Sharat\_Sripada\_HW4

# install.packages('tm')  
# install.packages('tmap')  
# install.packages('quanteda')  
# install.packages('philentropy')  
# install.packages('factoextra')  
  
library(tm)

## Loading required package: NLP

library(tmap)  
library(quanteda)

## Package version: 2.1.1

## Parallel computing: 2 of 4 threads used.

## See https://quanteda.io for tutorials and examples.

##   
## Attaching package: 'quanteda'

## The following objects are masked from 'package:tm':  
##   
## as.DocumentTermMatrix, stopwords

## The following objects are masked from 'package:NLP':  
##   
## meta, meta<-

## The following object is masked from 'package:utils':  
##   
## View

library(RColorBrewer)  
library(wordcloud)  
library(philentropy)  
library(factoextra)

## Loading required package: ggplot2

##   
## Attaching package: 'ggplot2'

## The following object is masked from 'package:NLP':  
##   
## annotate

## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa

## Introduction

This week delves on concepts of clustering viz. k-means, HAC and various distance measurements that aid in the process namely, Eucledian and cosine methods. In particular, the home-work will attempt to solve the problem of classifying disputed papers between authors Hamilton and Madison.

We begin our analysis by ingesting a corpus of documents and running through the following pipelines:

* loading the documents using the R Corpus function
* build a document term matrix (DTM)
* visualize wordclouds
* dive into the core concepts of clustering
* classify disputed documents from results of clustering

#load the data/corpus  
FedPapersCorpus <- Corpus(DirSource("/Users/venkatasharatsripada/Downloads/IST707repo-master/FedPapersCorpus"))  
numFedPapers <- length(FedPapersCorpus)

summary(FedPapersCorpus)

## Length Class Mode  
## dispt\_fed\_49.txt 2 PlainTextDocument list  
## dispt\_fed\_50.txt 2 PlainTextDocument list  
## dispt\_fed\_51.txt 2 PlainTextDocument list  
## dispt\_fed\_52.txt 2 PlainTextDocument list  
## dispt\_fed\_53.txt 2 PlainTextDocument list  
## dispt\_fed\_54.txt 2 PlainTextDocument list  
## dispt\_fed\_55.txt 2 PlainTextDocument list  
## dispt\_fed\_56.txt 2 PlainTextDocument list  
## dispt\_fed\_57.txt 2 PlainTextDocument list  
## dispt\_fed\_62.txt 2 PlainTextDocument list  
## dispt\_fed\_63.txt 2 PlainTextDocument list  
## Hamilton\_fed\_1.txt 2 PlainTextDocument list  
## Hamilton\_fed\_11.txt 2 PlainTextDocument list  
## Hamilton\_fed\_12.txt 2 PlainTextDocument list  
## Hamilton\_fed\_13.txt 2 PlainTextDocument list  
## Hamilton\_fed\_15.txt 2 PlainTextDocument list  
## Hamilton\_fed\_16.txt 2 PlainTextDocument list  
## Hamilton\_fed\_17.txt 2 PlainTextDocument list  
## Hamilton\_fed\_21.txt 2 PlainTextDocument list  
## Hamilton\_fed\_22.txt 2 PlainTextDocument list  
## Hamilton\_fed\_23.txt 2 PlainTextDocument list  
## Hamilton\_fed\_24.txt 2 PlainTextDocument list  
## Hamilton\_fed\_25.txt 2 PlainTextDocument list  
## Hamilton\_fed\_26.txt 2 PlainTextDocument list  
## Hamilton\_fed\_27.txt 2 PlainTextDocument list  
## Hamilton\_fed\_28.txt 2 PlainTextDocument list  
## Hamilton\_fed\_29.txt 2 PlainTextDocument list  
## Hamilton\_fed\_30.txt 2 PlainTextDocument list  
## Hamilton\_fed\_31.txt 2 PlainTextDocument list  
## Hamilton\_fed\_32.txt 2 PlainTextDocument list  
## Hamilton\_fed\_33.txt 2 PlainTextDocument list  
## Hamilton\_fed\_34.txt 2 PlainTextDocument list  
## Hamilton\_fed\_35.txt 2 PlainTextDocument list  
## Hamilton\_fed\_36.txt 2 PlainTextDocument list  
## Hamilton\_fed\_59.txt 2 PlainTextDocument list  
## Hamilton\_fed\_6.txt 2 PlainTextDocument list  
## Hamilton\_fed\_60.txt 2 PlainTextDocument list  
## Hamilton\_fed\_61.txt 2 PlainTextDocument list  
## Hamilton\_fed\_65.txt 2 PlainTextDocument list  
## Hamilton\_fed\_66.txt 2 PlainTextDocument list  
## Hamilton\_fed\_67.txt 2 PlainTextDocument list  
## Hamilton\_fed\_68.txt 2 PlainTextDocument list  
## Hamilton\_fed\_69.txt 2 PlainTextDocument list  
## Hamilton\_fed\_7.txt 2 PlainTextDocument list  
## Hamilton\_fed\_70.txt 2 PlainTextDocument list  
## Hamilton\_fed\_71.txt 2 PlainTextDocument list  
## Hamilton\_fed\_72.txt 2 PlainTextDocument list  
## Hamilton\_fed\_73.txt 2 PlainTextDocument list  
## Hamilton\_fed\_74.txt 2 PlainTextDocument list  
## Hamilton\_fed\_75.txt 2 PlainTextDocument list  
## Hamilton\_fed\_76.txt 2 PlainTextDocument list  
## Hamilton\_fed\_77.txt 2 PlainTextDocument list  
## Hamilton\_fed\_78.txt 2 PlainTextDocument list  
## Hamilton\_fed\_79.txt 2 PlainTextDocument list  
## Hamilton\_fed\_8.txt 2 PlainTextDocument list  
## Hamilton\_fed\_80.txt 2 PlainTextDocument list  
## Hamilton\_fed\_81.txt 2 PlainTextDocument list  
## Hamilton\_fed\_82.txt 2 PlainTextDocument list  
## Hamilton\_fed\_83.txt 2 PlainTextDocument list  
## Hamilton\_fed\_84.txt 2 PlainTextDocument list  
## Hamilton\_fed\_85.txt 2 PlainTextDocument list  
## Hamilton\_fed\_9.txt 2 PlainTextDocument list  
## HM\_fed\_18.txt 2 PlainTextDocument list  
## HM\_fed\_19.txt 2 PlainTextDocument list  
## HM\_fed\_20.txt 2 PlainTextDocument list  
## Jay\_fed\_2.txt 2 PlainTextDocument list  
## Jay\_fed\_3.txt 2 PlainTextDocument list  
## Jay\_fed\_4.txt 2 PlainTextDocument list  
## Jay\_fed\_5.txt 2 PlainTextDocument list  
## Jay\_fed\_64.txt 2 PlainTextDocument list  
## Madison\_fed\_10.txt 2 PlainTextDocument list  
## Madison\_fed\_14.txt 2 PlainTextDocument list  
## Madison\_fed\_37.txt 2 PlainTextDocument list  
## Madison\_fed\_38.txt 2 PlainTextDocument list  
## Madison\_fed\_39.txt 2 PlainTextDocument list  
## Madison\_fed\_40.txt 2 PlainTextDocument list  
## Madison\_fed\_41.txt 2 PlainTextDocument list  
## Madison\_fed\_42.txt 2 PlainTextDocument list  
## Madison\_fed\_43.txt 2 PlainTextDocument list  
## Madison\_fed\_44.txt 2 PlainTextDocument list  
## Madison\_fed\_45.txt 2 PlainTextDocument list  
## Madison\_fed\_46.txt 2 PlainTextDocument list  
## Madison\_fed\_47.txt 2 PlainTextDocument list  
## Madison\_fed\_48.txt 2 PlainTextDocument list  
## Madison\_fed\_58.txt 2 PlainTextDocument list

