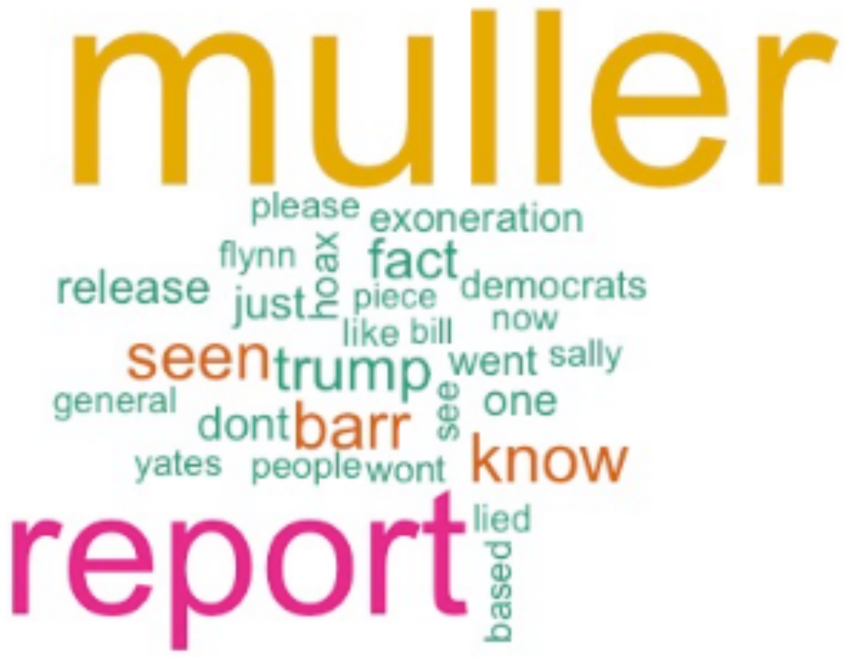
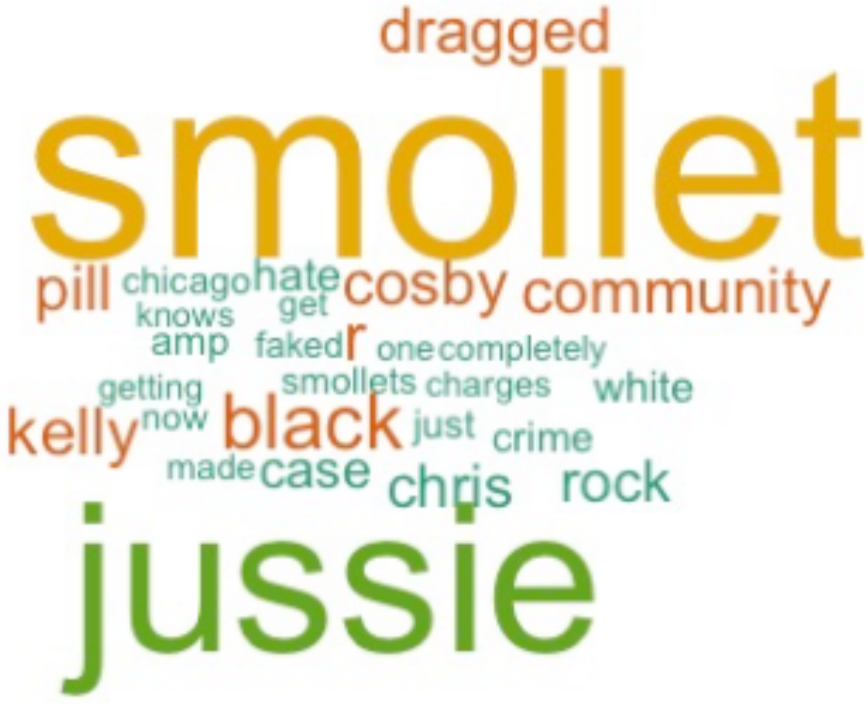
