HW 4 Federalist Papers

Dan Burke

11/2/2021

Data Preparation

```
#load needed libraries
library(stats)
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.1.1
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(ggfortify)
```

```
## Warning: package 'ggfortify' was built under R version 4.1.1
```

```
#Data Preparation

fedpapers <- read.csv("C:\\Users\\danbu\\Desktop\\Applied Machine Learning\\Week 4\\HW4\\fedPapers85.csv

#check for Complete Cases
sum(!complete.cases(fedpapers))
```

```
## [1] 0
```

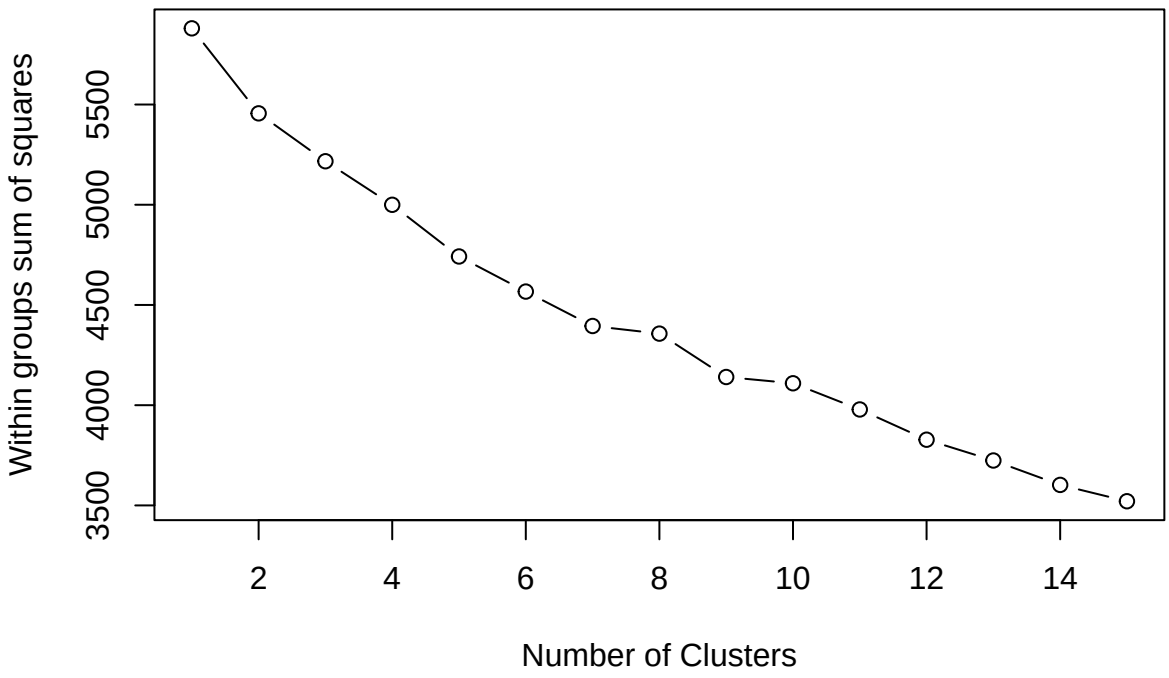
```
#Find Unique Values for Authors
authors <- unique(fedpapers$author)
authors
```

```
## [1] "dispt"    "Hamilton" "HM"       "Jay"      "Madison"
```

```
#drop the file name column
fedpapers <- fedpapers[, !names(fedpapers) %in% c("filename")]

knownpapers <- fedpapers[fedpapers$author == "Hamilton" | fedpapers$author == "Madison" | fedpapers$autl
unknownpapers <- fedpapers[which(fedpapers$author == "HM" | fedpapers$author == "dispt"),]
```