# meta(FedPapersCorpus[[1]])

#Ignore extremely rare words - <2% of documents  
(minTermFreq <- 0.02 \* numFedPapers)

## [1] 1.7

#Also, ignore common words - >75%-95% of documents  
(maxTermFreq <- 0.95 \* numFedPapers)

## [1] 80.75

#   
Papers\_DTM <- DocumentTermMatrix(FedPapersCorpus,  
 control=list(  
 stopwords=TRUE,  
 wordLengths=c(3,15),  
 removePunctuation=T,  
 removeNumbers=T,  
 tolower=T,  
 stemming=T,  
 remove\_separators=T,  
 bounds=list(global=c(minTermFreq, maxTermFreq))  
 ))  
DTM <- as.matrix(Papers\_DTM)  
(DTM[1:11,1:10])

## Terms  
## Docs abandon abat abb abet abil abl ablest abolish abolit abort  
## dispt\_fed\_49.txt 0 0 0 0 0 2 0 0 0 0  
## dispt\_fed\_50.txt 0 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_51.txt 0 0 0 0 0 1 0 0 0 0  
## dispt\_fed\_52.txt 0 0 0 0 1 1 0 0 0 0  
## dispt\_fed\_53.txt 0 1 0 0 0 0 0 0 0 0  
## dispt\_fed\_54.txt 0 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_55.txt 0 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_56.txt 0 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_57.txt 0 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_62.txt 0 0 0 0 0 1 0 0 0 0  
## dispt\_fed\_63.txt 0 0 0 0 0 4 0 0 0 0

col\_WordFreq <- colSums(as.matrix(Papers\_DTM))  
(head(col\_WordFreq))

## abandon abat abb abet abil abl   
## 9 2 5 2 15 74

#Length of all words  
(length(col\_WordFreq))

## [1] 3370

(row\_WordFreq <- rowSums(as.matrix(Papers\_DTM)))

