Project 1

Ruochen Liu rl2841

February 1, 2017

What's this project for?

Given these inaugurations, I want to focus on the language itself, trying to find some way to evaluate an inauguration mathematically. Lexical density is an important factor in NLP (Natural Language Processing) which is simply a measure of how informative and how formal a text is. More precisely, lexical words are simply nouns, adjectives, verbs, and adverbs. We can tell these words are informative easily, but to tell they are formal requires some papers (about nominalization) of Halliday.

Use lexical functions to get the lexical words and lexical rate of each sentence, and then we can find if there is a trend in results, like the change of density through the text, the density of whole text, and difference between inaugurations. Maybe we can learn from that when we try to write a speech.

1. Load packages we need.

2. Data harvest and data clean.

```
##  [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
```

3. Generate list of sentences.

```
##           President Term Party Words            Date
## 1 George Washington    1  <NA>  1431 April 30, 1789
## 2 George Washington    1  <NA>  1431 April 30, 1789
## 3 George Washington    1  <NA>  1431 April 30, 1789
##
```

```
sentences
## 1
Fellow-Citizens of the Senate and of the House of Representatives:Among the
vicissitudes incident to life no event could have filled me with greater
anxieties than that of which the notification was transmitted by your order,
and received on the 14th day of the present month.
## 2 On the one hand, I was summoned by my country, whose voice I can never
hear but with veneration and love, from a retreat which I had chosen with the
fondest predilection, and, in my flattering hopes, with an immutable
decision, as the asylum of my declining years<U+0097>a retreat which was
rendered every day more necessary as well as more dear to me by the addition
of habit to inclination, and of frequent interruptions in my health to the
gradual waste committed on it by time.
## 3
On the other hand, the magnitude and difficulty of the trust to which the
voice of my country called me, being sufficient to awaken in the wisest and
most experienced of her citizens a distrustful scrutiny into his
qualifications, could not but overwhelm with despondence one who ought to be
peculiarly conscious of his own deficiencies.
##   word.count sent.id
## 1         45       1
## 2         86       2
## 3         56       3
```

4. Data Analysis to Get the lexical density of sentences and inaugurations.

May take 6 minutes or more.

