Problem Set 8

Clustering Part 1

[YOUR NAME]

Due Date: 2023-11-26

Getting Set Up

Open `RStudio` and create a new RMarkDown file (`.Rmd`) by going to `File -> New File -> R Markdown...` . Accept defaults and save this file as `[LAST NAME]_ps8.Rmd` to your `code` folder.

Copy and paste the contents of this file into your `[LAST NAME]_ps8.Rmd` file. Then change the `author: [YOUR NAME]` (line 4) to your name.

All of the following questions should be answered in this `.Rmd` file. There are code chunks with incomplete code that need to be filled in.

This problem set is worth 10 total points, plus two extra credit points. The point values for each question are indicated in brackets below. To receive full credit, you must both have the correct code **and include a comment describing what each line does**. In addition, some questions ask you to provide a written response in addition to the code. Furthermore, some of the code chunks are totally empty, requiring you to try writing the code from scratch. Make sure to comment each line, explaining what it is doing!

You are free to rely on whatever resources you need to complete this problem set, including lecture notes, lecture presentations, Google, your classmates…you name it. However, the final submission must be complete by you. There are no group assignments. To submit, compiled the completed problem set and upload the PDF file to Brightspace by midnight on 2023/11/26.

Good luck!

ChatGPT Link [Optional]

*Copy the link to ChatGPT you used here: _____.

Question 0

Require `tidyverse` and `tidytext` (for calculating AUC), and load the `Trump_tweet_words.Rds` (https://github.com/jbisbee1/DS1000_F2023/blob/main/Lectures/8_Clustering_NLP/data/Trump_tweet_words.Rds?raw=true) data to an object called `tweet_words` . (Tip: use the `read_rds()` function with the link to the raw data.)

Also, load the `Trumptweets.Rds` (https://github.com/jbisbee1/DS1000_F2023/blob/main/Lectures/8_Clustering_NLP/data/Trumptweets.Rds?raw=true) data to an object called `tweets` .

```
require(tidyverse)
```

```
## Loading required package: tidyverse
```

```
## Warning: package 'tidyverse' was built under R version 4.2.3
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
## Warning: package 'tibble' was built under R version 4.2.3
```

```
## Warning: package 'tidyr' was built under R version 4.2.3
```

```
## Warning: package 'readr' was built under R version 4.2.3
```

```
## Warning: package 'purrr' was built under R version 4.2.3
```

```
## Warning: package 'dplyr' was built under R version 4.2.3
```

```
## Warning: package 'forcats' was built under R version 4.2.3
```

```
## Warning: package 'lubridate' was built under R version 4.2.3
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.3     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ ggplot2   3.4.4     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.0
## ✓ purrr     1.0.2
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
e errors
```

```
require(tidytext)
```

```
## Loading required package: tidytext
```

```
tweet_words <- read_rds(file="https://github.com/jbisbee1/DS1000_F2023/blob/main/Lectures/8_Clus
tering_NLP/data/Trump_tweet_words.Rds?raw=true")

tweets <- read_rds(file="https://github.com/jbisbee1/DS1000_F2023/blob/main/Lectures/8_Clusterin
g_NLP/data/Trumptweets.Rds?raw=true")
```

Question 1 [1 point + 2 EC]

Using the `tweet_words` object, calculate the most frequently used word by year.

a. Which is Trump's most commonly used word in 2010 and how often did he use it? [1 point]