## dispt\_fed\_49.txt dispt\_fed\_50.txt dispt\_fed\_51.txt dispt\_fed\_52.txt   
## 677 480 783 743   
## dispt\_fed\_53.txt dispt\_fed\_54.txt dispt\_fed\_55.txt dispt\_fed\_56.txt   
## 903 766 865 649   
## dispt\_fed\_57.txt dispt\_fed\_62.txt dispt\_fed\_63.txt Hamilton\_fed\_1.txt   
## 889 983 1244 659   
## Hamilton\_fed\_11.txt Hamilton\_fed\_12.txt Hamilton\_fed\_13.txt Hamilton\_fed\_15.txt   
## 1020 901 400 1256   
## Hamilton\_fed\_16.txt Hamilton\_fed\_17.txt Hamilton\_fed\_21.txt Hamilton\_fed\_22.txt   
## 814 663 823 1494   
## Hamilton\_fed\_23.txt Hamilton\_fed\_24.txt Hamilton\_fed\_25.txt Hamilton\_fed\_26.txt   
## 717 826 825 983   
## Hamilton\_fed\_27.txt Hamilton\_fed\_28.txt Hamilton\_fed\_29.txt Hamilton\_fed\_30.txt   
## 573 639 876 819   
## Hamilton\_fed\_31.txt Hamilton\_fed\_32.txt Hamilton\_fed\_33.txt Hamilton\_fed\_34.txt   
## 673 589 640 883   
## Hamilton\_fed\_35.txt Hamilton\_fed\_36.txt Hamilton\_fed\_59.txt Hamilton\_fed\_6.txt   
## 942 1095 720 868   
## Hamilton\_fed\_60.txt Hamilton\_fed\_61.txt Hamilton\_fed\_65.txt Hamilton\_fed\_66.txt   
## 892 591 816 899   
## Hamilton\_fed\_67.txt Hamilton\_fed\_68.txt Hamilton\_fed\_69.txt Hamilton\_fed\_7.txt   
## 688 604 1174 952   
## Hamilton\_fed\_70.txt Hamilton\_fed\_71.txt Hamilton\_fed\_72.txt Hamilton\_fed\_73.txt   
## 1295 677 842 941   
## Hamilton\_fed\_74.txt Hamilton\_fed\_75.txt Hamilton\_fed\_76.txt Hamilton\_fed\_77.txt   
## 422 822 796 798   
## Hamilton\_fed\_78.txt Hamilton\_fed\_79.txt Hamilton\_fed\_8.txt Hamilton\_fed\_80.txt   
## 1245 421 892 974   
## Hamilton\_fed\_81.txt Hamilton\_fed\_82.txt Hamilton\_fed\_83.txt Hamilton\_fed\_84.txt   
## 1581 642 2374 1656   
## Hamilton\_fed\_85.txt Hamilton\_fed\_9.txt HM\_fed\_18.txt HM\_fed\_19.txt   
## 1114 808 926 907   
## HM\_fed\_20.txt Jay\_fed\_2.txt Jay\_fed\_3.txt Jay\_fed\_4.txt   
## 692 709 622 663   
## Jay\_fed\_5.txt Jay\_fed\_64.txt Madison\_fed\_10.txt Madison\_fed\_14.txt   
## 605 966 1316 882   
## Madison\_fed\_37.txt Madison\_fed\_38.txt Madison\_fed\_39.txt Madison\_fed\_40.txt   
## 1122 1348 981 1132   
## Madison\_fed\_41.txt Madison\_fed\_42.txt Madison\_fed\_43.txt Madison\_fed\_44.txt   
## 1479 1140 1344 1178   
## Madison\_fed\_45.txt Madison\_fed\_46.txt Madison\_fed\_47.txt Madison\_fed\_48.txt   
## 810 980 1167 738   
## Madison\_fed\_58.txt   
## 847

### Normalization

#create a normalized version of Papers\_DTM  
Papers\_M <- as.matrix(Papers\_DTM)  
Papers\_M\_N1 <- apply(Papers\_M, 1, function(i) round(i/sum(i),3))  
Papers\_Matrix\_Norm <- t(Papers\_M\_N1)  
  
#compare the original and normalized version  
(Papers\_M[c(1:11),c(1000:1010)])

## Terms  
## Docs edit effect effectu efficaci effici effort eight eighth  
## dispt\_fed\_49.txt 0 1 1 0 0 0 0 0  
## dispt\_fed\_50.txt 0 3 0 0 0 0 0 0  
## dispt\_fed\_51.txt 0 0 0 0 0 0 0 0  
## dispt\_fed\_52.txt 0 1 1 0 0 0 0 0  
## dispt\_fed\_53.txt 0 2 1 0 0 0 0 0  
## dispt\_fed\_54.txt 0 3 0 2 0 0 0 0  
## dispt\_fed\_55.txt 0 0 0 0 0 0 1 0  
## dispt\_fed\_56.txt 0 2 0 0 0 0 3 0  
## dispt\_fed\_57.txt 0 0 2 0 0 0 0 0  
## dispt\_fed\_62.txt 0 4 0 0 0 0 0 0  
## dispt\_fed\_63.txt 0 2 2 0 0 0 0 0  
## Terms  
## Docs either elaps elect  
## dispt\_fed\_49.txt 1 0 1  
## dispt\_fed\_50.txt 3 0 2  
## dispt\_fed\_51.txt 0 0 1  
## dispt\_fed\_52.txt 0 0 21  
## dispt\_fed\_53.txt 2 1 20  
## dispt\_fed\_54.txt 0 0 1  
## dispt\_fed\_55.txt 2 0 3  
## dispt\_fed\_56.txt 2 0 3  
## dispt\_fed\_57.txt 0 0 10  
## dispt\_fed\_62.txt 0 0 2  
## dispt\_fed\_63.txt 0 0 14

(Papers\_Matrix\_Norm[c(1:11),c(1000:1010)])