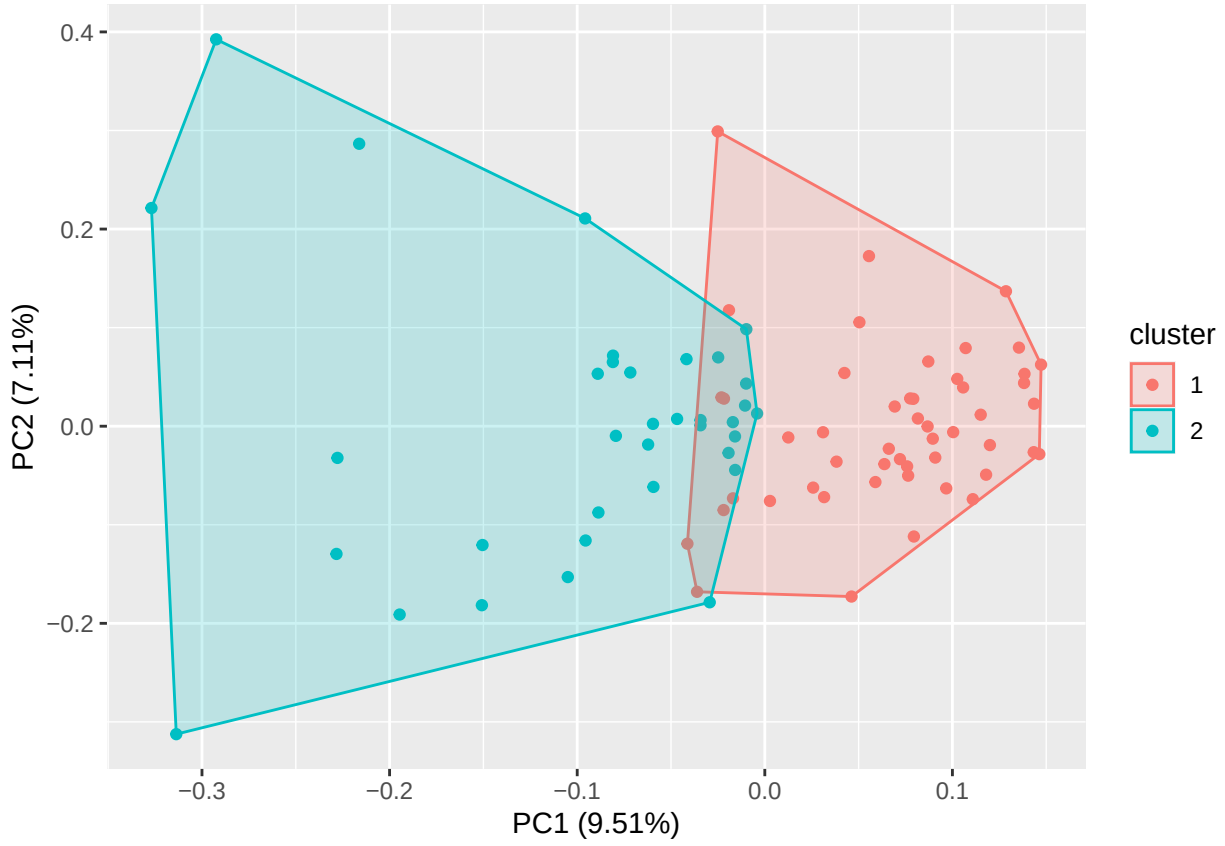
K Means

```
set.seed(1234)

#Unsupervised Learning - Convert data to unlabeled
kmeansInput <- data.frame(fedpapers[,-1])

kmeansInput <- scale(kmeansInput, center = T, scale = T)

#WSS Plot to Choose Maximum number of Clusters

wssplot <- function(data, nc=15, seed=1234){
                  wss <- (nrow(data)-1)*sum(apply(data,2,var))
                      for (i in 2:nc){
                set.seed(seed)
                    wss[i] <- sum(kmeans(data, centers=i)$withinss)}
              plot(1:nc, wss, type="b", xlab="Number of Clusters",
                           ylab="Within groups sum of squares")
              wss
}

wssplot(kmeansInput)
```




```
##  [1] 5880.000 5455.868 5217.033 4999.715 4741.856 4566.792 4395.029 4357.018
##  [9] 4140.531 4109.360 3978.680 3827.741 3724.084 3602.242 3520.792
```

```
#K Means Cluster
km = kmeans(kmeansInput, 2, nstart = 30)


#Look at the Cluster Plot
autoplot(km, kmeansInput, frame=TRUE)
```



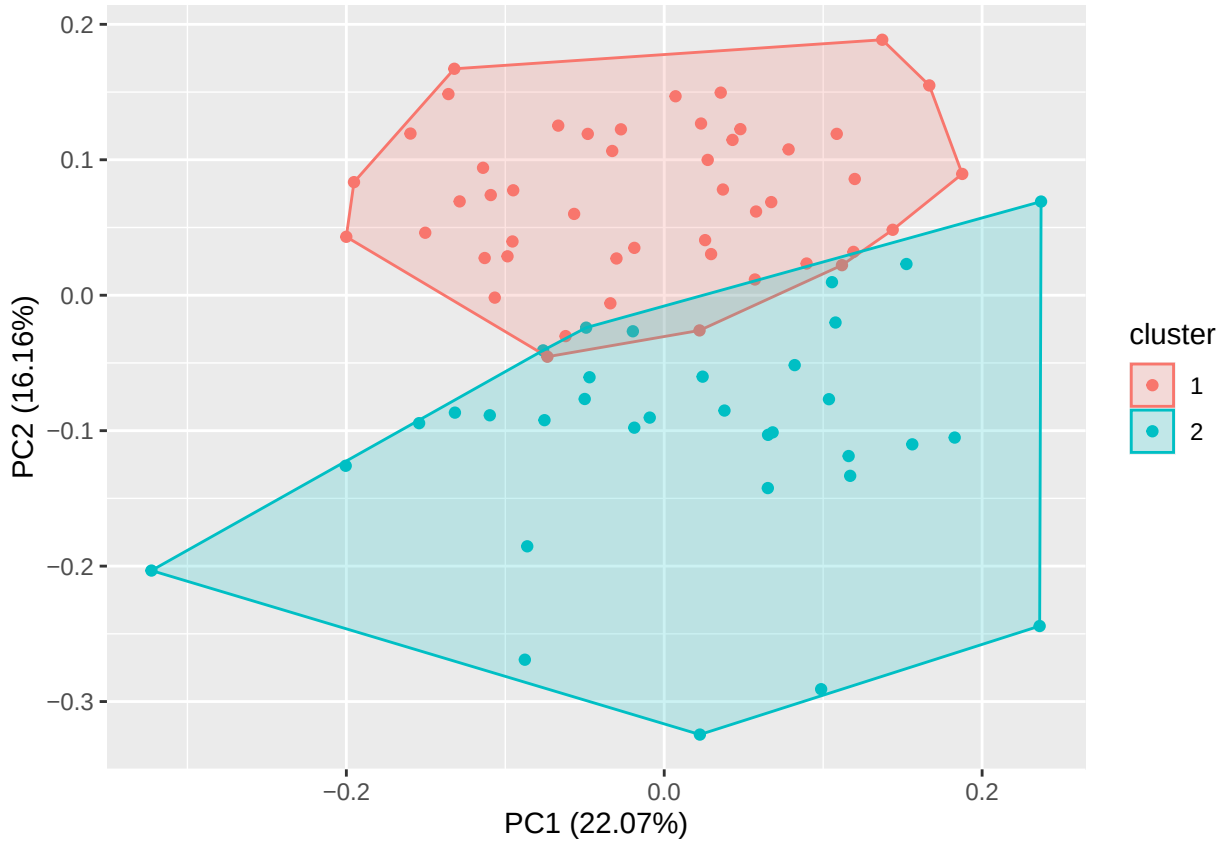

```
#Check Centers
#km$centers

table(fedpapers$author, km$cluster)
```

```
##
##              1  2
##   dispt      1 10
##   Hamilton 49  2
##   HM         0  3
##   Jay        0  5
##   Madison    0 15
```