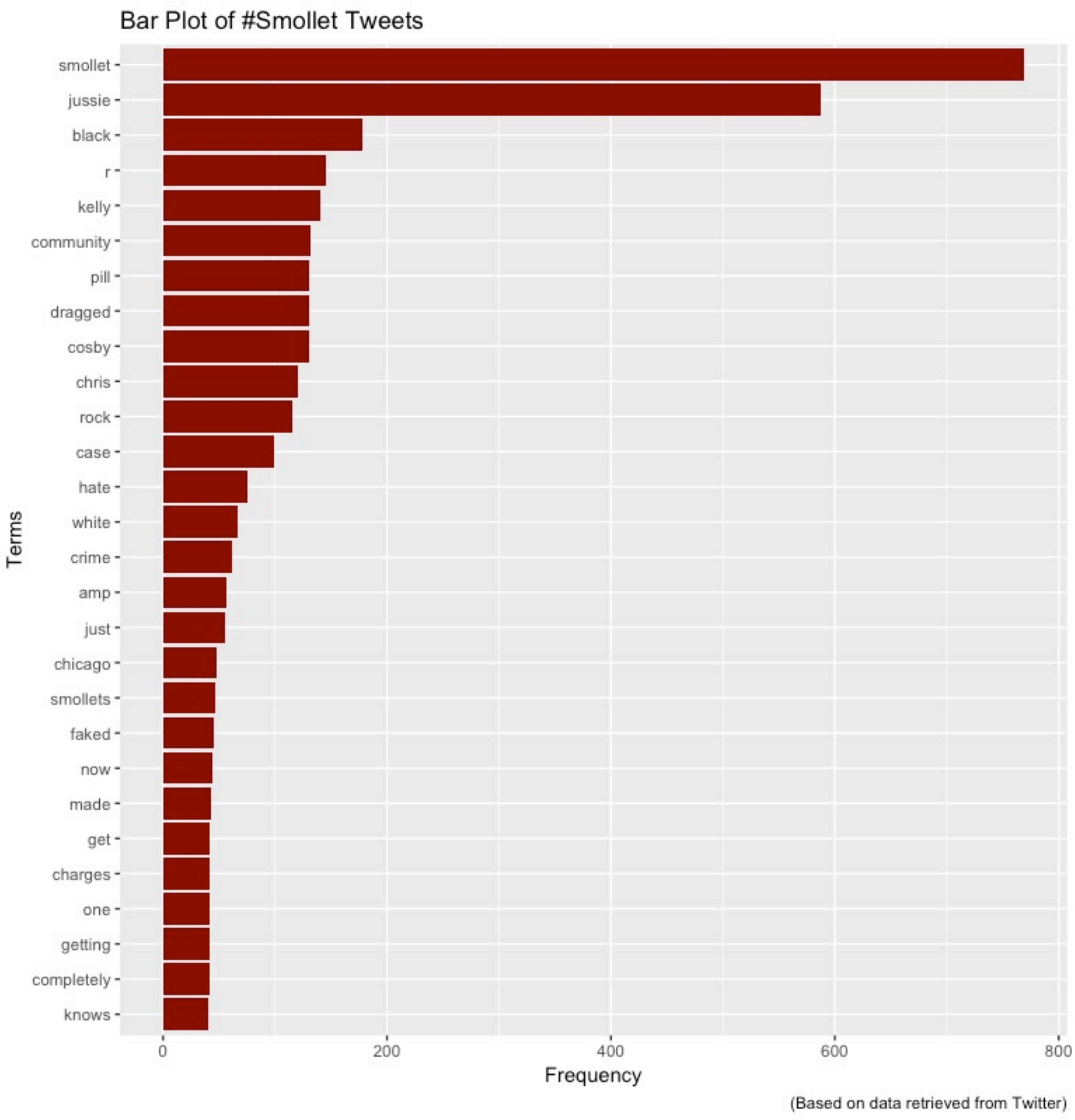