## Terms  
## Docs edit effect effectu efficaci effici effort eight eighth  
## dispt\_fed\_49.txt 0 0.001 0.001 0.000 0 0 0.000 0  
## dispt\_fed\_50.txt 0 0.006 0.000 0.000 0 0 0.000 0  
## dispt\_fed\_51.txt 0 0.000 0.000 0.000 0 0 0.000 0  
## dispt\_fed\_52.txt 0 0.001 0.001 0.000 0 0 0.000 0  
## dispt\_fed\_53.txt 0 0.002 0.001 0.000 0 0 0.000 0  
## dispt\_fed\_54.txt 0 0.004 0.000 0.003 0 0 0.000 0  
## dispt\_fed\_55.txt 0 0.000 0.000 0.000 0 0 0.001 0  
## dispt\_fed\_56.txt 0 0.003 0.000 0.000 0 0 0.005 0  
## dispt\_fed\_57.txt 0 0.000 0.002 0.000 0 0 0.000 0  
## dispt\_fed\_62.txt 0 0.004 0.000 0.000 0 0 0.000 0  
## dispt\_fed\_63.txt 0 0.002 0.002 0.000 0 0 0.000 0  
## Terms  
## Docs either elaps elect  
## dispt\_fed\_49.txt 0.001 0.000 0.001  
## dispt\_fed\_50.txt 0.006 0.000 0.004  
## dispt\_fed\_51.txt 0.000 0.000 0.001  
## dispt\_fed\_52.txt 0.000 0.000 0.028  
## dispt\_fed\_53.txt 0.002 0.001 0.022  
## dispt\_fed\_54.txt 0.000 0.000 0.001  
## dispt\_fed\_55.txt 0.002 0.000 0.003  
## dispt\_fed\_56.txt 0.003 0.000 0.005  
## dispt\_fed\_57.txt 0.000 0.000 0.011  
## dispt\_fed\_62.txt 0.000 0.000 0.002  
## dispt\_fed\_63.txt 0.000 0.000 0.011

#verify for word 'embarrass' in document 'dispt\_fed\_62.txt' if the   
#normalization math is correct  
  
(row\_WordFreq)

## dispt\_fed\_49.txt dispt\_fed\_50.txt dispt\_fed\_51.txt dispt\_fed\_52.txt   
## 677 480 783 743   
## dispt\_fed\_53.txt dispt\_fed\_54.txt dispt\_fed\_55.txt dispt\_fed\_56.txt   
## 903 766 865 649   
## dispt\_fed\_57.txt dispt\_fed\_62.txt dispt\_fed\_63.txt Hamilton\_fed\_1.txt   
## 889 983 1244 659   
## Hamilton\_fed\_11.txt Hamilton\_fed\_12.txt Hamilton\_fed\_13.txt Hamilton\_fed\_15.txt   
## 1020 901 400 1256   
## Hamilton\_fed\_16.txt Hamilton\_fed\_17.txt Hamilton\_fed\_21.txt Hamilton\_fed\_22.txt   
## 814 663 823 1494   
## Hamilton\_fed\_23.txt Hamilton\_fed\_24.txt Hamilton\_fed\_25.txt Hamilton\_fed\_26.txt   
## 717 826 825 983   
## Hamilton\_fed\_27.txt Hamilton\_fed\_28.txt Hamilton\_fed\_29.txt Hamilton\_fed\_30.txt   
## 573 639 876 819   
## Hamilton\_fed\_31.txt Hamilton\_fed\_32.txt Hamilton\_fed\_33.txt Hamilton\_fed\_34.txt   
## 673 589 640 883   
## Hamilton\_fed\_35.txt Hamilton\_fed\_36.txt Hamilton\_fed\_59.txt Hamilton\_fed\_6.txt   
## 942 1095 720 868   
## Hamilton\_fed\_60.txt Hamilton\_fed\_61.txt Hamilton\_fed\_65.txt Hamilton\_fed\_66.txt   
## 892 591 816 899   
## Hamilton\_fed\_67.txt Hamilton\_fed\_68.txt Hamilton\_fed\_69.txt Hamilton\_fed\_7.txt   
## 688 604 1174 952   
## Hamilton\_fed\_70.txt Hamilton\_fed\_71.txt Hamilton\_fed\_72.txt Hamilton\_fed\_73.txt   
## 1295 677 842 941   
## Hamilton\_fed\_74.txt Hamilton\_fed\_75.txt Hamilton\_fed\_76.txt Hamilton\_fed\_77.txt   
## 422 822 796 798   
## Hamilton\_fed\_78.txt Hamilton\_fed\_79.txt Hamilton\_fed\_8.txt Hamilton\_fed\_80.txt   
## 1245 421 892 974   
## Hamilton\_fed\_81.txt Hamilton\_fed\_82.txt Hamilton\_fed\_83.txt Hamilton\_fed\_84.txt   
## 1581 642 2374 1656   
## Hamilton\_fed\_85.txt Hamilton\_fed\_9.txt HM\_fed\_18.txt HM\_fed\_19.txt   
## 1114 808 926 907   
## HM\_fed\_20.txt Jay\_fed\_2.txt Jay\_fed\_3.txt Jay\_fed\_4.txt   
## 692 709 622 663   
## Jay\_fed\_5.txt Jay\_fed\_64.txt Madison\_fed\_10.txt Madison\_fed\_14.txt   
## 605 966 1316 882   
## Madison\_fed\_37.txt Madison\_fed\_38.txt Madison\_fed\_39.txt Madison\_fed\_40.txt   
## 1122 1348 981 1132   
## Madison\_fed\_41.txt Madison\_fed\_42.txt Madison\_fed\_43.txt Madison\_fed\_44.txt   
## 1479 1140 1344 1178   
## Madison\_fed\_45.txt Madison\_fed\_46.txt Madison\_fed\_47.txt Madison\_fed\_48.txt   
## 810 980 1167 738   
## Madison\_fed\_58.txt   
## 847

#dispt\_fed\_62 has 798 words in total  
#there are 2x words of 'embarrass' so, 2/798 = 0.0025 ~0.003 (3 places after decimal)

### Data-structures

Papers\_dtm\_matrix <- as.matrix(Papers\_DTM)  
str(Papers\_dtm\_matrix)

## num [1:85, 1:3370] 0 0 0 0 0 0 0 0 0 0 ...  
## - attr(\*, "dimnames")=List of 2  
## ..$ Docs : chr [1:85] "dispt\_fed\_49.txt" "dispt\_fed\_50.txt" "dispt\_fed\_51.txt" "dispt\_fed\_52.txt" ...  
## ..$ Terms: chr [1:3370] "abandon" "abat" "abb" "abet" ...

