IST 707 Homework 4

Due Date: 5/5/2021

Introduction

The federalist papers are a series of 85 historical documents written by Alexander Hamilton, James Madison, and John Jay - who at the time of the writing were trying to push ratification of the US constitution. The papers were published anonymously in the newspapers in the city of New York in the late 1700's. It is known who the true author of 74 of the documents is, but 11 of the documents remain uknown because their identity was never revealed.

The mystery of the federalist papers has yet to be solved. Statititians for years have attempted to figure it out, and even though there have been some convincing results, the topic is still controversial. The following report will explore a dataset consisting of word frequencies from the federalist paper collection. More specifically, the goal is to assign authors using clustering techniques.

Load packages as needed

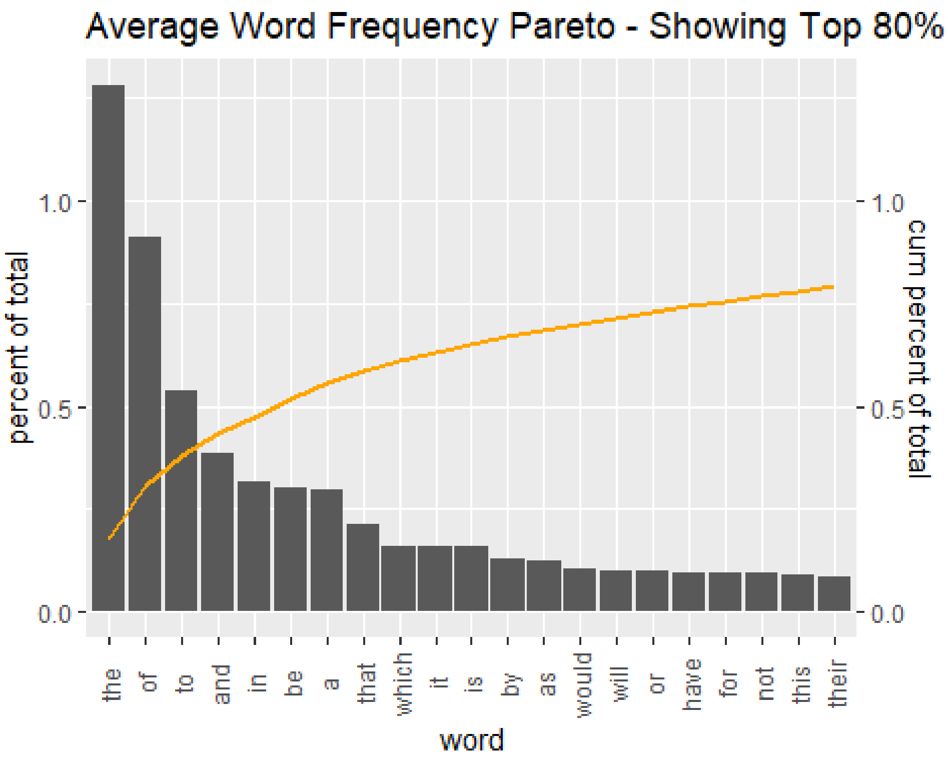
```
require(readr)

require(sqldf)

require(dplyr)

require(ggplot2)

require(reshape2)

require(matrixStats)

require(factoextra)

require(gridExtra)
```

Load the federalist papers dataset

```
fedpapers <- read_csv("fedPapers85.csv")

##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   author = col_character(),
##   filename = col_character()
## )
## i Use `spec()` for the full column specifications.
```

Data Exploration

```r
# word frequencies pareto distribution

pareto_words      <- colnames(fedpapers) [3: ncol(fedpapers)]
pareto_freq_all   <- colMeans(fedpapers[, 3: ncol(fedpapers)])
pareto_freq_ham    <- colMeans(fedpapers[which(fedpapers$author == 'Hamilton'), 3: ncol(fedpapers)])
pareto_freq_jay    <- colMeans(fedpapers[which(fedpapers$author == 'Jay'), 3: ncol(fedpapers)])
pareto_freq_mad    <- colMeans(fedpapers[which(fedpapers$author == 'Madison'), 3: ncol(fedpapers)])

pareto_table      <- data.frame(cbind(
  pareto_words,
  pareto_freq_all,
  pareto_freq_ham,
  pareto_freq_jay,
  pareto_freq_mad))

pareto_table[, c(2:5)] <- pareto_table[, c(2:5)] %>% mutate_all(as.numeric)

pareto_table <- sqldf('
select
  pareto_words,
  pareto_freq_all,
  pctTotal,
  sum(pctTotal) over (order by pareto_freq_all desc) as cumpctTotal,
  pareto_freq_ham,
  pareto_freq_jay,
  pareto_freq_mad
from (
  select pareto_words,
         pareto_freq_all,
         pareto_freq_all / (select sum(pareto_freq_all) as temp from pareto_table) as pctTotal,
         pareto_freq_ham,
         pareto_freq_jay,
         pareto_freq_mad
  from pareto_table)
sub')

ggplot(pareto_table[which(pareto_table$cumpctTotal < 0.8),])  +
  geom_col(aes(
    x = reorder(pareto_words, -pareto_freq_all),
    y = pareto_freq_all)) +
  geom_line(aes(
    x = reorder(pareto_words, - pareto_freq_all),
    y = cumpctTotal), size = 1, color = 'orange', group = 1) +
  scale_y_continuous(
    sec.axis = sec_axis(~., name = "cum percent of total")) +
  xlab("word") +
  ylab("percent of total") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5)) +
  ggtitle('Average Word Frequency Pareto - Showing Top 80%')
```