b. EXTRA CREDIT: can you determine what this word means, using the `tweets` object to see it in context? [1 point] Based on this analysis, do you think we should drop this word? Why? [1 point] **HINT** get the list of tweet IDs (`document`) from the `tweet_words` object, then filter the `tweets` object based on the `id` column. (NB: `document` in the `tweet_words` object is the same as the `id` column from the `tweets` object.)

```
# INSERT CODE HERE
tweet_words %>%
  count(Tweeting.year,word) %>%
  group_by(Tweeting.year) %>%
  arrange(desc(n)) %>%
  slice(1)
```

```
## # A tibble: 13 × 3
## # Groups:   Tweeting.year [13]
##    Tweeting.year word          n
##    <fct>         <chr>     <int>
##  1 2009          donald       42
##  2 2010          pm           46
##  3 2011          cont         71
##  4 2012          cont        349
##  5 2013          trump       788
##  6 2014          trump      1067
##  7 2015          trump      1620
##  8 2016          hillary     459
##  9 2017          amp         470
## 10 2018          amp         524
## 11 2019          president  1086
## 12 2020          president  1266
## 13 2021          georgia      23
```

```
ids <- tweet_words %>%
  filter(word == 'pm' & Tweeting.year == 2010)

tweets %>%
  filter(id %in% ids$document) %>%
  select(content)
```

```
## # A tibble: 41 × 1
##    content
##    <chr>
##  1 Celebrity Apprentice returns to NBC, Sunday, 3/14, 9-11PM ET/PT. Outstanding…
##  2 From Donald Trump: “I’m so proud of my wife Melania and the launch of her ne…
##  3 The Celebrity Apprentice has a two-hour premiere this Sunday, March 14th, at…
##  4 The new season of the Celebrity Apprentice is off to a great start-- last ni…
##  5 Tune in to The Marriage Ref onThursday night at 10 p.m. on NBC--I'm on the p…
##  6 To put on your calendar for May: Miss USA 2010, live from Las Vegas on May 1…
##  7 Melania and I will be appearing on Larry King Live tonight, 9 p.m. on CNN. B…
##  8 Melania will be on QVC tomorrow night at 9 p.m. ET to introduce her beautifu…
##  9 Be sure to look for my beautiful wife Melania Trump tonight on QVC at 9 pm E…
## 10 Looking forward to the 2010 Miss USA Pageant, Sunday May 16 on NBC, 7 p.m. E…
## # ℹ 31 more rows
```

- Trump’s most commonly used word in 2010 is “pm”, which he used 47 times. EC: Looking at the content of the tweets from 2010 that contain the word “pm”, it seems clear that Trump was promoting his Celebrity Apprentice TV show, which aired in the evenings.

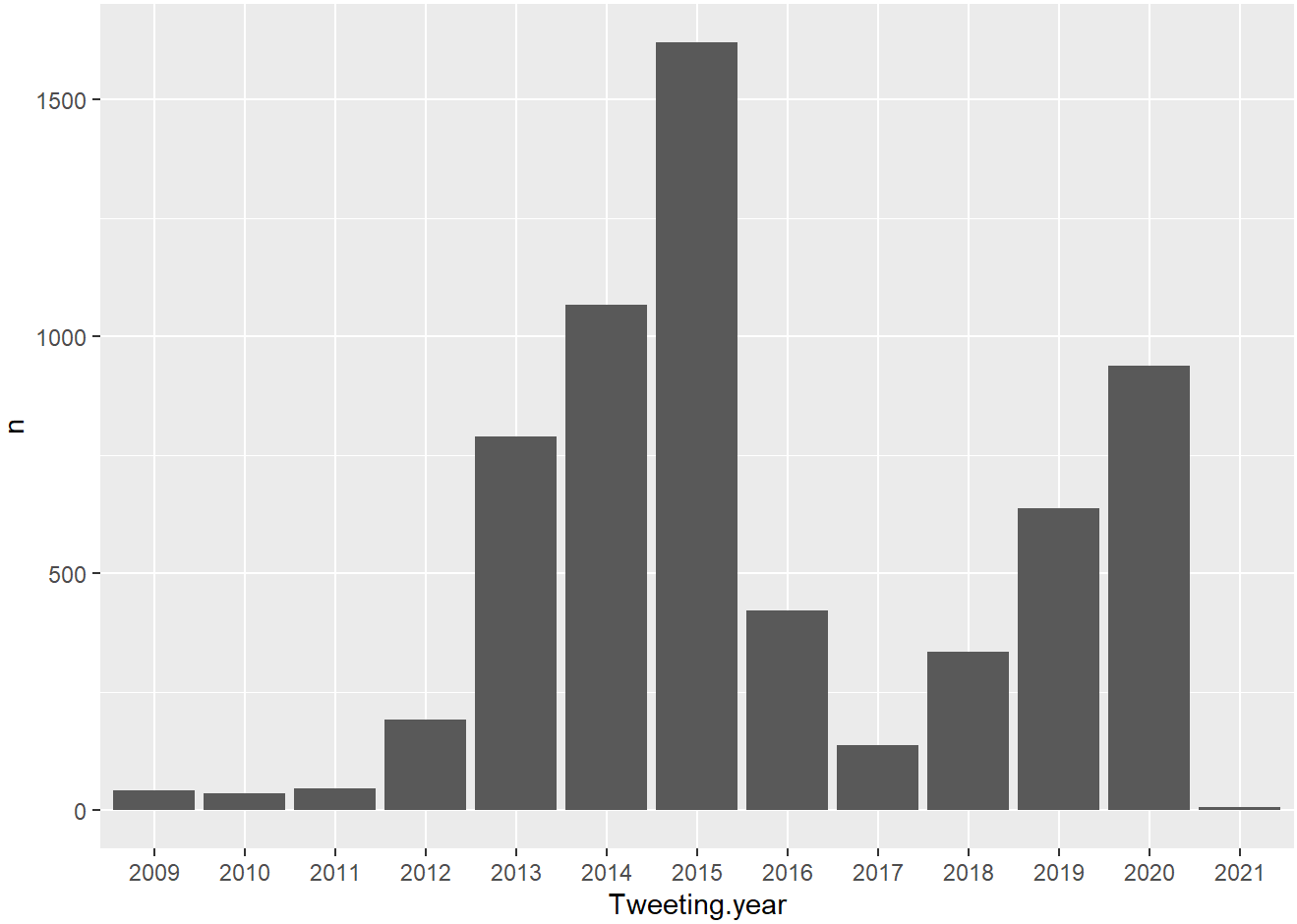
Question 2 [4 points]