Papers\_dtm\_matrix[c(1:11),c(2:10)]

## Terms  
## Docs abat abb abet abil abl ablest abolish abolit abort  
## dispt\_fed\_49.txt 0 0 0 0 2 0 0 0 0  
## dispt\_fed\_50.txt 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_51.txt 0 0 0 0 1 0 0 0 0  
## dispt\_fed\_52.txt 0 0 0 1 1 0 0 0 0  
## dispt\_fed\_53.txt 1 0 0 0 0 0 0 0 0  
## dispt\_fed\_54.txt 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_55.txt 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_56.txt 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_57.txt 0 0 0 0 0 0 0 0 0  
## dispt\_fed\_62.txt 0 0 0 0 1 0 0 0 0  
## dispt\_fed\_63.txt 0 0 0 0 4 0 0 0 0

### Convert to a data-frame

Papers\_DF <- as.data.frame(as.matrix(Papers\_DTM))  
str(Papers\_DF)

## 'data.frame': 85 obs. of 3370 variables:  
## $ abandon : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abat : num 0 0 0 0 1 0 0 0 0 0 ...  
## $ abb : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abet : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abil : num 0 0 0 1 0 0 0 0 0 0 ...  
## $ abl : num 2 0 1 1 0 0 0 0 0 1 ...  
## $ ablest : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abolish : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abolit : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abort : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abound : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abridg : num 0 0 0 1 0 0 0 0 0 0 ...  
## $ abroad : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ absolut : num 0 2 2 1 0 0 0 0 0 0 ...  
## $ absorb : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abstain : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abstract : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ absurd : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abund : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ abus : num 1 1 2 1 1 0 0 0 0 0 ...  
## $ abyss : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ acced : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ accept : num 0 0 0 0 0 0 0 0 0 1 ...  
## $ access : num 0 0 0 2 0 0 0 0 0 0 ...  
## $ accid : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ accident : num 0 0 0 1 0 0 0 0 0 0 ...  
## $ accommod : num 0 0 0 0 1 0 0 0 0 0 ...  
## $ accompani : num 0 0 0 0 0 0 0 1 0 0 ...  
## $ accomplic : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ accomplish : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ accord : num 0 0 0 0 1 2 2 1 1 0 ...  
## $ account : num 0 0 0 0 0 0 1 0 0 0 ...  
## $ accumul : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ accur : num 1 0 0 0 1 0 0 0 0 1 ...  
## $ accuraci : num 0 0 0 0 0 1 0 0 0 0 ...  
## $ accus : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ accustom : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ achaean : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ acknowledg : num 0 1 0 0 0 0 0 0 0 1 ...  
## $ acquaint : num 1 0 0 0 2 0 0 2 0 1 ...  
## $ acquiesc : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ acquir : num 1 0 0 0 5 0 0 2 0 0 ...  
## $ acquisit : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ act : num 0 0 0 1 2 1 0 1 0 1 ...  
## $ action : num 0 0 1 0 0 0 0 0 0 1 ...  
## $ activ : num 0 4 0 0 0 0 0 0 0 0 ...  
## $ actor : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ actual : num 1 2 0 0 4 0 0 0 1 0 ...  
## $ actuat : num 0 0 0 0 0 0 1 0 1 0 ...  
## $ adapt : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ add : num 0 0 0 0 1 0 0 1 1 0 ...  
## $ addict : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ addit : num 0 0 1 1 0 0 0 0 1 1 ...  
## $ address : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ adduc : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ adept : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ adequ : num 1 1 0 0 0 0 0 0 0 0 ...  
## $ adher : num 0 0 1 0 0 1 0 0 0 0 ...  
## $ adjourn : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ adjud : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ adjust : num 0 0 0 0 0 1 0 0 0 0 ...  
## $ administ : num 0 0 2 0 0 0 0 0 0 1 ...  
## $ administr : num 1 2 1 0 0 0 0 0 1 0 ...  
## $ admir : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ admiralti : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ admiss : num 0 0 0 0 0 1 0 0 1 1 ...  
## $ admit : num 1 0 3 0 1 5 2 0 1 0 ...  
## $ admitt : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ admonish : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ admonit : num 0 0 0 0 0 0 0 0 0 1 ...  
## $ adopt : num 0 0 0 1 0 1 0 0 0 1 ...  
## $ advanc : num 0 0 0 0 1 0 0 1 1 2 ...  
## $ advantag : num 4 1 0 2 2 4 0 1 0 7 ...  
## $ adventiti : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ adventur : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ advers : num 2 0 0 0 0 0 0 0 0 0 ...  
## $ adversari : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ advert : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ advertis : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ advic : num 0 0 0 0 0 0 0 0 0 1 ...  
## $ advis : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ advoc : num 0 0 0 0 0 1 0 1 0 0 ...  
## $ affair : num 0 0 1 0 9 0 1 5 0 4 ...  
## $ affect : num 0 0 0 1 0 0 0 0 1 1 ...  
## $ affin : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ affirm : num 0 0 0 0 2 0 0 0 0 1 ...  
## $ afford : num 0 0 0 0 1 0 0 0 0 0 ...  
## $ affront : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ afraid : num 0 0 0 0 0 0 1 0 0 0 ...  
## $ afterward : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ age : num 0 0 0 1 0 0 0 0 0 2 ...  
## $ agenc : num 0 0 1 0 0 0 0 0 0 1 ...  
## $ agent : num 1 1 0 0 0 0 0 0 0 0 ...  
## $ aggrand : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ aggrandiz : num 1 0 0 0 0 0 0 0 1 0 ...  
## $ aggreg : num 0 0 0 0 0 2 0 0 0 0 ...  
## $ aggress : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ aggressor : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ agit : num 0 0 0 0 0 0 0 0 0 0 ...  
## [list output truncated]