muller
please
exoneration
flynn
hoax
fact
release
just
piece
democrats
now
like bill
seen
trump
went
sally
general
see
one
dont
barr
know
yates
people
wont
lied
report
based

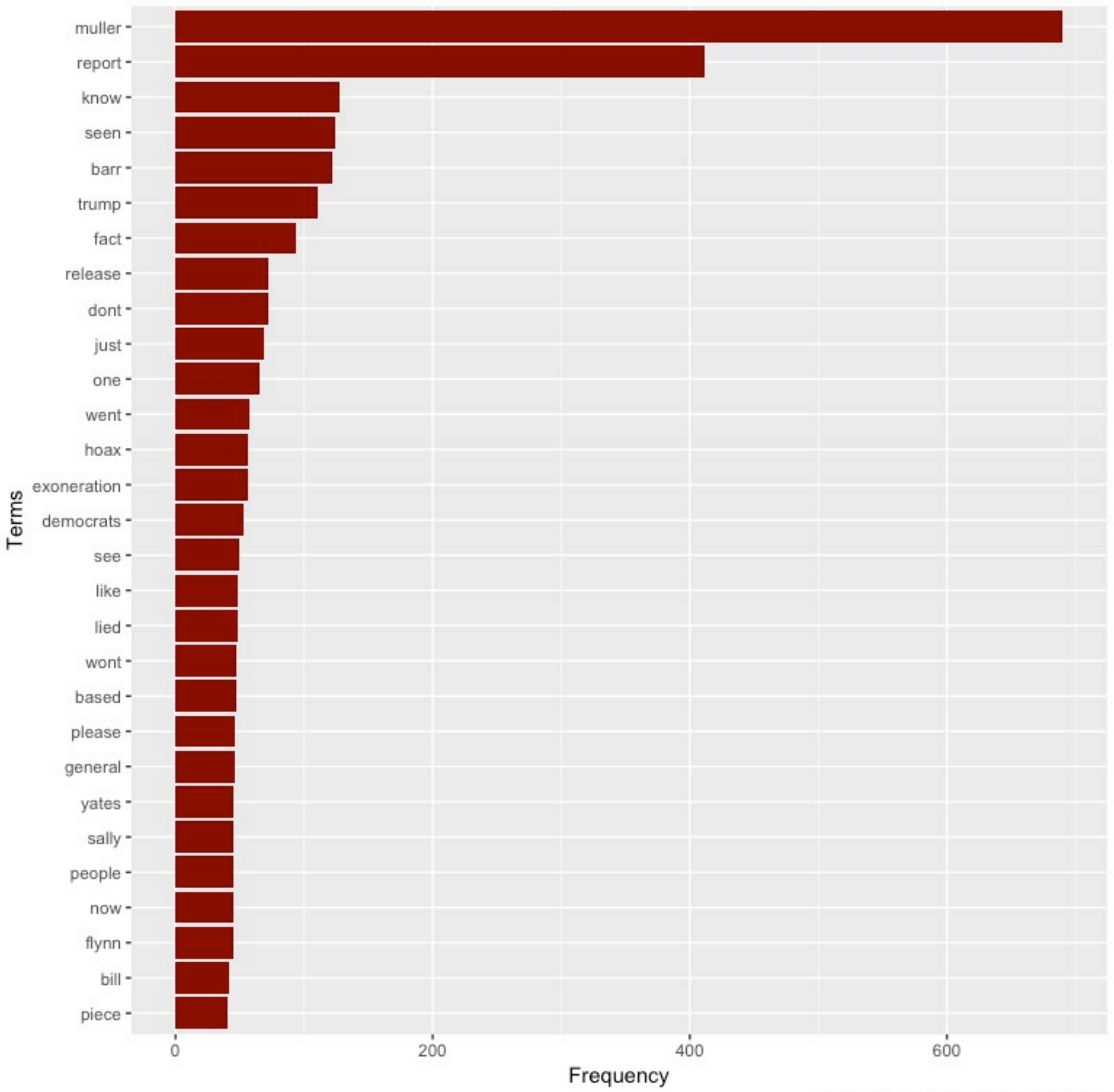

dragged
smollet
pill
chicago
hate
cosby
community
knows
get
amp
faked
r
one
completely
getting
smollets
charges
white
kelly
now
black
just
crime
made
case
chris
rock
jussie

```
> sort1.termdf_Muller
                   term freq
muller           muller  690
report           report  411
know               know  128
seen               seen  124
barr               barr  122
trump             trump  111
fact               fact   94
dont               dont   72
release         release   72
just               just   69
one                 one   65
went               went   58
hoax               hoax   56
exoneration exoneration   56
democrats     democrats   53
see                 see   49
lied               lied   48
like               like   48
based             based   47
wont               wont   47
please           please   46
general         general   46
flynn             flynn   45
sally             sally   45
yates             yates   45
now                 now   45
people           people   45
bill               bill   42
piece             piece   40
```

```
> sort1.termdf_Smollet
                 term freq
smollet       smollet  769
jussie         jussie  587
black           black  178
r                   r  145
kelly           kelly  140
community   community  132
cosby           cosby  131
dragged       dragged  131
pill             pill  131
chris           chris  121
rock             rock  116
case             case   99
hate             hate   75
white           white   67
crime           crime   62
amp               amp   57
just             just   55
chicago       chicago   48
smollets     smollets   47
faked           faked   45
now               now   44
made             made   43
charges       charges   42
get               get   42
getting       getting   41
```


Bar Plot of #Smollet Tweets
smollet
jussie
black
r
kelly
community
pill
dragged
cosby
chris
rock
case
hate
white
crime
amp
just
chicago
smollets
faked
now
made
get
charges
one
getting
completely
knows
Terms
0
200
400
600
800
Frequency
(Based on data retrieved from Twitter)

Bar Plot of #Muller Tweets



(Based on data retrieved from Twitter)

Correlations with Word Associations (Muller):

```
$exoneration
 smoky traded   room    nam player   rahm  maybe jussie   barr
  0.98   0.98   0.96   0.96   0.94   0.94   0.83   0.79   0.41

$hoax
flynn       know      sally      yates       fact    general
 0.89       0.89       0.89       0.89       0.88       0.88

based       lied       went     disgrace    killary   ruined
0.87          0.86         0.78        0.41            0.41 0.41

victim   russiagate   national   life          another      called
0.39         0.37          0.36        0.36          0.33          0.29
```

Correlations with Word Associations (Smollet):