Below, I clusterwith only Hamilton, Madison, HM and "dispt" authors. I do this as the disputed papers are disputed between Hamilton and Madison, not to include "Jay".

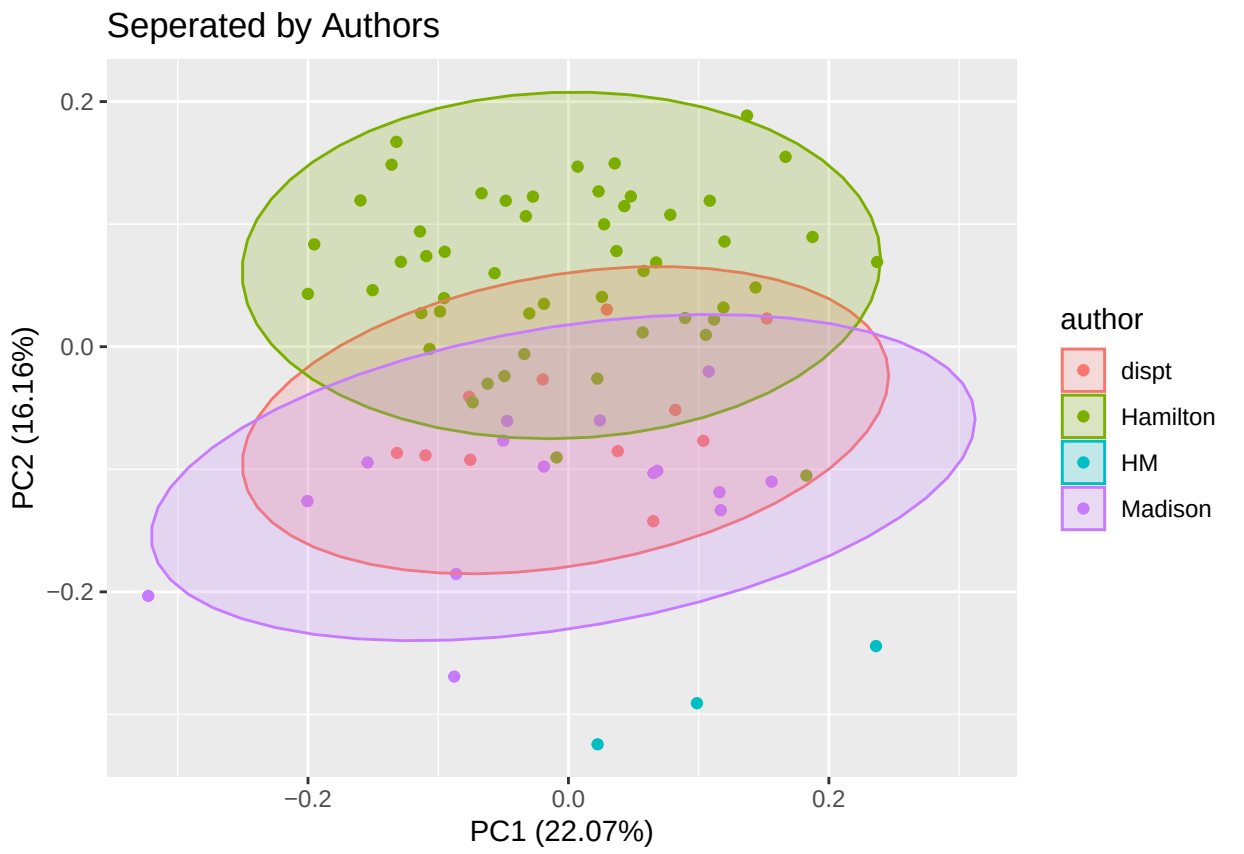
```
kmeansInput2 <- data.frame(fedpapers[fedpapers$author == "Hamilton" | fedpapers$author == "Madison" | f

kmeansInput2scaled <- scale(kmeansInput2[,-1])

#Running K means with 2 clusters
```

```
km2 = kmeans(kmeansInput2scaled, 2)

#Plot again with Autoplot
autoplot(km2, kmeansInput2, frame=TRUE)
```




```
autoplot(km2, kmeansInput2, colour = 'author',  frame.type = 't') + ggtitle("Seperated by Authors")
```

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




```
#view Cluster Centers
#km2$centers

table(kmeansInput2$author, km2$cluster)
```

```
##
##              1  2
##   dispt      1 10
##   Hamilton 46  5
##   HM         0  3
##   Madison    0 15
```