### Example word cloud

Breaking the word clouds based on the document list: - 1:11 -> disputed papers - 12:62 -> Hamilton papers - 63:70 -> Ignoring HM\_fed*, Jay\_fed* papers - 71:85 -> Madison papers

disputedpaperswc <- wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11,])

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : repres  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : branch  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : bodi  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : exampl  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : small  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, :  
## american could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : member  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : senat  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, :  
## maryland could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, :  
## charact could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, :  
## passion could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, :  
## advantag could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, :  
## independ could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : hous  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : former  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, :  
## without could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : might  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : mani  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : known  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : defect  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : measur  
## could not be fit on page. It will not be plotted.

## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, :  
## legislatur could not be fit on page. It will not be plotted.

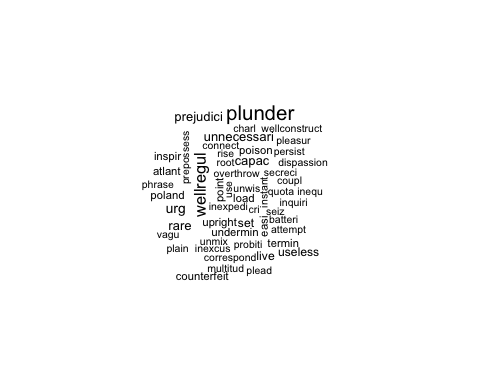
## Warning in wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[11, : two  
## could not be fit on page. It will not be plotted.



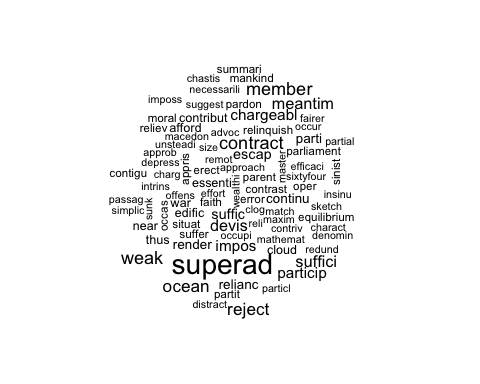
(head(sort(as.matrix(Papers\_DTM)[11,], decreasing = TRUE), n=50))

## senat repres bodi can elect measur corrupt   
## 24 18 15 14 14 11 9   
## nation constitut former reason year assembl exampl   
## 9 8 8 8 8 7 7   
## two annual danger everi evid feder import   
## 7 6 6 6 6 6 6   
## latter object particular public advantag ancient answer   
## 6 6 6 6 5 5 5   
## appear charact fact first hous institut less   
## 5 5 5 5 5 5 5   
## mani member might oper order popular probabl   
## 5 5 5 5 5 5 5   
## republ respons small term time whole without   
## 5 5 5 5 5 5 5   
## abl   
## 4

HamiltonPapersWC <- wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[12:62, ])



MadisonPapersWC <- wordcloud(colnames(Papers\_dtm\_matrix), Papers\_dtm\_matrix[71:85, ])



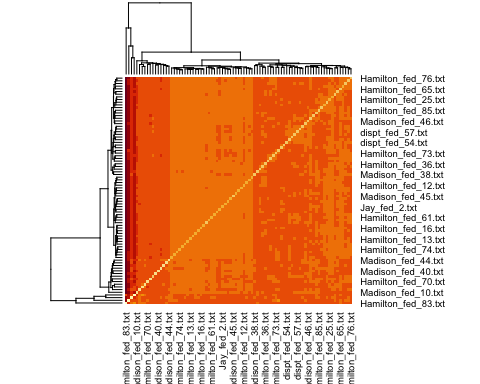
## Analysis

### Distance metrics

m <- Papers\_dtm\_matrix  
m\_norm <- Papers\_Matrix\_Norm  
  
distMatrix\_E <- distance(m, method='euclidean', use.row.names = TRUE)

## Metric: 'euclidean'; comparing: 85 vectors.

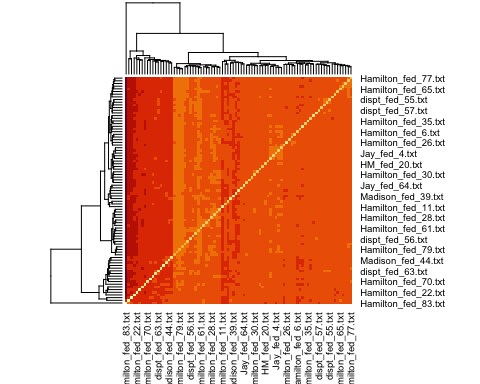
# print(distMatrix\_E)  
heatmap(distMatrix\_E)



distMatrix\_M <- distance(m, method='manhattan', use.row.names = TRUE)