a. Plot the total number of times the word “trump” is used each year. [1 point]

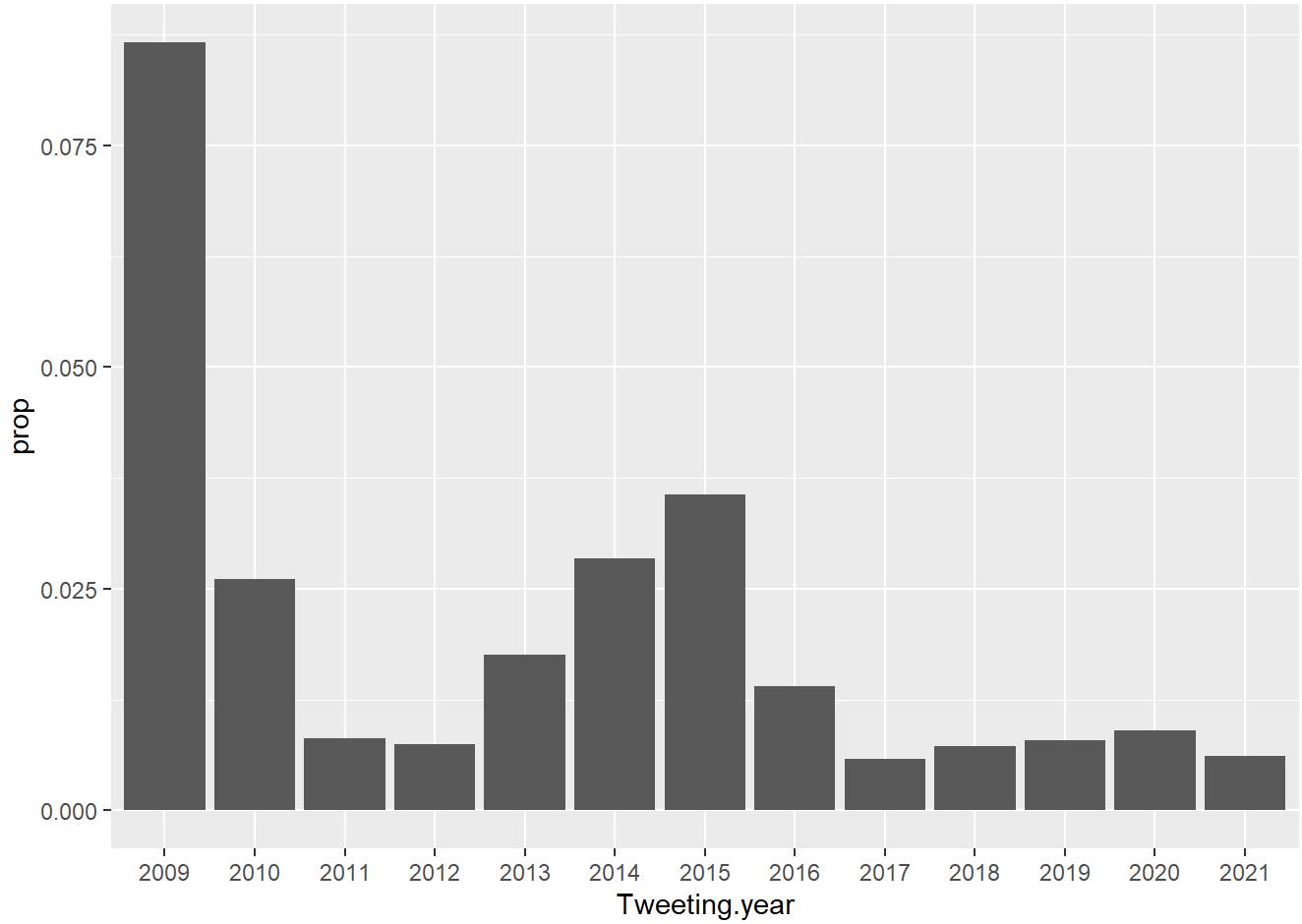
b. Plot the proportion of times the word “trump” is used each year. Make sure to justify your choice of `geom_...()` ! [1 point]