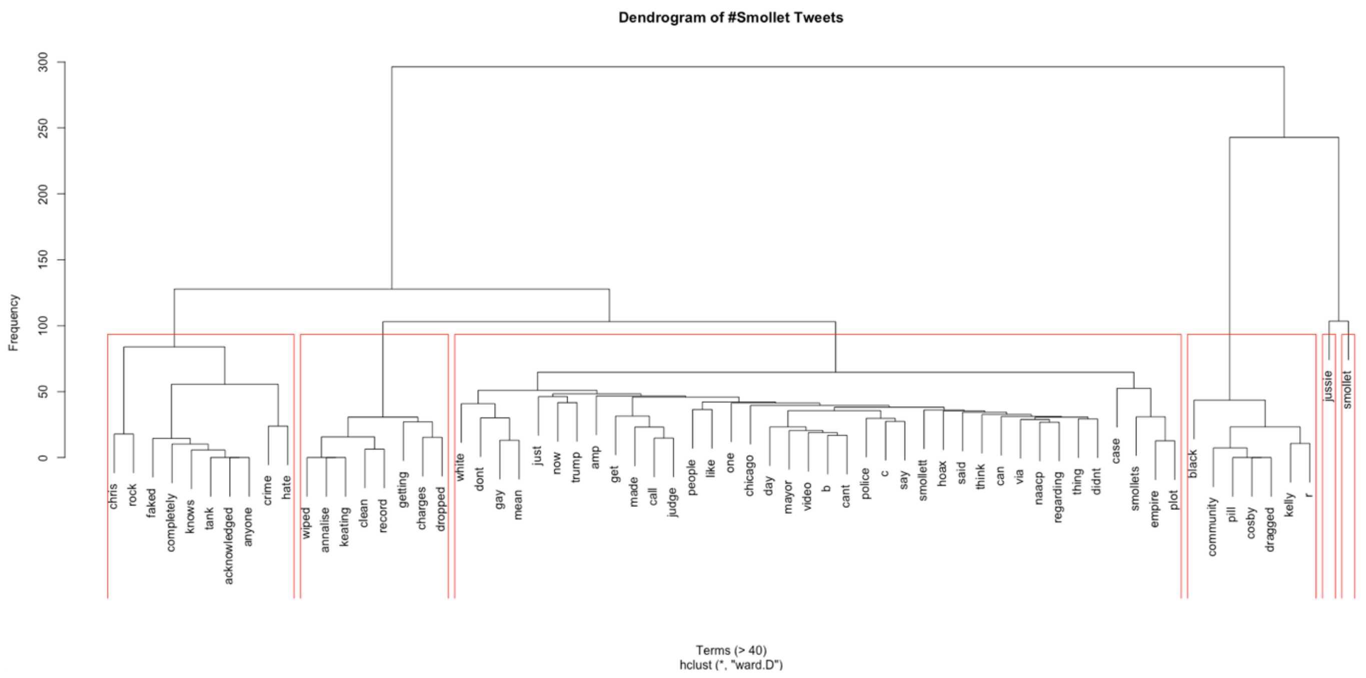
```
$dropped
 charges annalise  keating    wiped    clean   record  getting
    0.86     0.83     0.83     0.83     0.80     0.80     0.64

$hoax
diagram      droppe    overlaps      russia  scientific     showing
0.37            0.37        0.37       0.37          0.37          0.37

hateful      pulled      responsible      likely        main
0.33            0.33              0.30               0.29           0.29

andy  effecthoax        lgbt       ngo
0.29         0.29            0.29      0.29
```

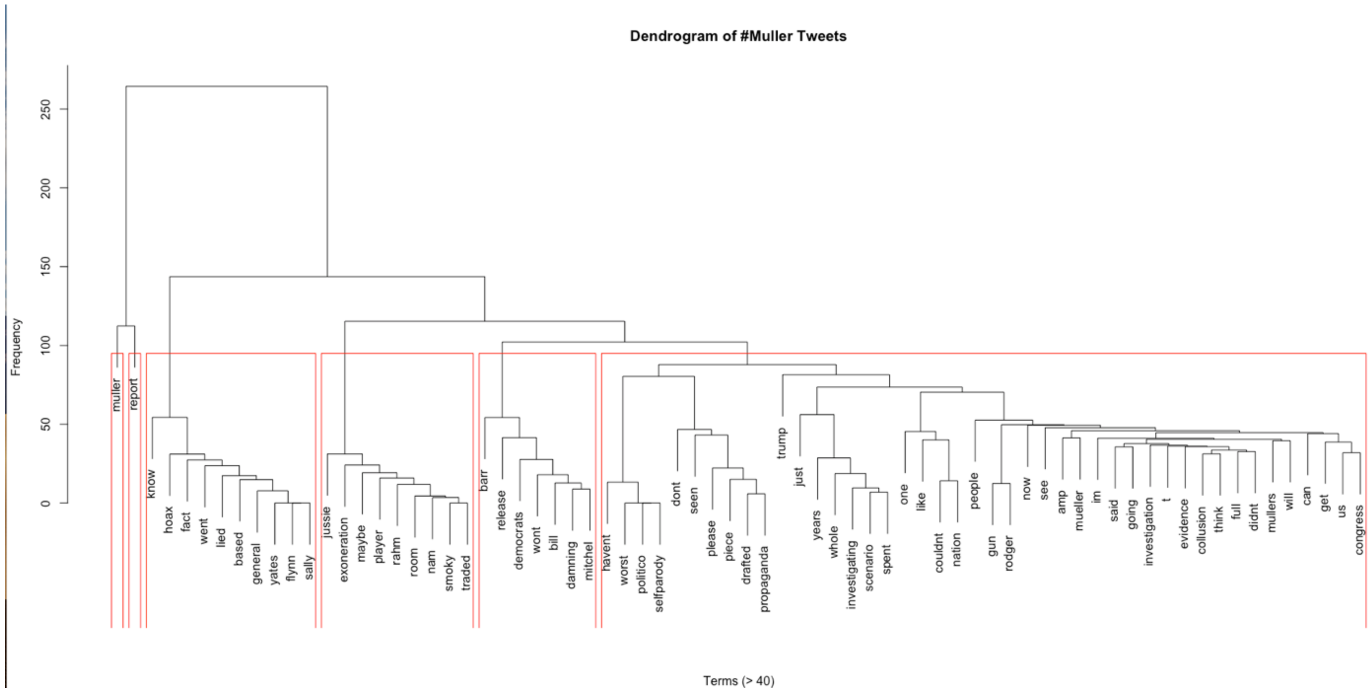
Clustering with K-MEANS algorithm (Muller):