## Metric: 'manhattan'; comparing: 85 vectors.

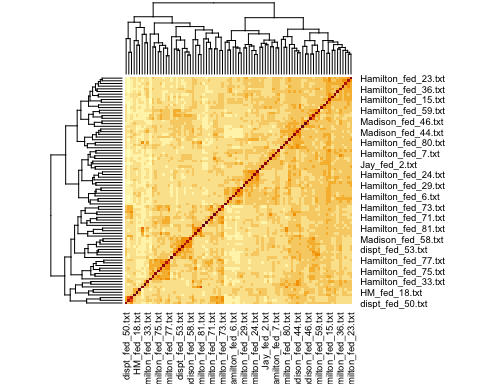
# print(distMatrix\_M)  
heatmap(distMatrix\_M)



distMatrix\_C <- distance(m, method = 'cosine', use.row.names = TRUE)

## Metric: 'cosine'; comparing: 85 vectors.

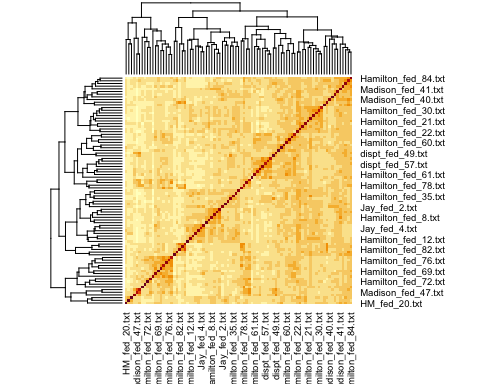
# print(distMatrix\_C)  
heatmap(distMatrix\_C)



distMatrix\_C\_norm <- distance(m\_norm, method='cosine', use.row.names = TRUE)

## Metric: 'cosine'; comparing: 85 vectors.

# print(distMatrix\_C\_norm)  
heatmap(distMatrix\_C\_norm)



The dist() function has issues with ‘cosine’ methods. Instead, used distance() function and obtain cosine similarity visualization. Heat-maps prove cosine similarity measurements are likely more suitable for document analysis.

### Data

We will explore the following two methods to cluster the data and determine an author to the disputed papers:

* K-means algorithm
* HAC algorithm

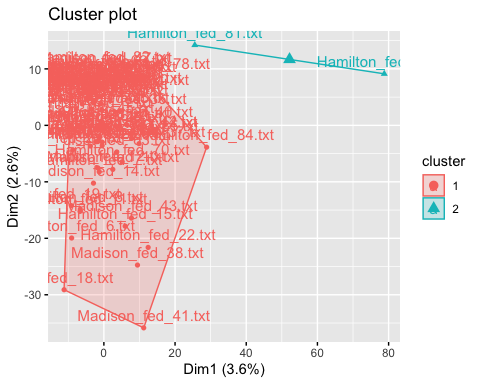
Given that the number of authors here are namely Hamilton and Madison, we will start with choosing number of clusters = 2.

First, is the k-means algorithm:

k <- 2  
set.seed(5)  
km.res <- kmeans(Papers\_dtm\_matrix, k, nstart=100, iter.max=50)  
str(km.res)

## List of 9  
## $ cluster : Named int [1:85] 1 1 1 1 1 1 1 1 1 1 ...  
## ..- attr(\*, "names")= chr [1:85] "dispt\_fed\_49.txt" "dispt\_fed\_50.txt" "dispt\_fed\_51.txt" "dispt\_fed\_52.txt" ...  
## $ centers : num [1:2, 1:3370] 0.1084 0 0.0241 0 0.0602 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:2] "1" "2"  
## .. ..$ : chr [1:3370] "abandon" "abat" "abb" "abet" ...  
## $ totss : num 202176  
## $ withinss : num [1:2] 174195 6448  
## $ tot.withinss: num 180642  
## $ betweenss : num 21533  
## $ size : int [1:2] 83 2  
## $ iter : int 1  
## $ ifault : int 0  
## - attr(\*, "class")= chr "kmeans"

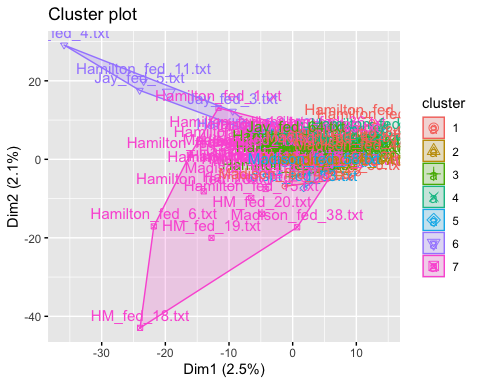
#plot a visualization  
fviz\_cluster(km.res, Papers\_dtm\_matrix)



k <- 7  
km.res <- kmeans(Papers\_Matrix\_Norm, k, nstart=50, iter.max=50)  
str(km.res)

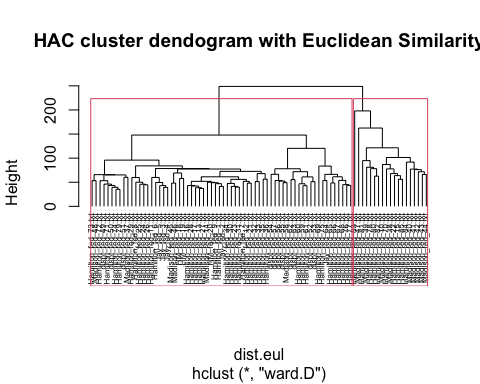
## List of 9  
## $ cluster : Named int [1:85] 1 1 1 5 5 5 5 5 5 7 ...  
## ..- attr(\*, "names")= chr [1:85] "dispt\_fed\_49.txt" "dispt\_fed\_50.txt" "dispt\_fed\_51.txt" "dispt\_fed\_52.txt" ...  
## $ centers : num [1:7, 1:3370] 7.69e-05 0.00 7.14e-05 0.00 0.00 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:7] "1" "2" "3" "4" ...  
## .. ..$ : chr [1:3370] "abandon" "abat" "abb" "abet" ...  
## $ totss : num 0.226  
## $ withinss : num [1:7] 0.03396 0.00231 0.02952 0.00754 0.02239 ...  
## $ tot.withinss: num 0.174  
## $ betweenss : num 0.0514  
## $ size : int [1:7] 13 2 14 4 10 5 37  
## $ iter : int 4  
## $ ifault : int 0  
## - attr(\*, "class")= chr "kmeans"