c. Why are these plots so different? Which measure is better? Why? [2 points]

```
# INSERT CODE HERE
# a.
tweet_words %>%
  count(Tweeting.year,word) %>%
  filter(word == 'trump') %>%
  ggplot(aes(x = Tweeting.year,y = n)) +
  geom_bar(stat = 'identity')
```




```
# b.
tweet_words %>%
  count(Tweeting.year,word) %>%
  group_by(Tweeting.year) %>%
  mutate(totWords = sum(n)) %>%
  ungroup() %>%
  mutate(prop = n / totWords) %>%
  filter(word == 'trump') %>%
  ggplot(aes(x = Tweeting.year,y = prop)) +
  geom_bar(stat = 'identity')
```



- These plots look different because Trump tweeted much more frequently in 2012 - 2016 than he did in 2009 - 2011, but he tweeted about himself as a fraction of total tweets much more frequently in 2009 than any other year. This comparison reveals that the proportion of total tweets is a better measure of Trump's behavior, since it accurately measures the quantity of interest. If we relied on the total tweets, we would conclude he was much more self-obsessed in 2015 than any other year, but this conclusion conflates how much he tweets in general with how much he tweets about himself.

Question 3 [3 points]

We want to only look at tweets written during Trump's final year as president until he was kicked off Twitter (January 1st, 2020 through January 8th, 2021), and are interested if there are patterns in what he talks about.

Prepare the data for topic modeling via k-means clustering, filtering to the final year of his presidency and using `document` as the document.

a. Create a document-term matrix (`dtm`), dropping any words that appear fewer than 20 times total. [1 point]

b. Calculate the TF-IDF using the appropriate function from the `tidytext` package. [1 point]