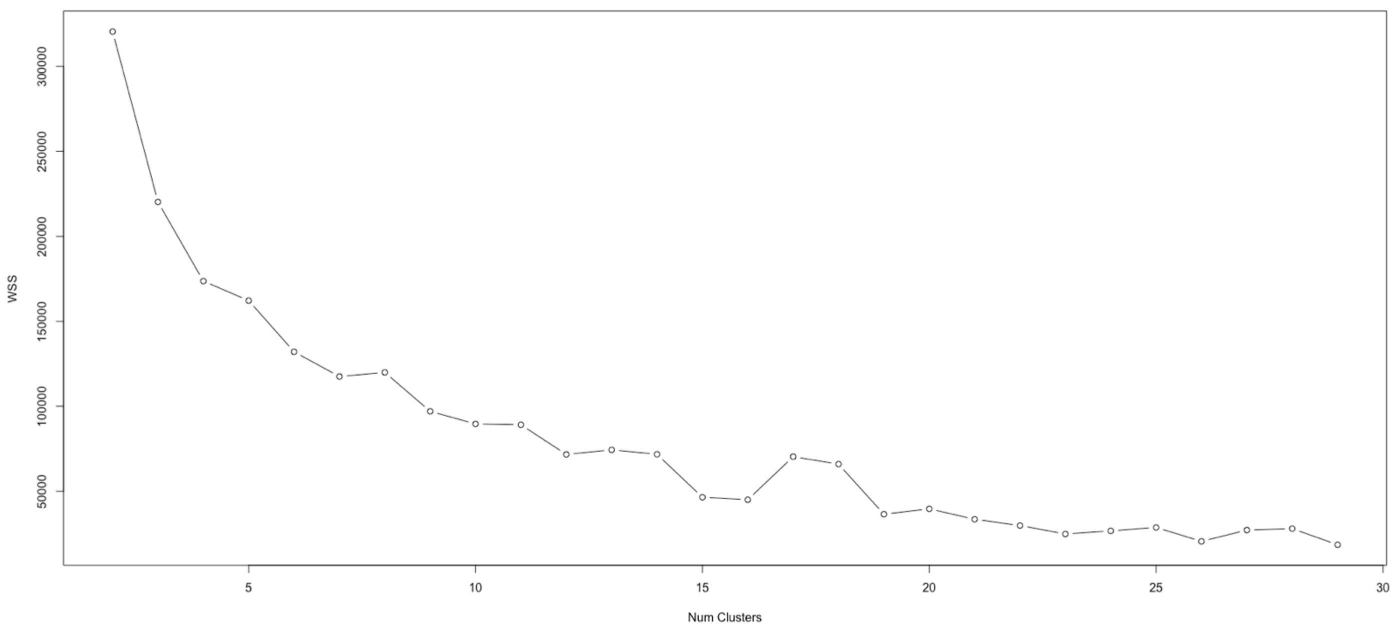
cluster 1: report muller seen dont trump
cluster 2: gun muller rodger went years
cluster 3: report release democrats barr bill
cluster 4: muller know fact trump barr

Clustering with K-MEANS algorithm (Smollet):

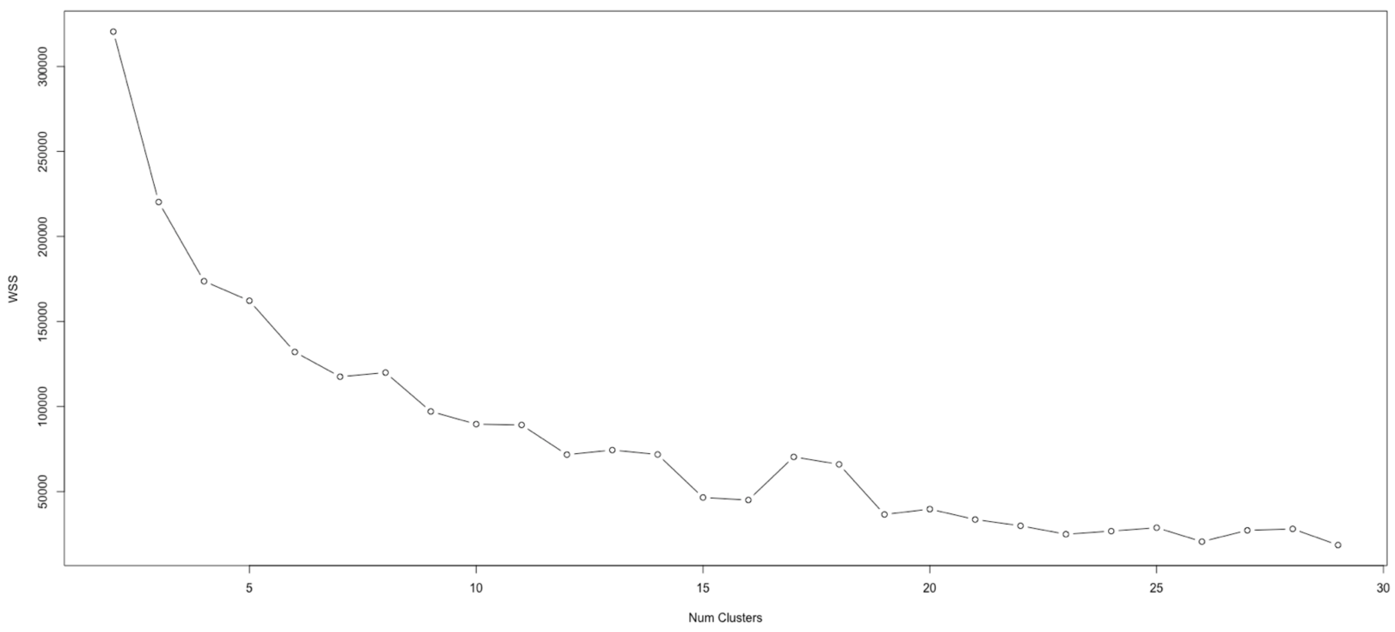
cluster 1: smollet jussie case white just
cluster 2: amp chicago jussie smollet police
cluster 3: jussie smollet black community cosby
cluster 4: chris rock smollet jussie faked


