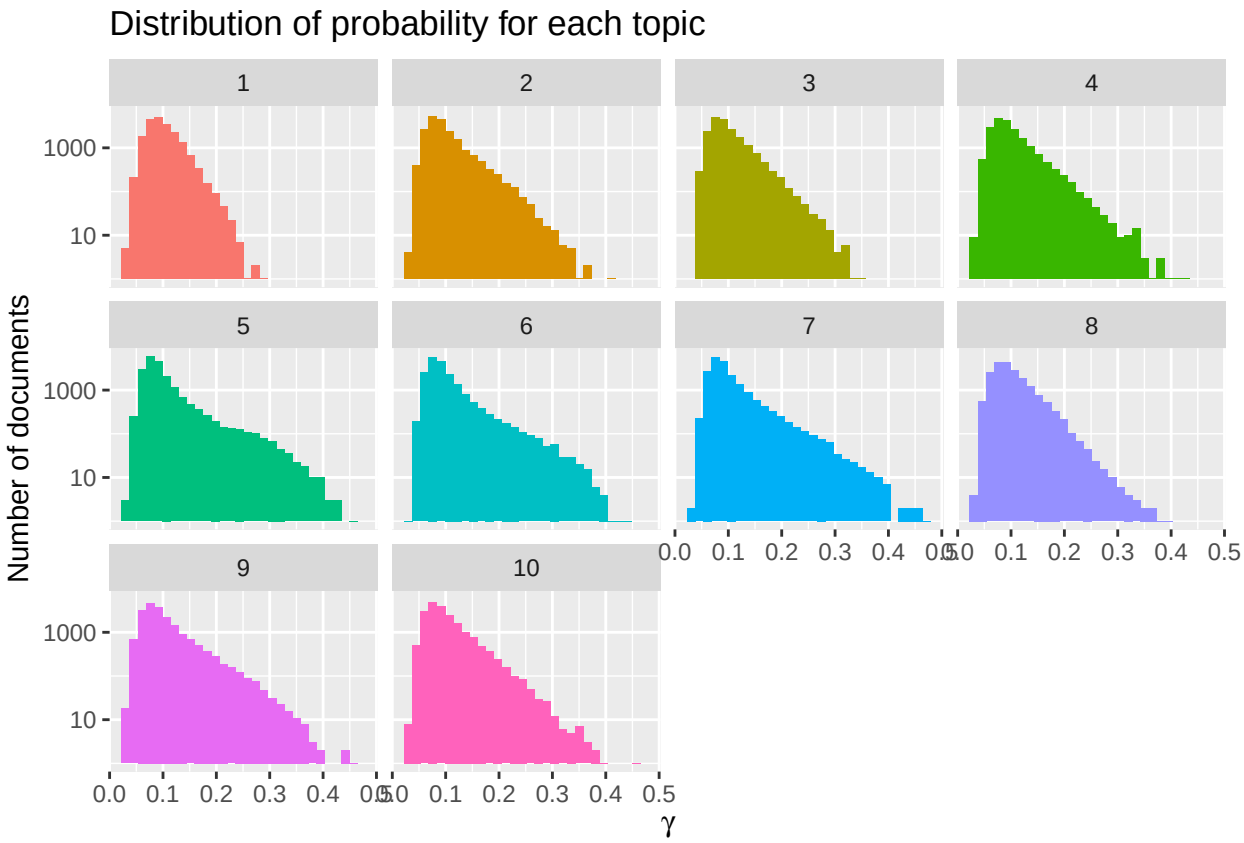
It is interesting to note that the recent presidents like Obama, Trump, two Bush, Clinton are clustered into one group, which is reasonable because they might have similar focuses and problems. But other groups do not have such a distinctive feature, their characteristics need a further research to explore.

It has to be noted that if we check the gamma values of the topic model, we will find the model doesn't perform well. And this is the reason why i make this part the last one and recommend you read it as a appendix.

```
lda_gamma <- tidy(ldaOut10, matrix = "gamma")
#lda_gamma1[,4:6]<-sapply(lda_gamma$document,split_filename)
lda_gamma1<-lda_gamma%>%
  group_by(document,topic)%>%
    summarise(
      mean(gamma)
    )
ggplot(lda_gamma, aes(gamma)) +
  geom_histogram(bins=20) +
  scale_y_log10() +
  labs(title = "Distribution of probabilities for all topics",
       y = "Number of documents", x = expression(gamma))
```




```
ggplot(lda_gamma, aes(gamma, fill = as.factor(topic))) +
  geom_histogram(show.legend = FALSE) +
  facet_wrap(~ topic, ncol = 4) +
  scale_y_log10() +
  labs(title = "Distribution of probability for each topic",
       y = "Number of documents", x = expression(gamma))
```


Distribution of probability for each topic
1
2
3
4
5
6
7
8
9
10
1000
10
Number of documents
0.0
0.1
0.2
0.3
0.4
0.5
γ