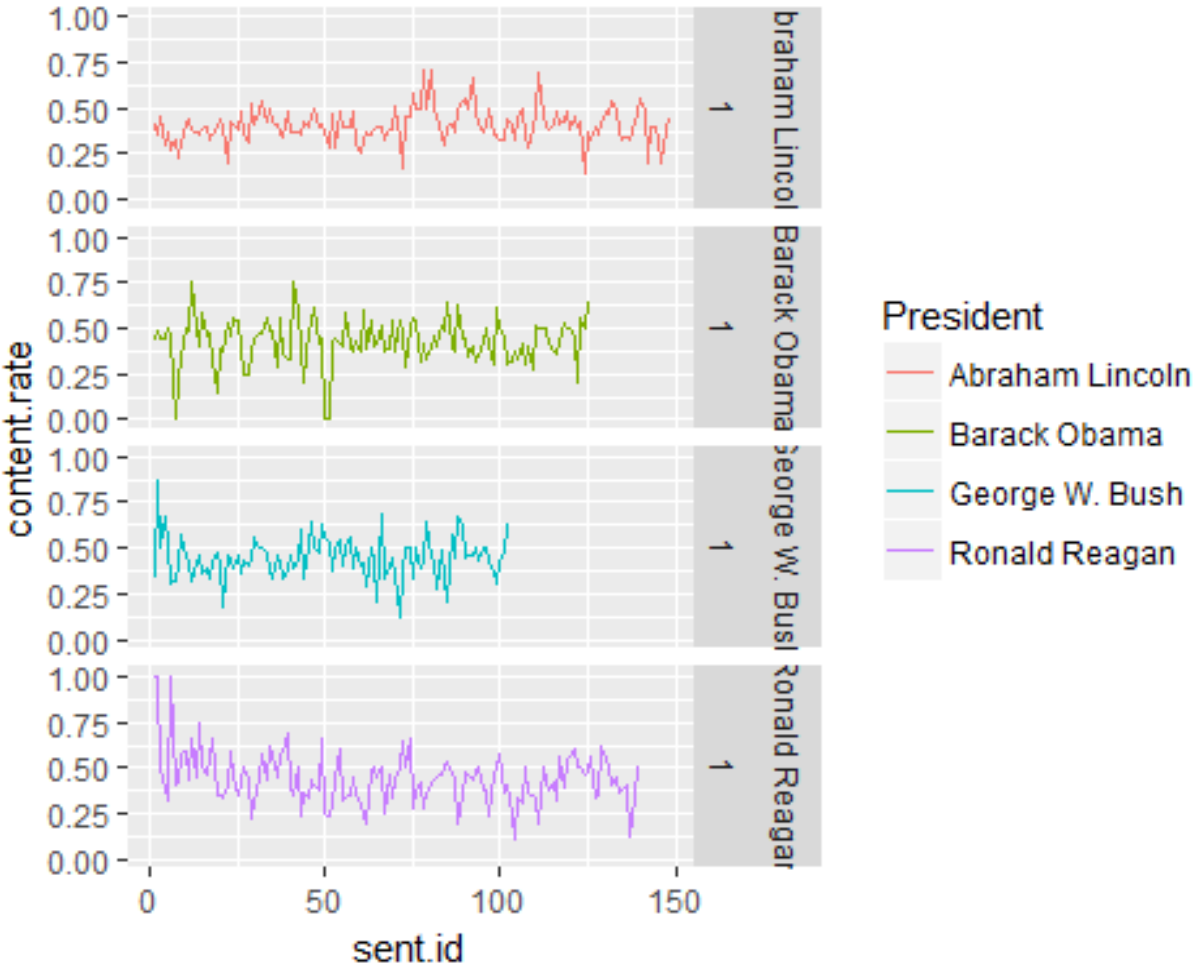
```
# Function for lexical density/content words of a sentence, using lexical
classification score.
lexical.rate <- function(s){
  t <- vector("numeric", length(s))
  for(i in 1:length(t)){
    t[i] <-  lexical_classification(s[i])[[4]]$ave.content.rate/100
  }
  return(t)
}
lexical.num <- function(s){
   t <- vector("numeric", length(s))
  for(i in 1:length(t)){
    t[i] <-  lexical_classification(s[i])[[4]]$n.content
  }
  return(t)
}

# Get the number and rate of content words for each sentences.
sentence.list$content.rate <- lexical.rate(sentence.list$sentences)
```