Summary of K Means

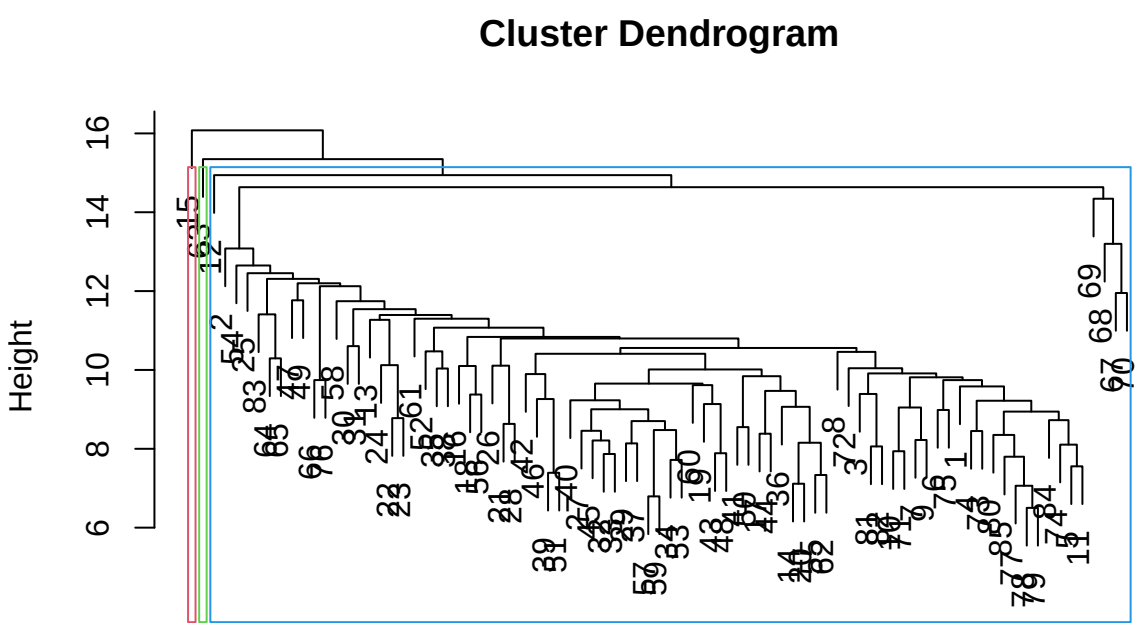
After reviewing the results from both outputs (Km & Km2) generated by k- Means. After reviewing "km2, the output points to the conclusion that that all papers possessing the"HM" author attribute are **likely** (based solely on the data provided) to have been authored by James Madison to include 10 of the 11 disputed papers.

HAC First pass Euclidean Distance

In the following code block I begin my HAC analysis by preparing the data set without labels. I then normalize that data frame, compute distance and clusters via Euclidean and average methods. This approach

gave undesirable results (table provided at end of code block), which gave no conclusive insight or clues as both Hamilton's and Madison's papers resided within the same cluster. It is because of the contents of cluster "1", that prompted a change in the method of distance calculation.

```
set.seed(786)



papers_df <- fedpapers

#Prepare a (Author) paper labels set
papers_label <- papers_df$author


#Normalize the Data
papers_dfsc <- scale(papers_df[,-1])


#Compute distance via Euclidean method
dist_mat <- dist(papers_dfsc, method = 'euclidean')

#calculate clusters by average method
hclust_avg <- hclust(dist_mat, method = 'average')



plot(hclust_avg)

cut_avg <- cutree(hclust_avg, k = 3)


rect.hclust(hclust_avg , k = 3, border = 2:6)
abline(h = 3, col = 'red')
```




```
library(dendextend)
```

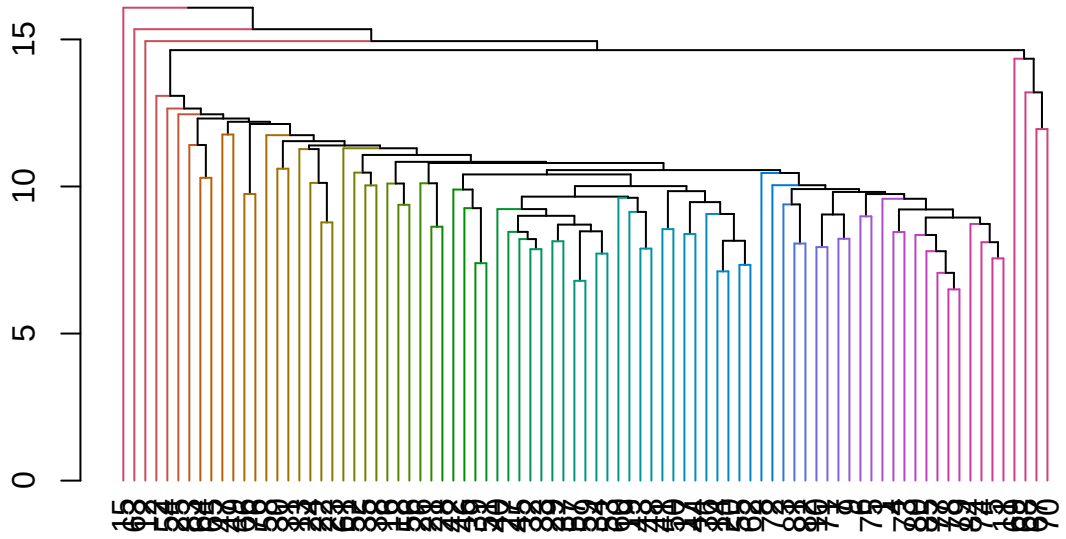
```
## Warning: package 'dendextend' was built under R version 4.1.1
```

```
##
## --------------------
## Welcome to dendextend version 1.15.2
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues
## You may ask questions at stackoverflow, use the r and dendextend tags:
##   https://stackoverflow.com/questions/tagged/dendextend
##
##  To suppress this message use:  suppressPackageStartupMessages(library(dendextend))
## --------------------
```

```
##
## Attaching package: 'dendextend'
```

```
## The following object is masked from 'package:stats':
##
##     cutree
```

```
avg_dend_obj <- as.dendrogram(hclust_avg)
avg_col_dend <- color_branches(avg_dend_obj, h = 3)
plot(avg_col_dend)
```




```
papers_df_cl <- mutate(papers_df, cluster = cut_avg)
count(papers_df_cl,cluster)
```

```
##   cluster  n
## 1       1 83
## 2       2  1
## 3       3  1
```

```
table(papers_df_cl$cluster,papers_label)
```