Dendrogram of #Smollet Tweets
Frequency
0
50
100
150
200
250
300
chris
rock
faked
completely
knows
tank
acknowledged
anyone
crime
hate
wiped
annalise
keating
clean
record
getting
charges
dropped
white
dont
gay
mean
just
now
trump
amp
get
made
call
judge
people
like
one
chicago
day
mayor
video
b
cant
police
c
say
smollett
hoax
said
think
can
via
naacp
regarding
thing
didnt
case
smolletts
empire
plot
black
community
pill
cosby
dragged
kelly
r
jussie
smollet
Terms (> 40)
hclust (*, "ward.D")

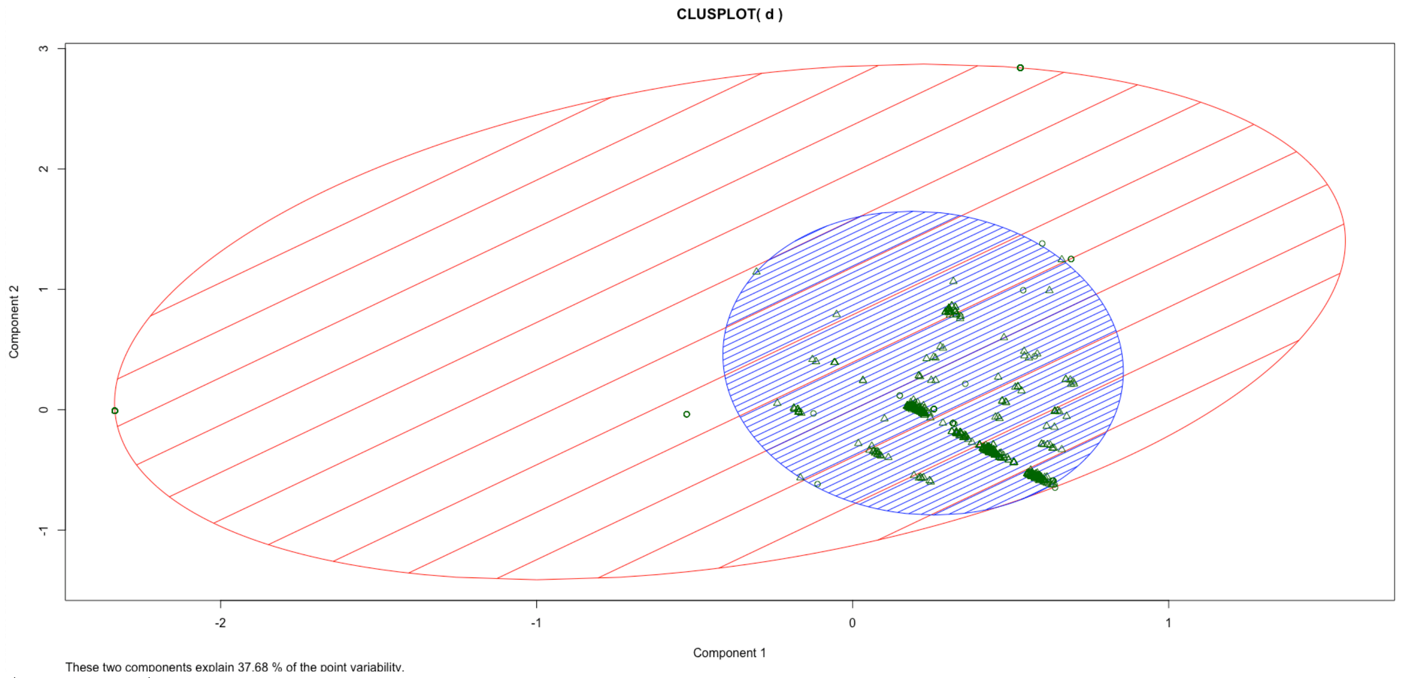

Dendrogram of #Muller Tweets
Frequency
0
50
100
150
200
250
muller
report
know
hoax
fact
went
lied
based
general
yates
flynn
sally
jussie
exoneration
maybe
player
rahm
room
nam
smoky
traded
barr
release
democrats
wont
bill
damning
mitchel
havent
worst
politico
selfparody
dont
seen
please
piece
drafted
propaganda
trump
just
years
whole
investigating
scenario
spent
one
like
couldnt
nation
people
gun
rodger
now
see
amp
mueller
im
said
going
investigation
t
evidence
collusion
think
full
didnt
mullers
will
can
get
us
congress
Terms (> 40)