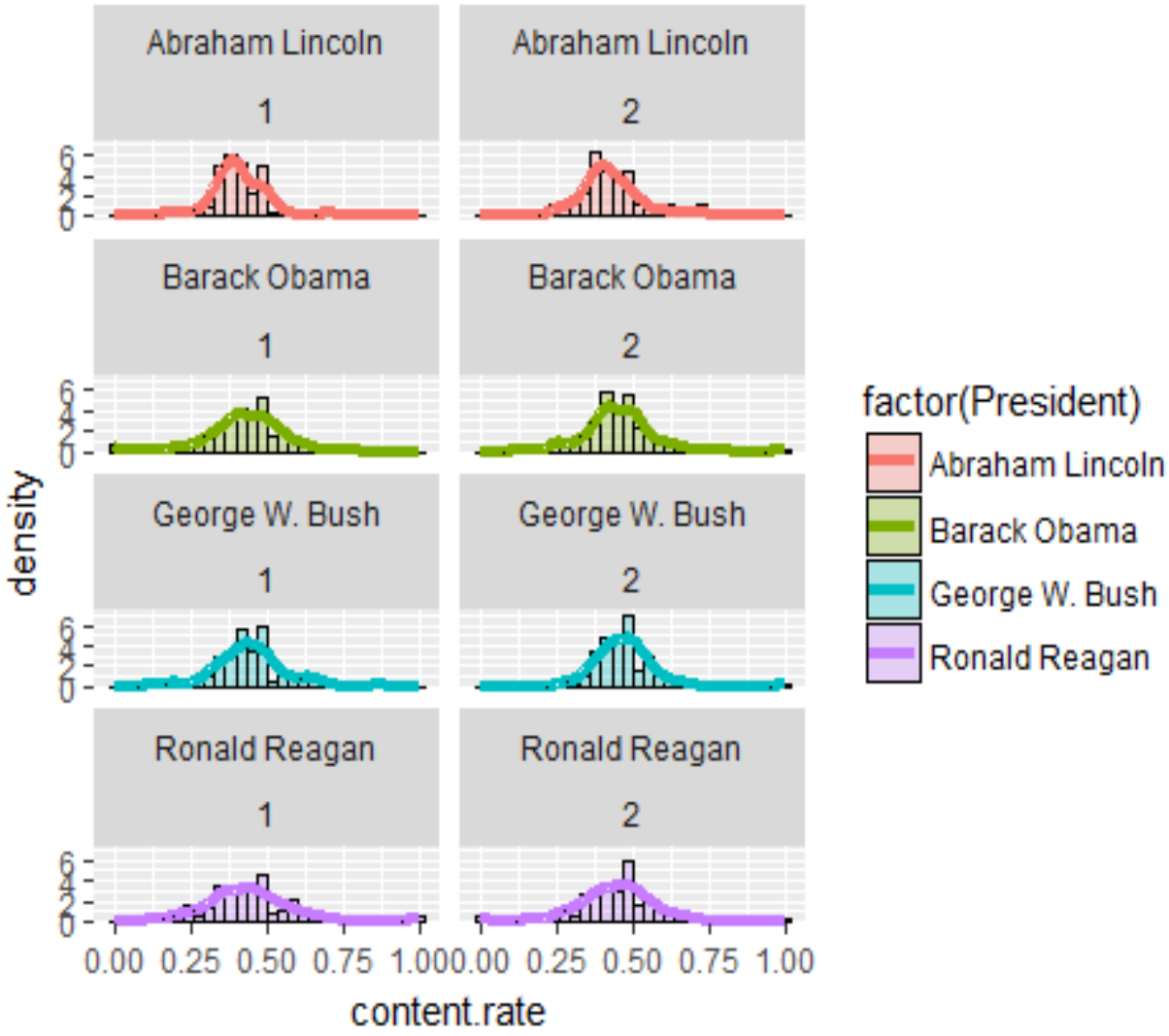
```
sentence.list$content.num <- lexical.num(sentence.list$sentences)
# Get the number of content words for each inauguration.
content.sum <- with(sentence.list, tapply(content.num, list(President, Term),
sum))
content.sum <- as.data.frame(t(as.matrix(content.sum)))
for(i in 1:nrow(speech.list)){
  nm <- speech.list$President[i]
  s <- speech.list$Term[i]
  speech.list$content.num[i] <- content.sum[nm][s,1]
}
speech.list$content.rate <- speech.list$content.num/speech.list$Words
```