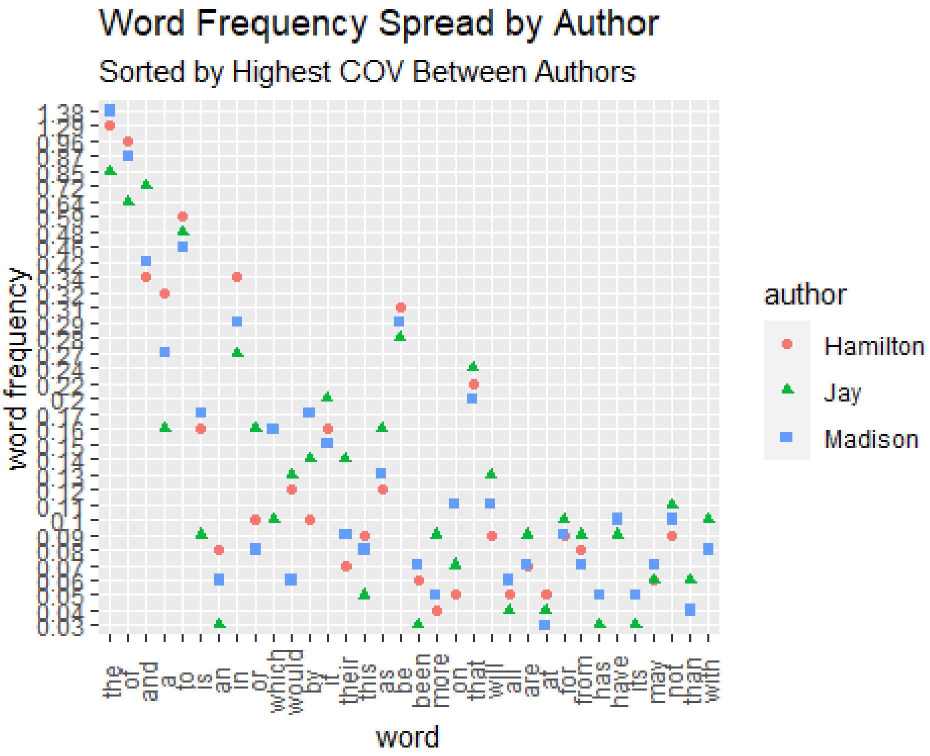


Observations

- articles, prepositions, conjunctions make up the large majority of words

- 21 words make up 80% of the cumulative average frequencies of all words

```
# biggest differences in word frequences - hamilton

# hamilton dataframe

word_differences_ham <- data.frame(cbind(
  'word' = pareto_table[, 1],
  'freq_ham' = pareto_table[, 5],
  'vs_jay' = pareto_table[, 5] / pareto_table[, 6],
  'freq_jay' = pareto_table[, 6],
  'score_jay' = pareto_table[, 5] * (pareto_table[, 5] / pareto_table[, 6]),
  'vs_mad' = pareto_table[, 5] / pareto_table[, 7],
  'freq_mad' = pareto_table[, 7],
  'score_mad' = pareto_table[, 5] * (pareto_table[, 5] / pareto_table[, 7])
))

word_differences_ham[, 2:8] <-
  word_differences_ham[, 2:8] %>%
  mutate_all(as.numeric) %>%
  mutate_all(round, 2)

word_differences_ham$COV <- round(as.numeric(
  rowSds(as.matrix(word_differences_ham[, c(2, 4, 7)])) /
  rowMeans(word_differences_ham[, c(2, 4, 7)])), 2)

word_differences_ham$COVscore <- round(as.numeric(
  word_differences_ham$freq_ham *
  word_differences_ham$COV), 2)

# hamilton top 10

head(word_differences_ham[order(-word_differences_ham[,10]), ], 10)

##     word freq_ham vs_jay freq_jay score_jay vs_mad freq_mad score_mad  COV
## 1    the     1.29   1.51     0.85      1.95   0.94     1.38      1.21 0.24
## 2     of     0.96   1.50     0.64      1.43   1.10     0.87      1.05 0.20
## 4    and     0.34   0.47     0.72      0.16   0.81     0.42      0.27 0.41
## 7      a     0.32   1.98     0.16      0.62   1.17     0.27      0.37 0.33
## 44  upon     0.05  26.29     0.00      1.24  21.51     0.00      1.02 1.73
## 3     to     0.59   1.22     0.48      0.72   1.29     0.46      0.76 0.14
## 11    is     0.16   1.70     0.09      0.27   0.93     0.17      0.15 0.31
## 5     in     0.34   1.27     0.27      0.44   1.20     0.29      0.41 0.12
## 9  which     0.16   1.63     0.10      0.26   0.98     0.16      0.16 0.25
## 14 would     0.12   0.98     0.13      0.12   2.02     0.06      0.25 0.37
##    COVscore
## 1      0.31
## 2      0.19
## 4      0.14
## 7      0.11
## 44     0.09
## 3      0.08
## 11     0.05
## 5      0.04
## 9      0.04
## 14     0.04

# comparison chart

comparison_chart <- as.data.frame(rbind(

as.data.frame(cbind(
  'word' = word_differences_ham[, 1],
  'word_frequency' = word_differences_ham[, 2],
  'author' = replicate(70, 'Hamilton'))),

as.data.frame(cbind(
  'word' = word_differences_ham[, 1],
  'word_frequency' = word_differences_ham[, 4],
  'author' = replicate(70, 'Jay'))),

as.data.frame(cbind(
  'word' = word_differences_ham[, 1],
  'word_frequency' = word_differences_ham[, 7],
  'author' = replicate(70, 'Madison')))

))

comparison_chart <- left_join(
  comparison_chart,
  pareto_table[, c(1, 2, 4)],
  by = c('word' = 'pareto_words'))

comparison_chart <- left_join(comparison_chart, word_differences_ham[, c(1, 10)], by = c('word' =
'word'))

ggplot(comparison_chart[which(comparison_chart$cumpctTotal < .90),], aes(
  x = reorder(word, -COVscore),
  y = word_frequency,
  color = author,
  shape = author)) +
  geom_point() +
  xlab("word") +
  ylab("word frequency") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5)) +
  ggtitle('Word Frequency Spread by Author',
  subtitle = 'Sorted by Highest COV Between Authors')
```



Observations

- Hamilton uses the word ‘upon’ over 20x more frequently than both the other authors, this could be an identifer

- Hamilton and Madison both use ‘your’ 1/3 of the amount than jay, does jay more often speak in the second person?