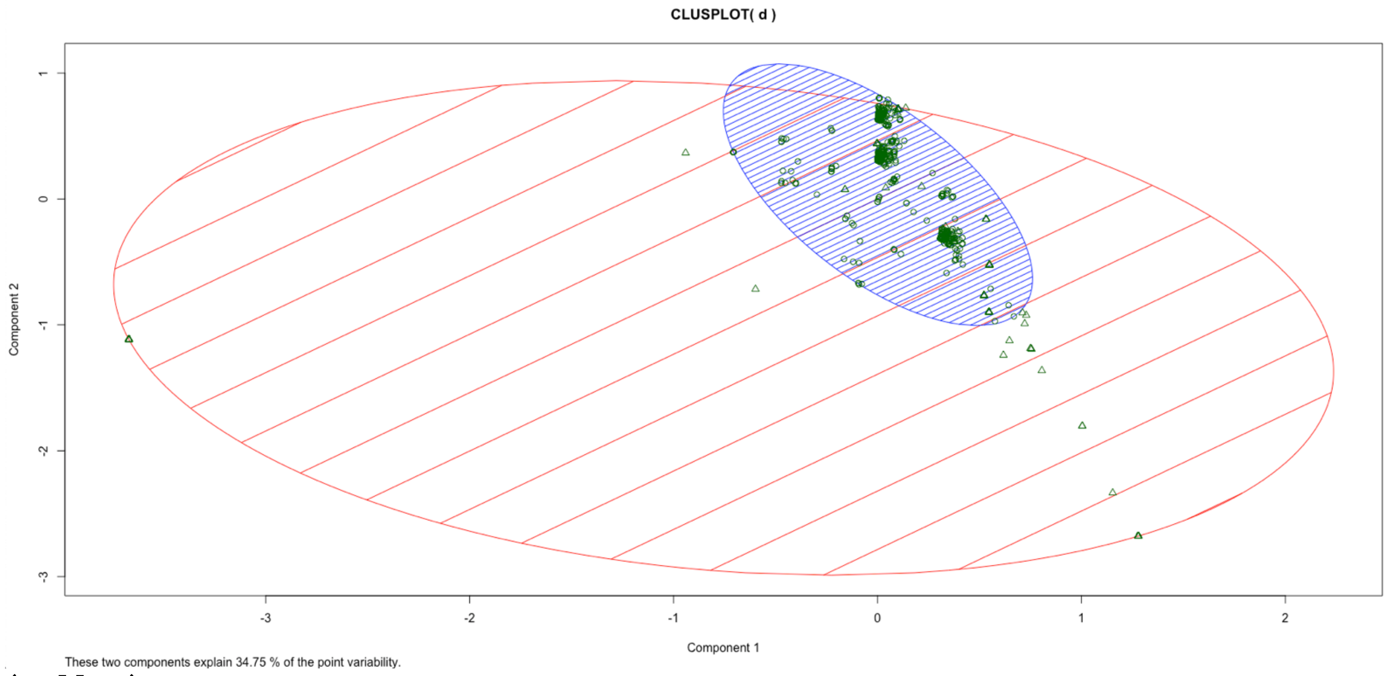
(smollet)



(muller)



(smollet)



(muller)