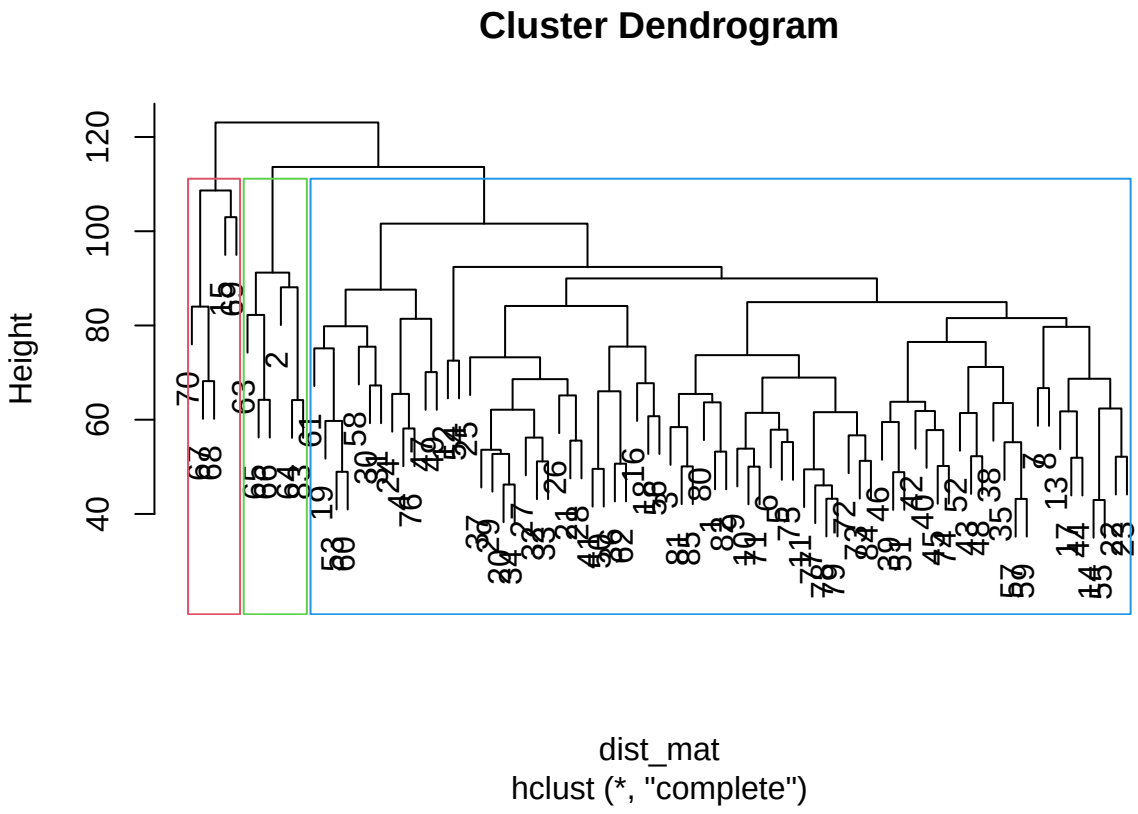
```
##    papers_label
##     dispt Hamilton HM Jay Madison
##   1    11       50  2   5      15
##   2     0        1  0   0       0
##   3     0        0  1   0       0
```