- Hamilton uses the word ‘by’ almost half as frequently as madison and it accounts for 10% of his total word frequency

- Hamilton uses the word ‘the’, ‘of’ 1.5x as much as jay and the word ‘and’ 0.5x as much as jay, but are similar to madison

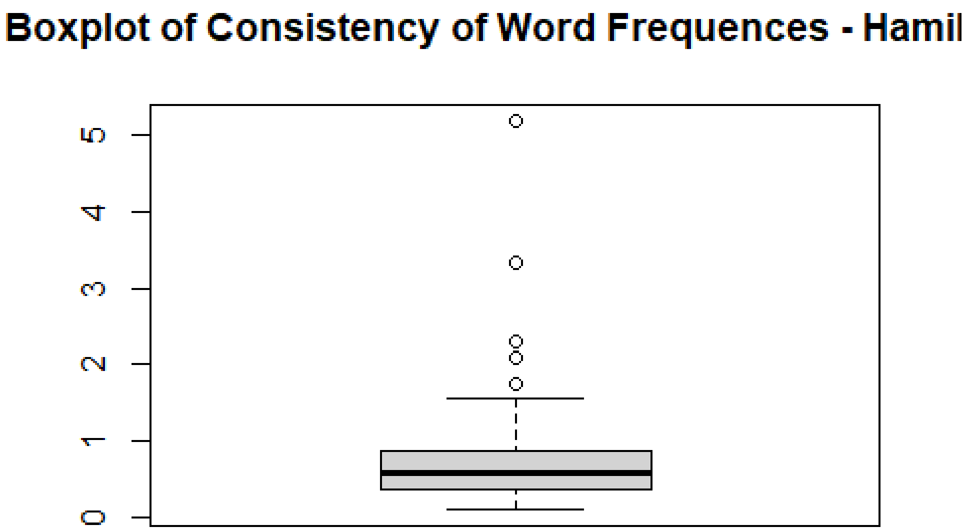
- For the most part hamilton and madison are closer on words and jay tends to be further away

```
# deep dive on madison vs hamilton

# what word frequencies are the most different?

word_matrix_ham_mad <- as.data.frame(cbind(
  'word' = pareto_table[, 1],
  'ham_freq' = pareto_table[, 5] %>% as.numeric() %>% round(2),
  'mad_freq' = pareto_table[, 7]%>% as.numeric() %>% round(2),
  'absdiff' = abs(pareto_table[, 5] - pareto_table[, 7]) %>% as.numeric() %>% round(2)))

head(sqldf('select * from word_matrix_ham_mad order by absdiff desc'), 10)
```

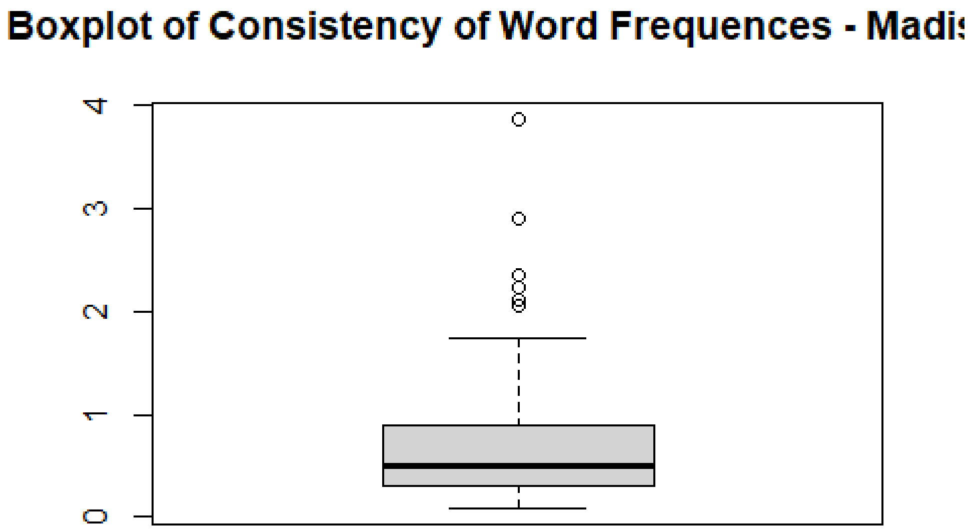
```
##     word ham_freq mad_freq absdiff
## 1     to     0.59     0.46    0.13
## 2    the     1.29     1.38    0.09
## 3     of     0.96     0.87    0.09
## 4    and     0.34     0.42    0.08
## 5     in     0.34     0.29    0.06
## 6     by      0.1     0.17    0.06
## 7  would     0.12     0.06    0.06
## 8     on     0.05     0.11    0.06
## 9      a     0.32     0.27    0.05
## 10  upon     0.05        0    0.05
```

```
word_matrix_ham <- fedpapers[which(fedpapers$author == 'Hamilton'), ]
word_matrix_mad <- fedpapers[which(fedpapers$author == 'Madison'), ]

# coefficient of variance to measure consistency

consistency_ham <- as.data.frame(cbind(
  'word' = colnames(fedpapers) [3: ncol(fedpapers)],
  'ham_sd' = colSds(as.matrix(word_matrix_ham[, 3: ncol(word_matrix_ham)])),
  'ham_avg' = colMeans(word_matrix_ham[, 3: ncol(word_matrix_ham)])))
consistency_ham[, c(2, 3)] <- consistency_ham[, c(2, 3)] %>% mutate_all(as.numeric)
consistency_ham$ham_cov <- as.numeric(consistency_ham$ham_sd / consistency_ham$ham_avg) %>%
round(2)
```

```
consistency_mad <- as.data.frame(cbind(
  'word' = colnames(fedpapers) [3: ncol(fedpapers)],
  'mad_sd' = colSds(as.matrix(word_matrix_mad[, 3: ncol(word_matrix_mad)])),
  'mad_avg' = colMeans(word_matrix_mad[, 3: ncol(word_matrix_mad)])))
consistency_mad[, c(2, 3)] <- consistency_mad[, c(2, 3)] %>% mutate_all(as.numeric)
consistency_mad$mad_cov <- as.numeric(consistency_mad$mad_sd / consistency_mad$mad_avg) %>%
round(2)

boxplot(consistency_ham$ham_cov,
  main = 'Boxplot of Consistency of Word Frequences - Hamilton')
```