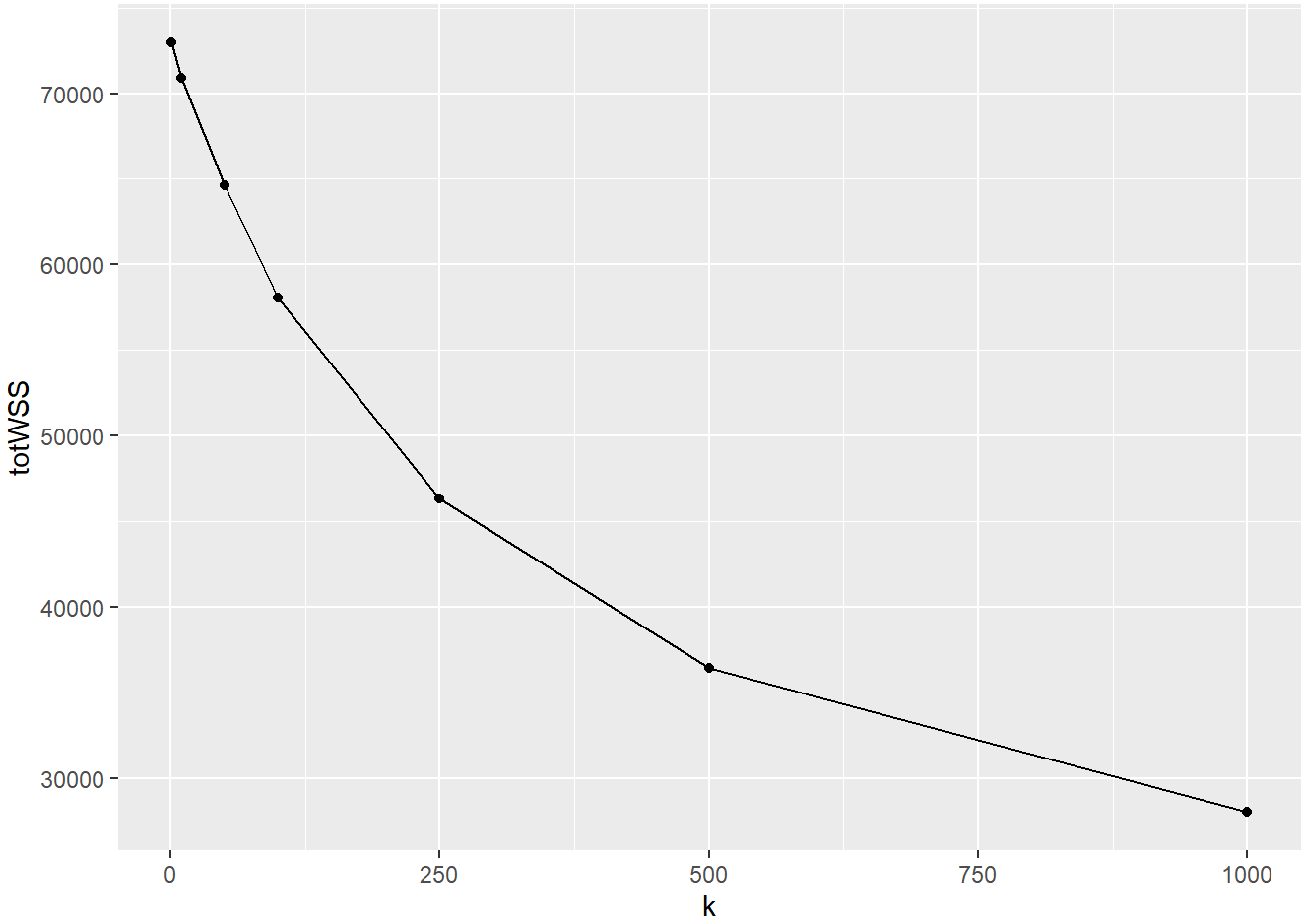
c. Cast the DTM to wide format using the `cast_dtm()` function. [1 point]

```
# INSERT CODE HERE
dtm <- tweet_words %>%
  filter(Tweeting.date > as.Date('2020-01-01') & Tweeting.date < as.Date('2021-01-08')) %>%
  count(document,word) %>%
  group_by(word) %>%
  mutate(tot_n = sum(n)) %>%
  ungroup() %>%
  filter(tot_n >20)

#b.
dtm.tfidf <- bind_tf_idf(tbl = dtm, term = word, document = document, n = n)

#c.
castdtm <- cast_dtm(data = dtm.tfidf, document = document, term = word, value = tf_idf)
```