HAC Second Pass - Manhattan Distance

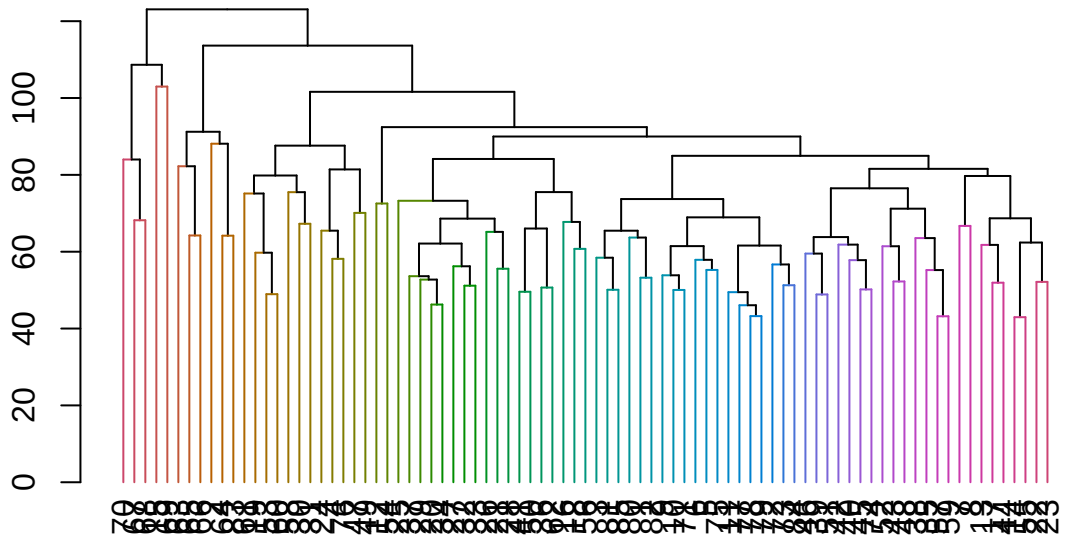
Utilizing a new distance method, similar results to the first pass were generated. Further exploration was done via different clustering methods also provided inconclusive results. Modifying the k parameter to 4

produced clusters to possessed nearly equal quantities of Hamilton and Madison papers, prompting further exploration in modifying distance and clustering calculation methods.

```
dist_mat <- dist(papers_dfsc, method = 'manhattan')

hclust_avg <- hclust(dist_mat, method = 'complete')



plot(hclust_avg)

cut_avg <- cutree(hclust_avg, k = 3)


rect.hclust(hclust_avg , k = 3, border = 2:6)
abline(h = 4, col = 'red')
```




```
library(dendextend)

avg_dend_obj <- as.dendrogram(hclust_avg)
avg_col_dend <- color_branches(avg_dend_obj, h = 4)
plot(avg_col_dend)
```




```
papers_df_cl <- mutate(papers_df, cluster = cut_avg)
count(papers_df_cl,cluster)
```

```
##   cluster  n
## 1       1 74
## 2       2  6
## 3       3  5
```

```
table(papers_df_cl$cluster,papers_label)
```

```
##    papers_label
##     dispt Hamilton HM Jay Madison
##   1    10       50  0   0      14
##   2     1        0  3   1       1
##   3     0        1  0   4       0
```

HAC Canberra - First Pass

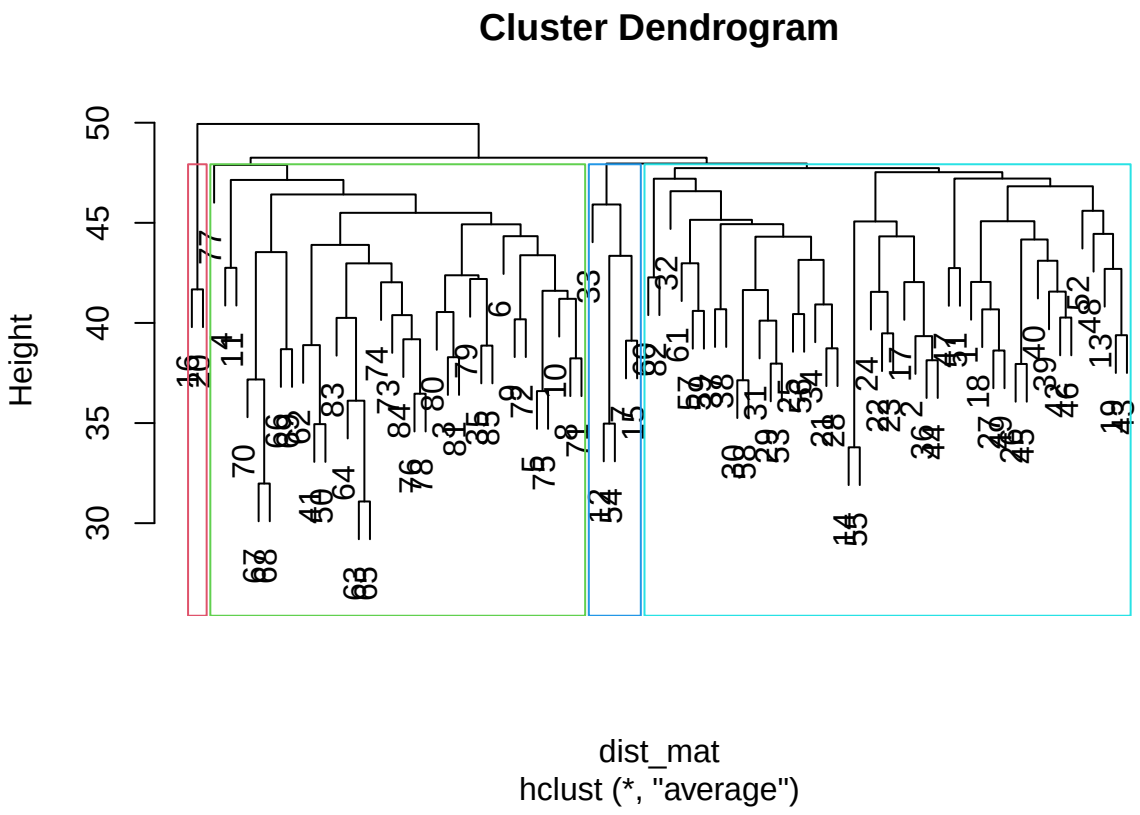
In an attempt to gain **broad** insight to identifying potential clusters; the following parameters we utilized: Distance Method - Canberra Clustering Method - Average k - 4

These parameters produced more insightful, yet not conclusive results. The table displayed (via R code) at the end of the below code block shows a vast majority of Hamilton papers residing within a single cluster

(41 total - Cluster 1), along with nearly all Madison papers residing within a seperate cluster (14 - cluster 2).