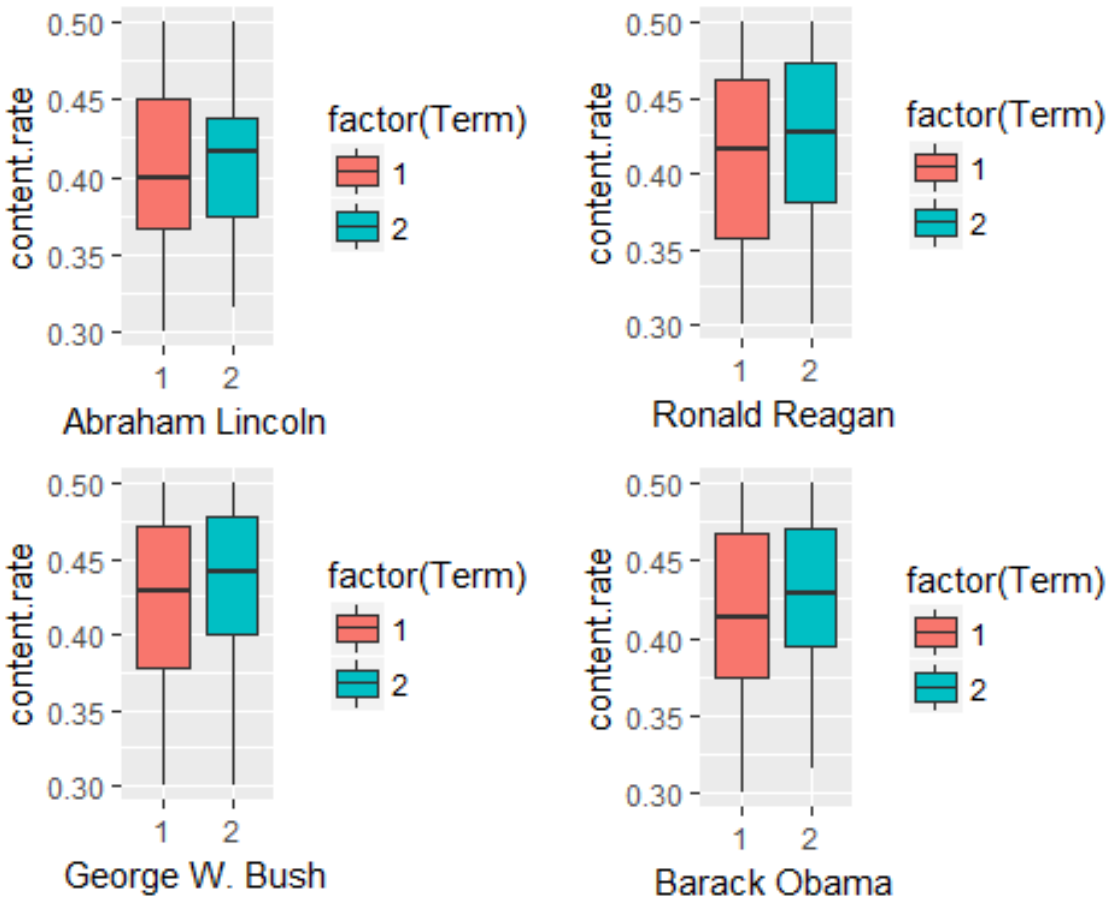
5. Visualization and analysis

I only choose four famous presidents for presentation. Generating all the figures may take 5 minutes.

```
# Generate the figures.
for(i in 1:nrow(speech.list)){
  png(paste("../output/",speech.list$President[i],"-
Term",speech.list$Term[i],".png", sep = ""), width = 500, height = 500 )

  nd <- sentence.list[sentence.list$President == speech.list$President[i] &
sentence.list$Term == speech.list$Term[i],]

  g1 <- ggplot(nd, aes(x = sent.id, y = content.rate,color = President))
+geom_line()+facet_grid(President+Term~.)

  g2 <- ggplot(nd, aes(x = content.rate)) +
  geom_histogram(aes(fill=factor(President),y=..density..),
alpha=0.3,colour='black')+
stat_density(geom='line',position='identity',size=1.5,
aes(colour=factor(President)))

  grid.arrange(g1, g2, ncol=1, nrow=2)

  dev.off()
}
```


1.00
0.75
0.50
0.25
0.00
content.rate
0
50
100
150
sent.id
braham Lincol
Barack Obama
eorge W. Busl
Ronald Reagar
1
President
Abraham Lincoln
Barack Obama
George W. Bush
Ronald Reagan


Abraham Lincoln
1
Abraham Lincoln
2
Barack Obama
1
Barack Obama
2
George W. Bush
1
George W. Bush
2
Ronald Reagan
1
Ronald Reagan
2
density
0
2
4
6
0.00
0.25
0.50
0.75
1.00
content.rate
factor(President)
Abraham Lincoln
Barack Obama
George W. Bush
Ronald Reagan



Let's take a look at the results.