R CODE:

```
#### TWITTER SCRAPER ########


install.packages("twitteR")
install.packages("ROAuth")
install.packages("RCurl")
install.packages("RJSONIO")
install.packages("stringr")
install.packages("httr")
install.packages("tm")
install.packages("wordcloud")

library(twitteR)
library(ROAuth)
library(RCurl)
library(RJSONIO)
library(stringr)
library(httr)
library(tm)
library(ggplot2)
library(wordcloud)
library(cluster)
library(stats)

#Consumer API keys
api_key <- "VYzwdUqxvZuxjmjW6Ulo4bS6N" #(API key)
api_secret <- "eUhotijA79Xt0G7vgdG60ojQzHeHVCIu81MtZVbPnVLPG#####" #(API secret key,
##### for security)

#Access token & access token secret
access_token <- "1109130826619473921-uvCCki1AjTzCbGLzAlwXZasQKxmMED" #(Access token)
access_token_secret <- "G7gCQ3fNEQ7YVwuwyS6Wp6kukVkVpLfrJz4rFOrX#####" #(Access token
secret, ##### for security)
#connect to Twitter
setup_twitter_oauth(api_key, api_secret, access_token, access_token_secret)

tweets_Smollet <- searchTwitter("Smollet", n=1000, lang = "en", resultType =
"recent")
tweetsSmolletCopy <- tweets_Smollet
tweets <- tweets_Smollet
tweetsDF1 <- twListToDF(tweets)

#check info
dim(tweetsDF1)
str(tweetsDF1)
class(tweetsDF1)
length(tweetsDF1)
attributes(tweetsDF1)
writeLines(tweetsDF1$text[1])



# check content of several tweets
```

```
for (i in 25:30) {
  cat(paste("[[", i, "]]", sep = ""))
  writeLines(strwrap(tweets[[i]]$getText(), width = 73))
}

rm(myCorpus)

########################      BEGIN CORPUS HERE     ###############################

# create corpus called 'myCorpus'
myCorpus <- Corpus(VectorSource(tweetsDF1$text))


######### DOCUMENT INSPECTION FUNCTION  #############
quickCheck <- function (x) {
  for (i in c(25:30)){
    cat(paste0("[", i, "]"))
    writeLines(strwrap(as.character(myCorpus[[i]]), width = 60))
  }
}


############################# BEGIN CLEANING HERE  #############################

#### 1. convert myCorpus into lowercase
myCorpus <- tm_map(myCorpus, content_transformer(tolower))

#### 2. remove URL
removeURL <- function(x) gsub("http[^[:space:]]*", "", x)
myCorpus <- tm_map(myCorpus, content_transformer(removeURL))
quickCheck(myCorpus)

### 3. remove Alias
removeAlias <- function(y) gsub("@[[:alnum:][:punct:]]*", "", y)
myCorpus <- tm_map(myCorpus, content_transformer(removeAlias))
quickCheck(myCorpus)

### 4. remove anything other than English letters or space
removeNumPunct <- function(x) gsub("[^[:alpha:][:space:]]*", "", x)
myCorpus <- tm_map(myCorpus, content_transformer(removeNumPunct))
quickCheck(myCorpus)

### 5. remove stopwords
#add custom stopwords
mystopwords <- c(stopwords("english"),"rt","íí", "o", "d", "got")
#remove stopwords
myCorpus <- tm_map(myCorpus,removeWords,mystopwords)

### 6. remove extra whitespace
myCorpus <- tm_map(myCorpus, stripWhitespace)
quickCheck(myCorpus)



rm(myTDM)
```

```
####################        BUILD TERM DOCUMENT MATRIX          ####################

myTDM <- TermDocumentMatrix(myCorpus, control = list(wordLengths=c(1,Inf)))
myTDM

### Frequent Terms
findFreqTerms(myTDM, lowfreq = 40)
termFrequency <- rowSums(as.matrix(myTDM))
termFrequency <- subset(termFrequency, termFrequency>=40)


# create data frame from termFrequency
termdf <- data.frame(term= names(termFrequency), freq = termFrequency)
termdf_Smollet <- termdf


## sort frequent terms by order
attach(termdf_Smollet)
sort1.termdf_Smollet <- termdf_Smollet[order(freq, decreasing = TRUE), ]
sort1.termdf_Smollet

### plot term frequency dataframe w/ ggplot2
ggplot(termdf_Smollet, aes(x = reorder(term, freq), y = freq)) +
  geom_bar(stat = "identity", fill="darkred") + coord_flip() +
  labs(title = "Bar Plot of #Smollet Tweets", caption = "(Based on data retrieved
from Twitter)") + ylab("Frequency") + xlab("Terms")


### create a Word Cloud
set.seed(142)
wordcloud(words = names(termFrequency), freq = termFrequency, max.words = 40, scale =
c(6, .5), colors = brewer.pal(6, "Dark2"))

### create a cluster DENDROGRAM
# simplify by removing sparse terms
myTDMsparse <- removeSparseTerms(myTDM, 0.98)
myTDMsparse

# transform to Matrix without sparse terms
m <- as.matrix(myTDMsparse)  # from LIST to MATRIX
is.matrix(m)


set.seed(122)
# cluster centers
distMatrix <- dist(scale(m))

fit <- hclust(distMatrix, method = "ward.D")
# plot DENDROGRAM
plot(fit, main = "Dendrogram of #Smollet Tweets", xlab = "Terms (> 40)", ylab =
"Frequency")
rect.hclust(fit, k = 6)


###  find correlations with Word Associations
```

```
findAssocs(myTDM, 'dropped', 0.25)
findAssocs(myTDM, 'hoax', 0.25)


###  Clustering with K-MEANS algorithm

m2 <- t(m)    #transpose matrix 'm' to 'm2'
set.seed(169)
k <- 4  #choose number of K
kmeansResult <- kmeans(m2, k)  #use KMEANS function w/ 'k' number of centroids
round(kmeansResult$centers, digits = 2)  #print out data

## sort top 5 terms by centroid
for (i in 1:k) {
  cat(paste("cluster ", i, ": ", sep = ""))
  s <- sort(kmeansResult$centers[i,], decreasing = T)
  cat(names(s)[1:5], "\n")
}


#### K-MEANS Cluster Plot
d <- dist(m2) # create distance vector with transposed matrix: m2
kfit <- kmeans(d, 2, nstart = 100)
clusplot(d, kfit$cluster, diss = T, color = T, shade = T, labels = 0, lines = 0)


## WSS elbow method (w/ info, re-do K Means if necessary)
wss <- 2:29
for (i in 2:29) wss[i] <- sum(kmeans(d, centers = i, nstart = 25)$withinss)
plot(2:29, wss[2:29], type = "b", xlab="Num Clusters", ylab= "WSS")
```