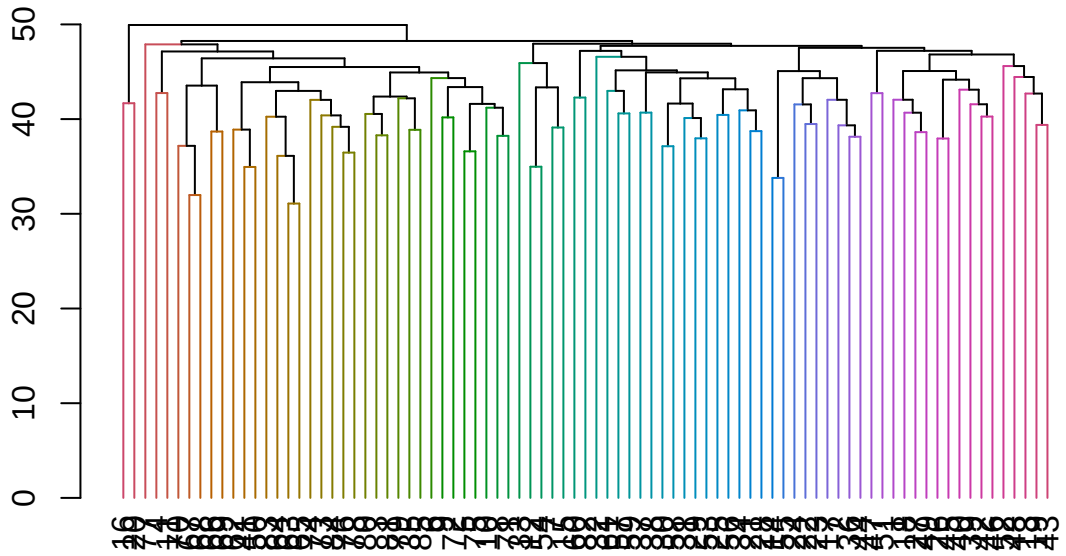
This indicates that the distance and cluster methods produced a more favorable result, broadly grouping each of the author's known works within a single cluster. This output does provide hints to who the disputed and "HM" labeled examples can be attributed; however another pass with the same distance and clustering methods, but a reduced K value may provide more meaningful insight (next code block).

```
dist_mat <- dist(papers_dfsc, method = 'canberra')

hclust_avg <- hclust(dist_mat, method = 'average')



plot(hclust_avg)

cut_avg <- cutree(hclust_avg, k = 4)


rect.hclust(hclust_avg , k = 4, border = 2:6)
abline(h = 4, col = 'red')
```




```
library(dendextend)

avg_dend_obj <- as.dendrogram(hclust_avg)
```

```
avg_col_dend <- color_branches(avg_dend_obj, h = 4)
plot(avg_col_dend)
```




```
papers_df_cl <- mutate(papers_df, cluster = cut_avg)
count(papers_df_cl,cluster)
```

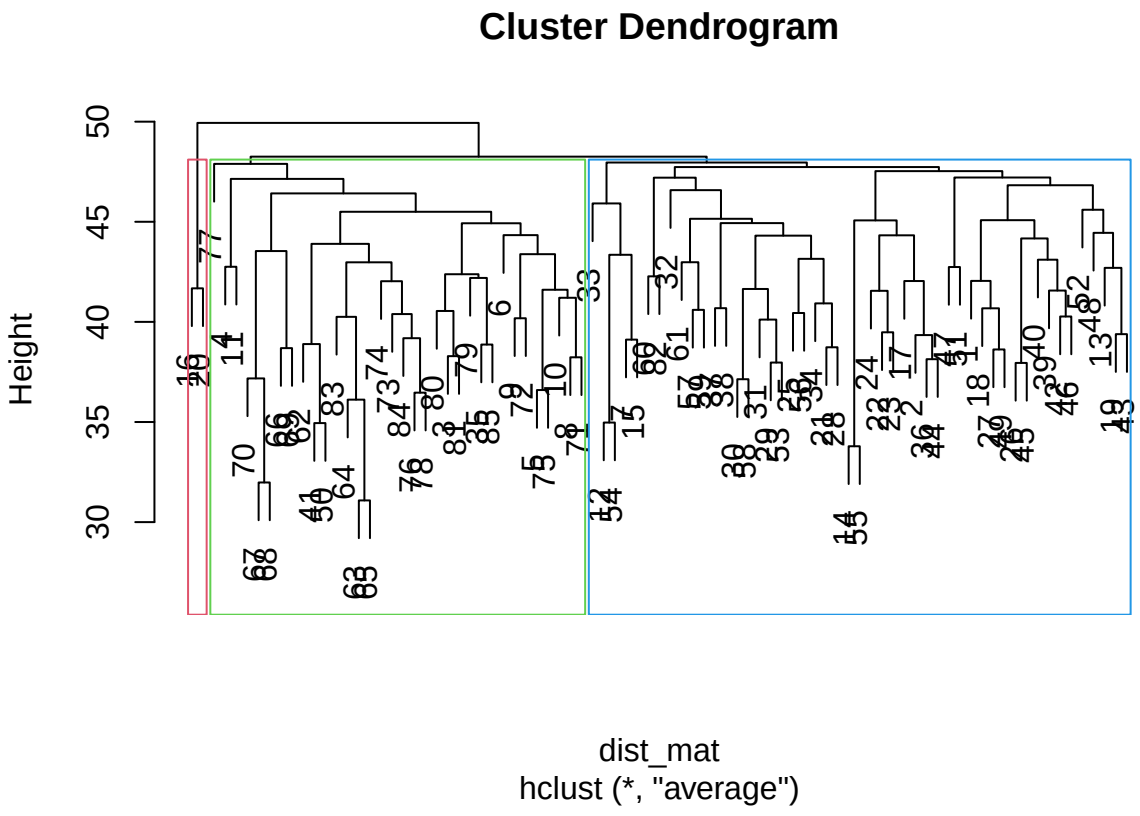
```
##   cluster  n
## 1       1 44
## 2       2 34
## 3       3  5
## 4       4  2
```

```
table(papers_df_cl$cluster,papers_label)
```