Question 4 [1 point]

Determine the optimal number of clusters / centers / topics / k by creating and manually inspecting an elbow plot. To save time, only examine the following sizes: `c(1,10,50,100,250,500,1000)` (this will still take a little while to run so be patient!). What value would you choose? [1 point]

```
set.seed(42) # Set common seed to ensure reproducability
#INSERT CODE HERE
totWSS <- NULL
for(k in c(1,10,50,100,250,500,1000)) {
  km_out <- kmeans(castdtm,
                centers = k,
                nstart = 5)

  totWSS <- data.frame(totWSS = km_out$tot.withinss,
                       k = k) %>%
    bind_rows(totWSS)
}

totWSS %>%
  ggplot(aes(x = k,y = totWSS)) +
  geom_point() +
  geom_line()
```



- I would choose somewhere between 250 and 500 clusters, based on this elbow plot.

Question 5 [2 points]

Re-run the k-means analysis using the number of clustered identified above and then `tidy()` the output.

a. Which are the top 5 most popular topics for Donald Trump in this period? [1 point]

b. Plot the top 10 highest scoring words for each of the top 5 most popular topics. What is each "about"? [1 point]

```
# INSERT CODE HERE
require(tidymodels)
```

```
## Loading required package: tidymodels
```

```
## ── Attaching packages ─────────────────────────────────────── tidymodels 1.0.0 ──
```