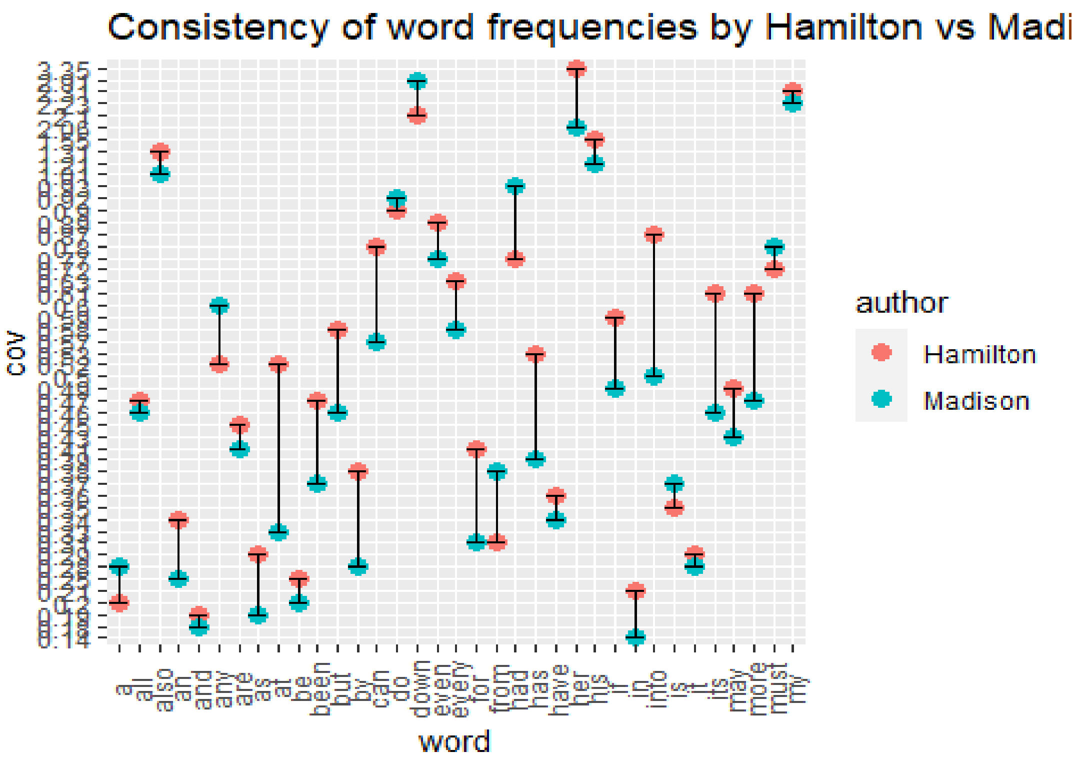


```
boxplot(consistency_mad$mad_cov,
  main = 'Boxplot of Consistency of Word Frequences - Madison')
```

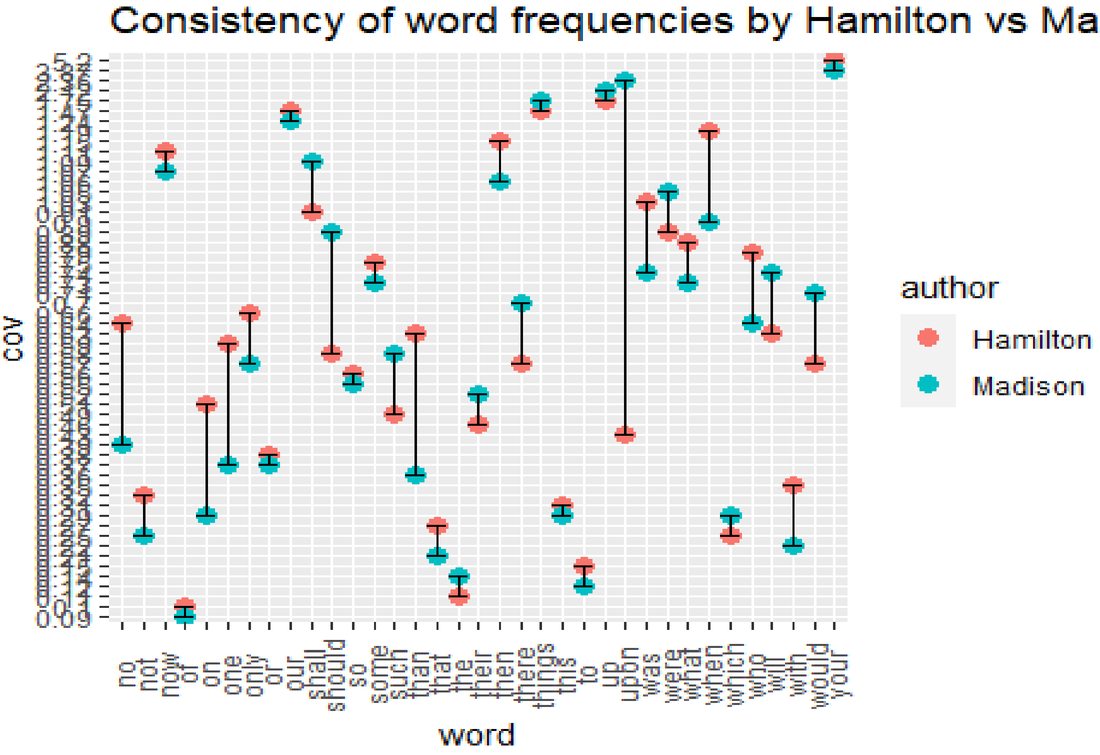


```
# there are some words that do not appear to be consistent
# what are these words and are they the same or different for each author

consistency_ham_mad <- left_join(consistency_ham, consistency_mad, by = c('word' = 'word'))
consistency_ham_mad$absdiff <- abs(consistency_ham_mad$ham_cov -
consistency_ham_mad$mad_cov)

head(consistency_ham_mad[order(-consistency_ham_mad[, 8]),], 10)

##      word      ham_sd     ham_avg ham_cov      mad_sd      mad_avg mad_cov
## 60   upon 0.020373012 0.047313725    0.43 0.005185144 0.0022000000    2.36
## 70   your 0.010807115 0.002078431    5.20 0.008778762 0.0022666667    3.87
## 24    her 0.023636761 0.007058824    3.35 0.016076602 0.0078000000    2.06
## 16   down 0.004154468 0.001980392    2.10 0.002711527 0.0009333333    2.91
## 59     up 0.008015622 0.004568627    1.75 0.002685056 0.0012666667    2.12
## 28   into 0.018627389 0.021313725    0.87 0.013442293 0.0268666667    0.50
## 25    his 0.049485692 0.031921569    1.55 0.022796825 0.0188666667    1.21
## 46 should 0.017014204 0.029392157    0.58 0.021096603 0.0237333333    0.89
## 3    also 0.006246003 0.004784314    1.31 0.011132107 0.0110666667    1.01
## 61    was 0.021328927 0.020607843    1.03 0.018933970 0.0257333333    0.74
##    absdiff
## 60    1.93
## 70    1.33
## 24    1.29
## 16    0.81
## 59    0.37
## 28    0.37
## 25    0.34
## 46    0.31
## 3     0.30
## 61    0.29

consistency_ham_mad <- rbind(

  data.frame(cbind(
    'word' = consistency_ham[, 1],
    'cov' = consistency_ham[, 4],
    'author' = 'Hamilton')),

  data.frame(cbind(
    'word' = consistency_mad[, 1],
    'cov' = consistency_mad[, 4],
    'author' = 'Madison'))

)

graphminmax <- sqldf('select word, min(cov) as min, max(cov) as max from consistency_ham_mad
group by word')
consistency_ham_mad <- sqldf('select * from consistency_ham_mad order by word, author asc')
consistency_ham_mad <- left_join(consistency_ham_mad, graphminmax, by = c('word' = 'word'))
```

```
ggplot(consistency_ham_mad[1:70,], aes(
 x = word,
 y = cov)) +
 geom_point(aes(
  color = author),
  size = 3) +
 geom_errorbar(aes(
 ymin = min,
 ymax = max)) +
 ggtitle('Consistency of word frequencies by Hamilton vs Madison (Words 1-35)') +
 theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```