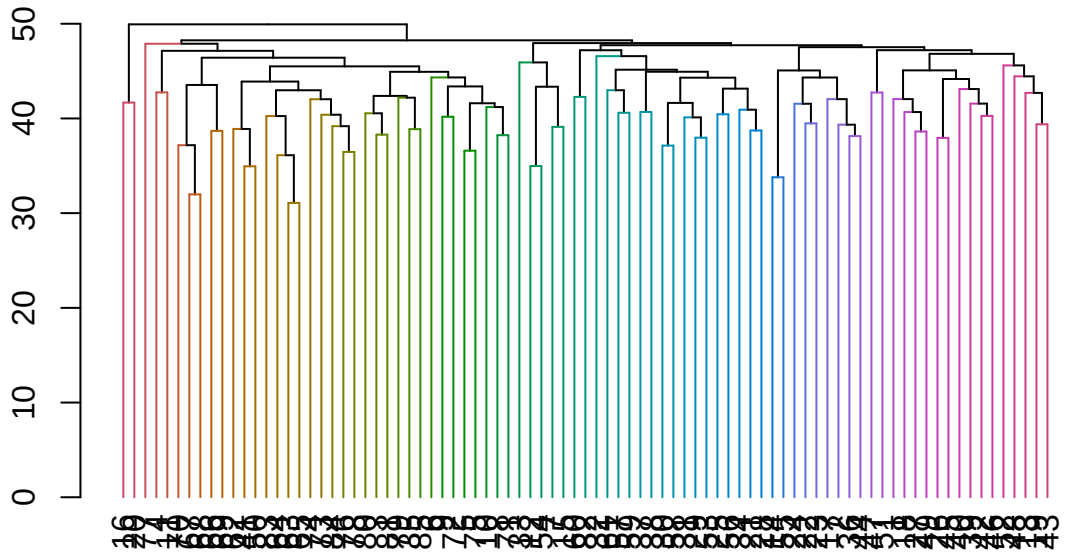
```
##    papers_label
##     dispt Hamilton HM Jay Madison
##   1     2       41  0   0       1
##   2     8        4  3   5      14
##   3     1        4  0   0       0
##   4     0        2  0   0       0
```

HAC Second Attempt - Canberra Distance Summary & Conclution (Estimated Author Attribution)

```
dist_mat <- dist(papers_dfsc, method = 'canberra')

hclust_avg <- hclust(dist_mat, method = 'average')



plot(hclust_avg)

cut_avg <- cutree(hclust_avg, k = 3)


rect.hclust(hclust_avg , k = 3, border = 2:6)
abline(h = 4, col = 'red')
```




```
library(dendextend)

avg_dend_obj <- as.dendrogram(hclust_avg)
avg_col_dend <- color_branches(avg_dend_obj, h = 4)
plot(avg_col_dend)
```




```
papers_df_cl <- mutate(papers_df, cluster = cut_avg)
count(papers_df_cl,cluster)
```

```
##   cluster  n
## 1       1 49
## 2       2 34
## 3       3  2
```

```
table(papers_df_cl$cluster,papers_label)
```

```
##    papers_label
##     dispt Hamilton HM Jay Madison
##   1     3       45  0   0       1
##   2     8        4  3   5      14
##   3     0        2  0   0       0
```

```
#HAC Result
print("HAC Result")
```

```
## [1] "HAC Result"
```

```
table(papers_df_cl$cluster,papers_label)
```

```
##    papers_label
##     dispt Hamilton HM Jay Madison
##   1     3       45  0   0       1
##   2     8        4  3   5      14
##   3     0        2  0   0       0
```

```
#K Means Result
print("K Means Result")
```

```
## [1] "K Means Result"
```

```
table(kmeansInput2$author, km2$cluster)
```

```
##
##              1  2
##   dispt      1 10
##   Hamilton 46  5
##   HM        0  3
##   Madison   0 15
```

When maintaining the same distance and clustering method parameters, but modifying the k value, we arrive at the most meaningful results thus far (table displayed above "table(papers_df_cl$cluster,papers_label)").

Here we see overwhelming clustering of Hamilton within cluster 1 and Madison within cluster 2. All "HM" (author attribute) example reside within cluster 2 as well as 8 of the disputed papers, given this result and the data provided Madison is most likely to have authored these papers.

Conclusion

After examining both k-Means and HAC results, their parameters and data provided, it is estimated from this analysis that James Madison is most likely to have written examples labeled both "dispt" and "HM".