The lexical density of an inauguration is well controlled around 44% (mean), which may be a factor for a good speech, and it is close to our daily communication.

Most writers put the most informative sentences in the first one third of the speech, and the skill of switch between short and long sentences is used perfectly, which can be told from the graphs.

In the last part of speech, there is always a sharp decrement in lexical density followed by a sharp increment.

Average densities of Democratic and Republican are almost the same, while others are lower.

47% 2nd term inaugurations have higher density than those in 1st term. However, in the last forty years, all presidents gave an inauguration with higher density when re-elected.

...

0. Linear regression

From the work above, I started to think that the inaugurations are not written by the presidents, and what we get from textmining is all about their writer teams, who had hundreds of meetings to discuss what to include and how to express.

So what did presidents do with inaugurations?

They delivered the speeches. So the health condition of a president is an influential factor of inaugurations. Because the length of an inauguration, content words density, sentence length and other factors can be influenced by president's energy. Every inauguration should be designed for that president, so I use the age of presidents as a factor of health condition and also response value, and factors of inaugurations as predictors to find if there is any relationship.

```
## lm0 <-
lm(age~words+cont.num+cont.rate+as.factor(party)+I(words/term)+I(sent.num^2)+
I(sent.num^3)+I(words/sent.num), data = data0)

##
## Call:
## lm(formula = age ~ words + cont.num + cont.rate + as.factor(party) +
##     I(words/term) + I(sent.num^2) + I(sent.num^3) + I(words/sent.num),
##     data = data0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.3732 -3.0043 -0.2581  1.3460 13.7801
##
## Coefficients:
##                                              Estimate Std. Error t value
## (Intercept)                                 1.396e+02  6.135e+01   2.275
## words                                      -3.131e-02  2.273e-02  -1.377
## cont.num                                    6.365e-02  5.166e-02   1.232
## cont.rate                                  -2.221e+02  1.407e+02  -1.578
## as.factor(party)Democratic-Republican Party 5.073e+00  3.115e+00   1.629
## as.factor(party)Fedralist                   6.773e+00  7.864e+00   0.861
## as.factor(party)Republican                  5.394e+00  2.042e+00   2.641
## as.factor(party)Whig                        6.230e+01  1.907e+01   3.266
## I(words/term)                              -3.968e-03  1.688e-03  -2.351
## I(sent.num^2)                               2.402e-03  1.331e-03   1.805
## I(sent.num^3)                              -9.143e-06  4.797e-06  -1.906
## I(words/sent.num)                           5.466e-01  5.151e-01   1.061
##                                            Pr(>|t|)
## (Intercept)                                 0.02951 *
## words                                       0.17775
## cont.num                                    0.22658
## cont.rate                                   0.12402
## as.factor(party)Democratic-Republican Party 0.11287
## as.factor(party)Fedralist                   0.39527
```

```
## as.factor(party)Republican                    0.01253 *
## as.factor(party)Whig                          0.00255 **
## I(words/term)                                 0.02487 *
## I(sent.num^2)                                 0.08020 .
## I(sent.num^3)                                 0.06541 .
## I(words/sent.num)                             0.29630
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.478 on 33 degrees of freedom
## Multiple R-squared:  0.4784, Adjusted R-squared:  0.3045
## F-statistic: 2.751 on 11 and 33 DF,  p-value: 0.01206

## RMSE is 4.691427
```

The RMSE is kind of okay because we can say someone is around 50 or 55, but the R square value is not good enough for a social data analysis.