```
ggplot(consistency_ham_mad[71:140,], aes(
 x = word,
 y = cov)) +
 geom_point(aes(
  color = author),
  size = 3) +
 geom_errorbar(aes(
 ymin = min,
 ymax = max)) +
 ggtitle('Consistency of word frequencies by Hamilton vs Madison (Words 36-70)') +
 theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```



```
# view the word upon in the fed papers data for Hamilton
t(fedpapers[which(fedpapers$author == 'Hamilton'), c(1, which(colnames(fedpapers) == 'upon'))])
```

```
##        [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
## author "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton"
## upon   "0.056"    "0.035"    "0.041"    "0.030"    "0.048"    "0.043"
##        [,7]       [,8]       [,9]       [,10]      [,11]      [,12]
## author "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton"
## upon   "0.045"    "0.044"    "0.050"    "0.057"    "0.052"    "0.007"
##        [,13]      [,14]      [,15]      [,16]      [,17]      [,18]
## author "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton"
## upon   "0.038"    "0.040"    "0.027"    "0.068"    "0.098"    "0.102"
##        [,19]      [,20]      [,21]      [,22]      [,23]      [,24]
## author "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton"
## upon   "0.020"    "0.080"    "0.068"    "0.059"    "0.033"    "0.023"
##        [,25]      [,26]      [,27]      [,28]      [,29]      [,30]
## author "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton"
## upon   "0.028"    "0.053"    "0.030"    "0.074"    "0.072"    "0.053"
##        [,31]      [,32]      [,33]      [,34]      [,35]      [,36]
## author "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton"
## upon   "0.020"    "0.061"    "0.069"    "0.024"    "0.026"    "0.036"
##        [,37]      [,38]      [,39]      [,40]      [,41]      [,42]
## author "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton"
## upon   "0.083"    "0.044"    "0.039"    "0.062"    "0.068"    "0.039"
##        [,43]      [,44]      [,45]      [,46]      [,47]      [,48]
## author "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton" "Hamilton"
## upon   "0.028"    "0.021"    "0.036"    "0.050"    "0.038"    "0.052"
##        [,49]      [,50]      [,51]
## author "Hamilton" "Hamilton" "Hamilton"
## upon   "0.047"    "0.067"    "0.029"
```

```
# view the word upon in the fed papers data for Madison
t(fedpapers[which(fedpapers$author == 'Madison'), c(1, which(colnames(fedpapers) == 'upon'))])
```

```
##        [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## author "Madison" "Madison" "Madison" "Madison" "Madison" "Madison" "Madison"
## upon   "0.000"   "0.000"   "0.005"   "0.018"   "0.000"   "0.000"   "0.000"
##        [,8]      [,9]      [,10]     [,11]     [,12]     [,13]     [,14]
## author "Madison" "Madison" "Madison" "Madison" "Madison" "Madison" "Madison"
## upon   "0.010"   "0.000"   "0.000"   "0.000"   "0.000"   "0.000"   "0.000"
##        [,15]
## author "Madison"
## upon   "0.000"
```

Observations

- there are some words for both authors that did not have consistent word frequencies

- Hamilton uses the word ‘upon’ very frequently and consistently, madison uses it very infrequently and inconsistently

- Hamilton is more consistent with use of the words ‘at’, ‘no’, ‘into’

- Madison is more consistent with the use of the words ‘such, ’there’