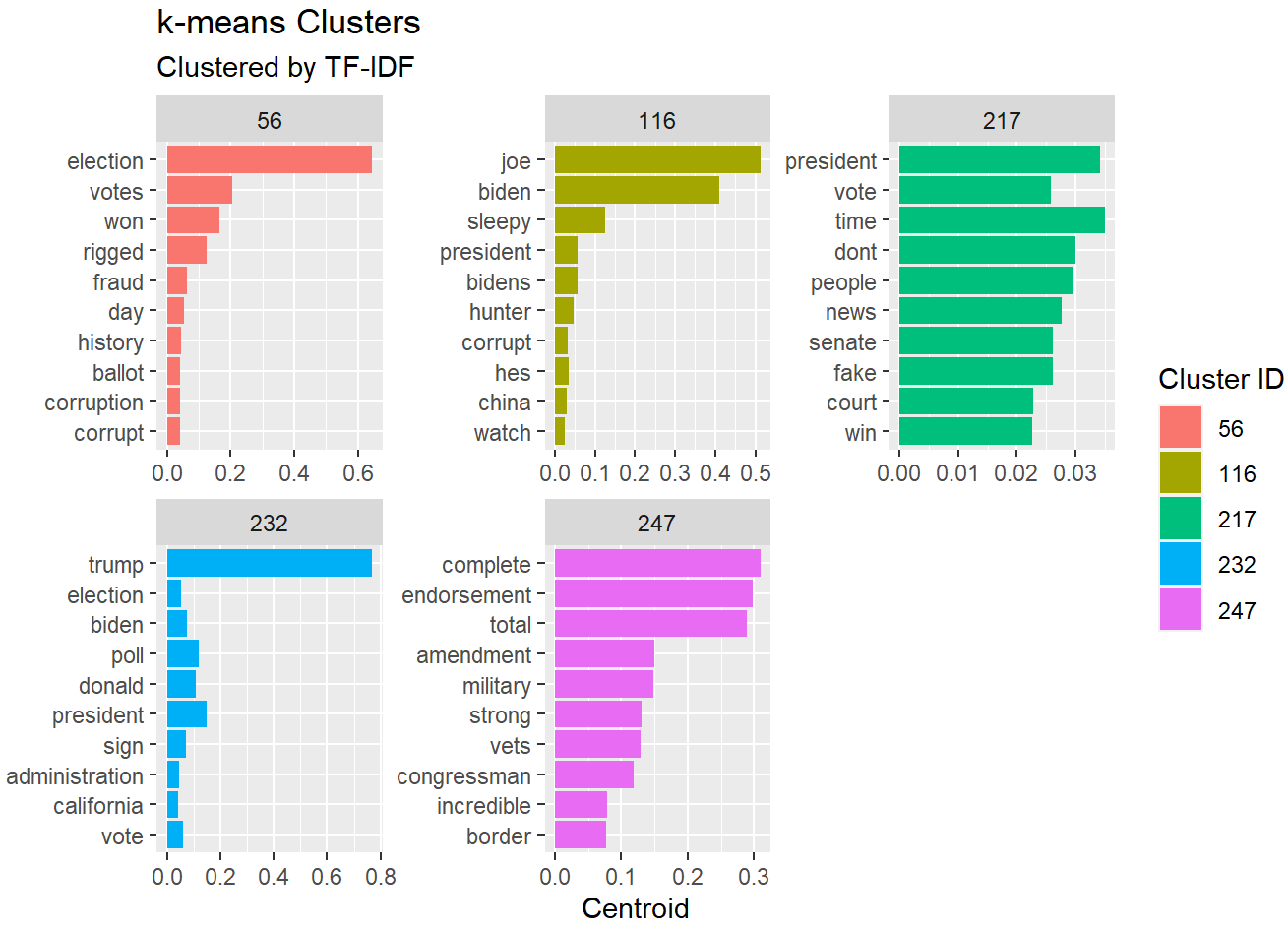
```
## ✓ broom        1.0.5     ✓ rsample      1.1.1
## ✓ dials        1.1.0     ✓ tune         1.0.1
## ✓ infer        1.0.4     ✓ workflows    1.1.2
## ✓ modeldata    1.0.1     ✓ workflowsets 1.0.0
## ✓ parsnip      1.0.3     ✓ yardstick    1.1.0
## ✓ recipes      1.0.3
```

```
## ── Conflicts ─────────────────────────────────────── tidymodels_conflicts() ──
## ✕ scales::discard() masks purrr::discard()
## ✕ dplyr::filter()   masks stats::filter()
## ✕ recipes::fixed()  masks stringr::fixed()
## ✕ dplyr::lag()      masks stats::lag()
## ✕ yardstick::spec() masks readr::spec()
## ✕ recipes::step()   masks stats::step()
## • Use suppressPackageStartupMessages() to eliminate package startup messages
```

```
km_out <- kmeans(castdtm,
                 centers = 250,
                 nstart = 25)

# For students who can't load tidymodels
# km_out_tidy <- as_tibble(km_out$centers) %>%
#   mutate(size = km_out$size,
#          withinss = km_out$withinss,
#          cluster = factor(row_number())) %>%
#   gather(word,mean_tfidf,-size,-cluster,-withinss)

km_out_tidy <- tidy(km_out) %>%
  gather(word,mean_tfidf,-size,-cluster,-withinss) %>%
  mutate(mean_tfidf = as.numeric(mean_tfidf))

#a.
(tops <- km_out_tidy %>%
  select(size,withinss,cluster) %>%
  distinct() %>%
  arrange(desc(size)) %>%
    slice(1:5))
```

```
## # A tibble: 5 × 3
##    size withinss cluster
##   <int>    <dbl> <fct>
## 1  2678   12893. 217
## 2   367    1053. 116
## 3   253     522. 247
## 4   232     902. 232
## 5   200     729. 56
```

```
#b.
km_out_tidy %>%
  filter(cluster %in% tops$cluster) %>%
  group_by(cluster) %>%
  arrange(-mean_tfidf) %>%
  slice(1:10) %>%
  ggplot(aes(x = mean_tfidf,y = reorder(word,mean_tfidf),
             fill = factor(cluster))) +
  geom_bar(stat = 'identity') +
  facet_wrap(~cluster,scales = 'free') +
  labs(title = 'k-means Clusters',
       subtitle = 'Clustered by TF-IDF',
       x = 'Centroid',
       y = NULL,
       fill = 'Cluster ID')
```



- There are two topics that are clearly about Trump's favorite topics in 2020: his claims about a rigged election, and his opponent Joe Biden. The other topics are less distinct, but nevertheless, appear to be about the military and two different subtopics pertaining to voting.