#plot a visualization  
fviz\_cluster(km.res, Papers\_Matrix\_Norm)

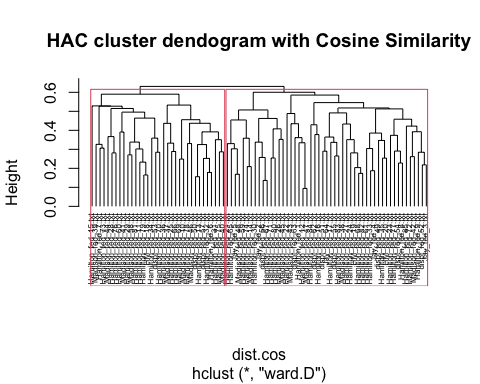


Now, we explore the HAC algorithms

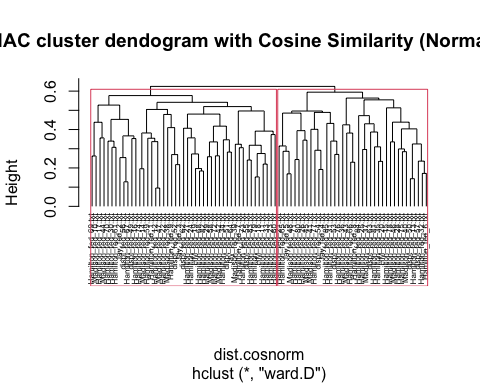
#Euclidean distance measure  
dist.eul <- as.dist(distMatrix\_E)  
groups\_E <- hclust(dist.eul, method='ward.D')  
  
#Visualizations  
plot(groups\_E, cex=0.5, font=22, hang=-1, main="HAC cluster dendogram with Euclidean Similarity")  
rect.hclust(groups\_E, k=2)



#Cosine distance measure  
dist.cos <- as.dist(distMatrix\_C)  
groups\_C <- hclust(dist.cos, method='ward.D')  
  
#Visualizations  
plot(groups\_C, cex=0.5, font=22, hang=-1, main="HAC cluster dendogram with Cosine Similarity")  
rect.hclust(groups\_C, k=2)



#Cosine distance measure (Normalized)  
dist.cosnorm <- as.dist(distMatrix\_C\_norm)  
groups\_C\_norm <- hclust(dist.cosnorm, method='ward.D')  
  
#Visualizations  
plot(groups\_C\_norm, cex=0.5, font=22, hang=-1, main="HAC cluster dendogram with Cosine Similarity (Normalized")  
rect.hclust(groups\_C\_norm, k=2)



### Analysis and Results

### K-Means

Here are some results/observations with experiments around different cluster sizes:

* cluster-size=2

**SSEs**

Within cluster sum of squares by cluster is high:

[1] 174194.7 6447.5

This is an indication of high deviation between data-points and the centroid which we would ideally like to be lower. To explore k-means further, we could consider using the k-medoids/expectation-max or PAM algorithms.

**Data**

Most of the data-points were grouped into cluster-1 and this did not help to clearly determine the author for the disputed papers.

* cluster-size=7

**SSE**

SSEs look a lot better with increased cluster-size

Within cluster sum of squares by cluster:

[1] 0.00754175 0.03396400 0.06862076 0.00231200 0.02952307 0.00990520 0.02239410

**Data**

Disputed papers were placed in clusters - 2, 7, 3:

* Number of disputed papers in cluster-2 = 3
* Number of disputed papers in cluster-7 = 7
* Number of disputed papers in cluster-3 = 1

Cluster-7 that has the highest papers does not have sufficient majority of Hamilton/Madison papers to make a decision.

Overall, k-means does not seem like a good algorithm for document analysis use-cases.

### HAC algorithm

In comparison, seems like plotting and analyzing dendograms, seems a plausible means to realize the exercise. To a very large extent we can classify the disputed documents to the corresponding authors.

## Conclusions

With Hierarchical Agglomerative Clustering (HAC) techniques (and dendograms to analyze the results) we conclude by analyzing one disputed document dispt\_fed\_49.txt across:

* Eucledian

In plot ‘HAC cluster dendogram with Euclidean Similarity’, see document ‘dispt\_fed\_49.txt’ present in the first-cluster on the left and is associated by nodes/leafs that belong to Hamilton so, we can conclude it was written by author Hamilton with moderate confidence.

* Cosine

In plot ‘HAC cluster dendogram with Cosine Similarity’, see document ‘dispt\_fed\_49.txt’ belonging to a cluster towards the end. Again, the nodes/leafs around it are documents by author Hamilton.

* Cosine-Normalized

Likewise, in plot ‘HAC cluster dendogram with Cosine Similarity (Normalized)’ the surrounding nodes/leafs are related to author Hamilton.

In similar lines, we could extend the study to all disputed documents and hence classify them between the two authors.