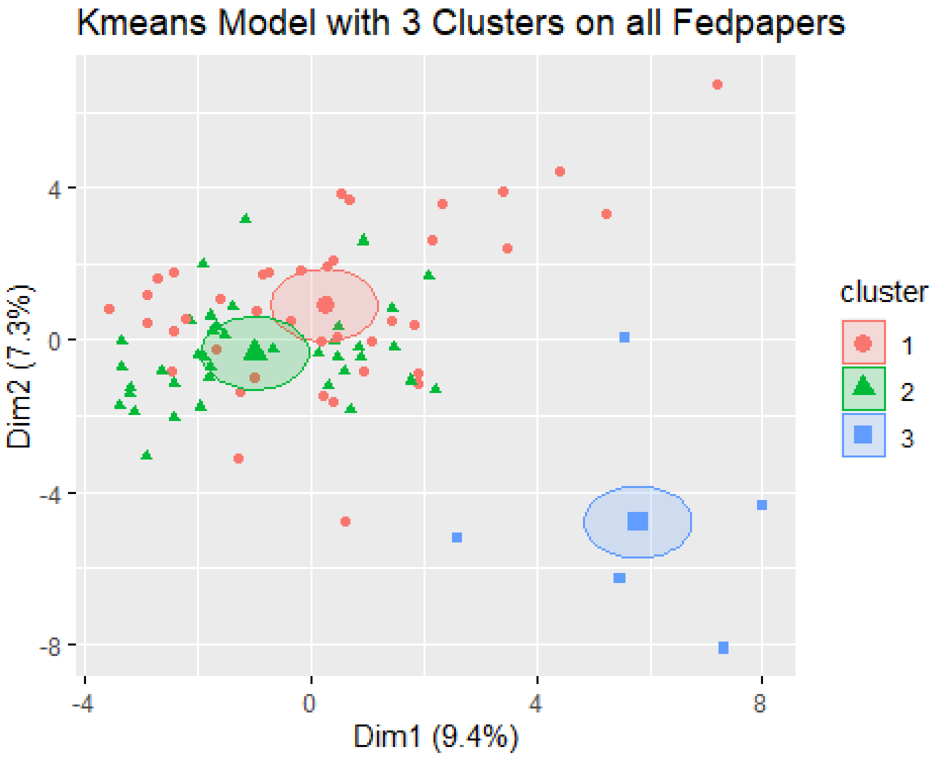
- In general, the authors are consistent with word frequencies from paper to paper.

Data Analysis

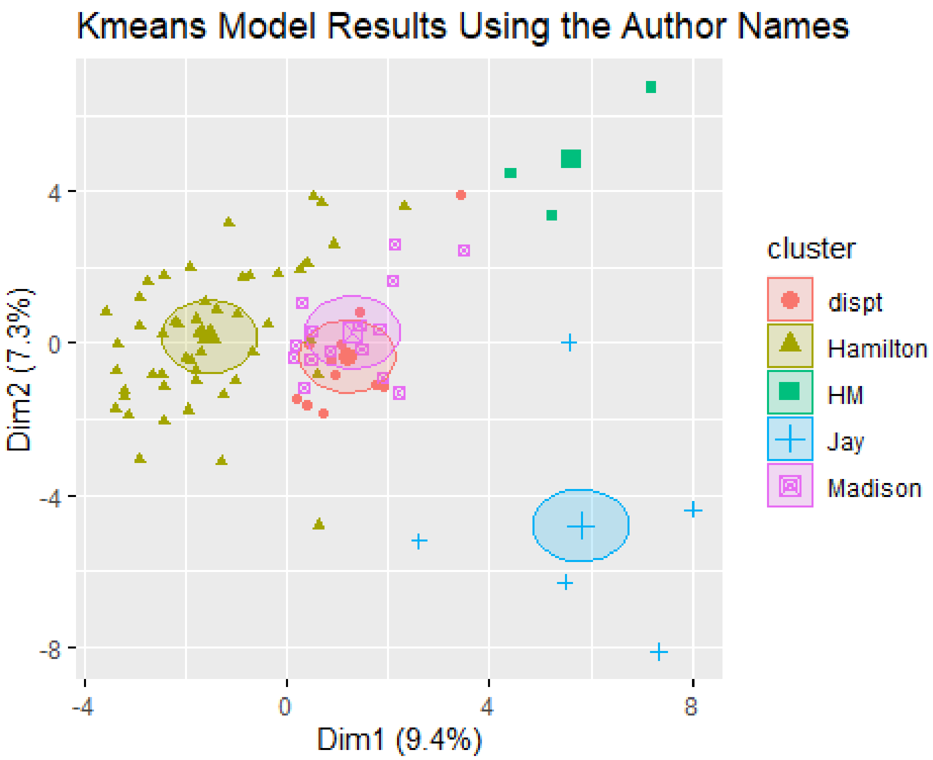
Starting off with clustering using all of the data, including the disputed papers, to observe the results - knowing that it may not be ideal right away.

```
# kmeans clustering

# compute k-means
clustfed <- fedpapers[, c(1, 3:ncol(fedpapers))]
set.seed(175)
kmfed <- kmeans(clustfed[, 2:ncol(clustfed)], 3, nstart = 25)

# show cluster means
kmfedmeans <- aggregate(clustfed[, 2:ncol(clustfed)], by=list(cluster=kmfed$cluster), mean)
kmfedmeans <- t(kmfedmeans) %>% as.data.frame() %>% round(2)
colnames(kmfedmeans) <- c('C1', 'C2', 'C3')
kmfedmeans <- kmfedmeans[-1, ]
clustfed$cluster <- kmfed$cluster

# plot the results
```

```
#plot one is partitioned by the 3 clusters from the model
fviz_cluster(kmfed, data = clustfed[, 2:ncol(clustfed)], geom = c("point"),ellipse.type = "euclid") +
  ggtitle('Kmeans Model with 3 Clusters on all Fedpapers')
```



```
kmfed$cluster <- clustfed$author

#plot two replaces the model clusters with the actual author names
fviz_cluster(kmfed, data = clustfed[, 2:ncol(clustfed)], geom = c("point"),ellipse.type = "euclid") +
  ggtitle('Kmeans Model Results Using the Author Names')

## Too few points to calculate an ellipse
```



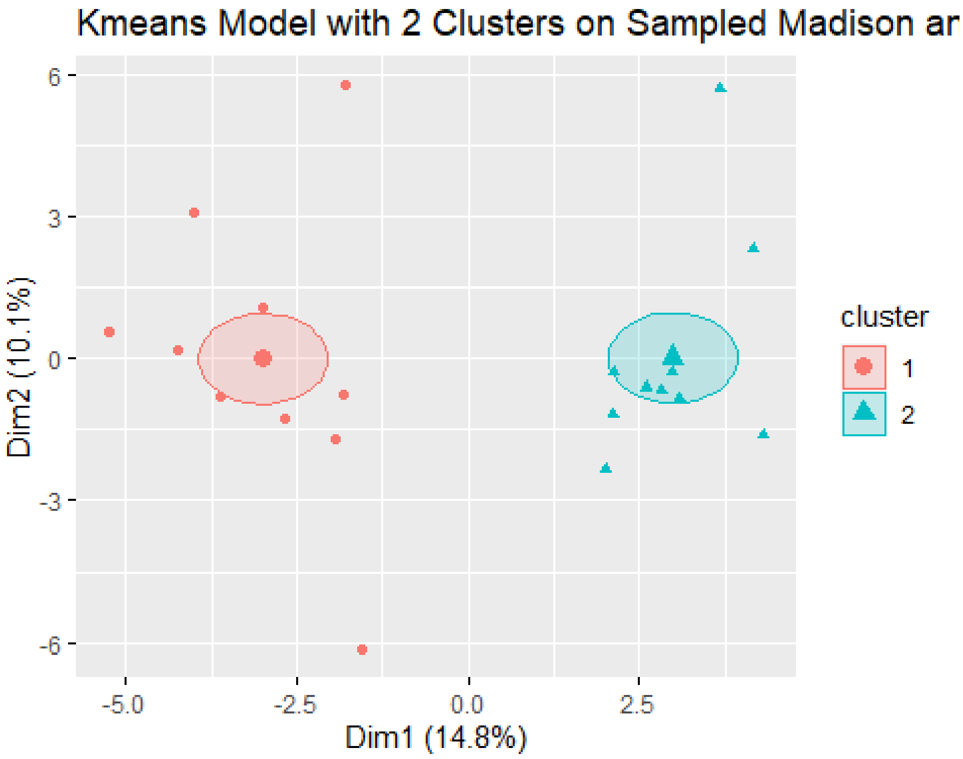
The model had a difficult time distinguishing between Madison and Hamilton. It was easily able to figure out Jay, however. But based on comparing the two graphs, it is interesting to note that the authors are distinguishable according to the same x and y coordinates.

A subsequent iteration of the kmeans algorithm will be performed - this time without Jay - because Jay is noise. The 3 duo and 11 uknown papers will also be removed from the data. A sample of 10 papers from each Madison and Hamilton will be used.

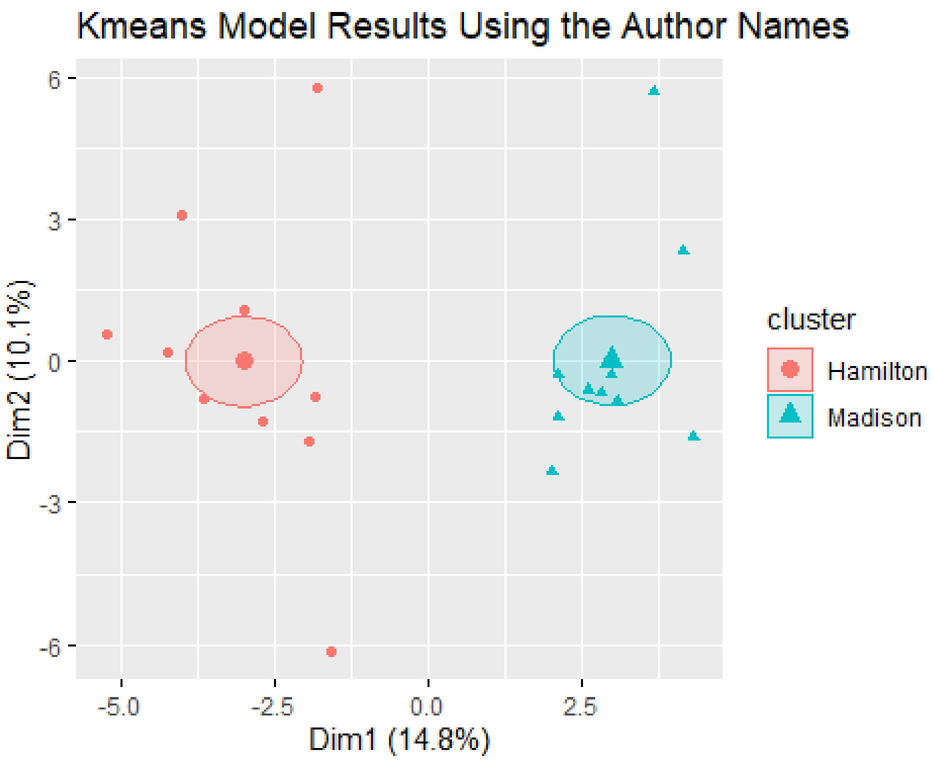
For Hamilton, a sample of 10 should be an adequate representation of the population, given the consistency of word frequencies from paper to paper as seen earlier. For Madison, a sample of 10 should also be adequate given that it is 2/3 of all of Madison’s papers.

```
# kmeans clustering - sampled

# compute k-means
tempfed <- sqldf('select author, row_number() over () as rownum from fedpapers')
hamrows <- tempfed[which(tempfed$author == 'Hamilton'), 2]
madrows <- tempfed[which(tempfed$author == 'Madison'), 2]

set.seed(190)
hamsample <- sample(hamrows, 10, replace = FALSE)
madsample <- sample(madrows, 10, replace = FALSE)

clustfed <- rbind(
  fedpapers[hamsample, c(1, 3:ncol(fedpapers))],
  fedpapers[madsample, c(1, 3:ncol(fedpapers))])
```

```
set.seed(175)
kmfed <- kmeans(clustfed[, 2:ncol(clustfed)], 2, nstart = 25)

# show cluster means
kmfedmeans <- aggregate(clustfed[, 2:ncol(clustfed)], by=list(cluster=kmfed$cluster), mean)
kmfedmeans <- t(kmfedmeans) %>% as.data.frame() %>% round(2)
colnames(kmfedmeans) <- c('C1', 'C2')
kmfedmeans <- kmfedmeans[-1, ]
clustfed$cluster <- kmfed$cluster

# plot the results

#plot one is partitioned by the 3 clusters from the model
fviz_cluster(kmfed, data = clustfed[, 2:ncol(clustfed)], geom = c("point"),ellipse.type = "euclid") +
  ggtitle('Kmeans Model with 2 Clusters on Sampled Madison and Hamilton')
```

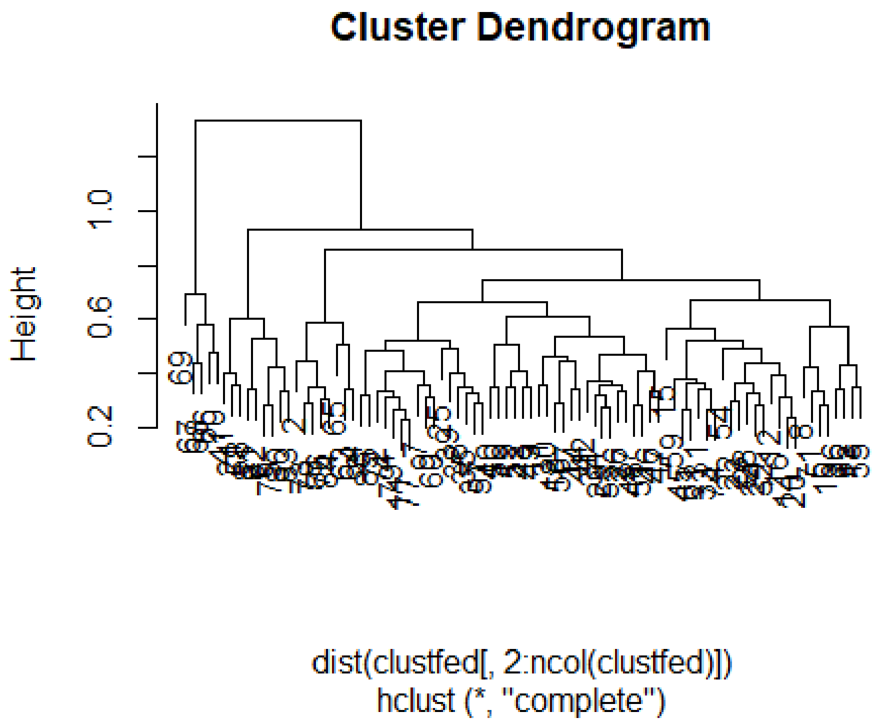


```
kmfed$cluster <- clustfed$author

#plot two replaces the model clusters with the actual author names
fviz_cluster(kmfed, data = clustfed[, 2:ncol(clustfed)], geom = c("point"),ellipse.type = "euclid") +
  ggtitle('Kmeans Model Results Using the Author Names')
```



The results here show that the model was able to detect the author, that is, whether it was Hamliton or Madison. This is because the clusters align with the actual class labels when comparing the two charts. Next, I will use an HAC clustering algorithm.

```
# compute HAC

clustfed <- fedpapers[, c(1, 3:ncol(fedpapers))]
hacfed <- hclust(dist(clustfed[, 2:ncol(clustfed)]))
plot(hacfed)
```



```
# try a cut at 3
hacfedcut <- cutree(hacfed, 3)
table(hacfedcut, clustfed$author)

##
## hacfedcut dispt Hamilton HM Jay Madison
##         1    10       48  3   0      10
##         2     1        3  0   0       5
##         3     0        0  0   5       0

# try a cut at 4
hacfedcut <- cutree(hacfed, 4)
table(hacfedcut, clustfed$author)

##
## hacfedcut dispt Hamilton HM Jay Madison
##         1     9       48  0   0       6
##         2     1        0  3   0       4
##         3     1        3  0   0       5
##         4     0        0  0   5       0

# try a cut at 5
hacfedcut <- cutree(hacfed, 5)
table(hacfedcut, clustfed$author)

##
## hacfedcut dispt Hamilton HM Jay Madison
##         1     6       28  0   0       4
##       2    1       0  3   0       4
##       3    3      20  0   0       2
##       4    1       3  0   0       5
##       5    0       0  0   5       0
```

In the HAC algorithm, it is interesting to note that for 3 clusters, all of disputed, Hamilton, and Madisons papers fall into cluster 1. For 4 clusters, the disputed and Hamilton papers virtually remain the same, while Madison splits out into several other clusters. For 5 clusters, Hamilton starts to get split out. My interpretation of these results is that there are some subtle differences between Hamilton and Madison that the model was able to pick up on.

Conclusion

In summary, from reviewing the three models, there is still conflicting evidence. The first model shows that the disputed papers are most similar to Madison. For example. Yet all of the disputed papers have the word ‘upon’ in them. The word ‘upon’ was virtually only used by Hamilton and was hardly ever used by Madison. The third model does not provide any new information that can be used for a decision.

I am not able to say which authors the unknown papers belong to with high confidence. My gut feel, however, is that they belong to Hamilton, given that the word upon is used frequently in the disputed papers.

Keeping in mind that this is a simplified dataset, there may be an opportunity to look at a more comprehensive dataset and see if it produces better results. It would be interesting to consider all of the words, and not just a subset of the words from the papers. There may be identifying words that are being missed out on because they are missing from the data.