M4A_Project

Ross Pingatore

10/30/2020

```
library(readxl)
medicare_for_all_1 <- read_excel("Data/medicare_for_all_1.xlsx")
single_payer_system_1 <- read_excel("Data/single_payer_system_1.xlsx")
universal_health_care_1 <- read_excel("Data/universal_health_care_1.xlsx")
medicare_for_all_2 <- read_excel("Data/medicare_for_all_2.xlsx")
single_payer_system_2 <- read_excel("Data/single_payer_system_2.xlsx")
universal_health_care_2 <- read_excel("Data/universal_health_care_2.xlsx")

data_names <- c(medicare_for_all_1,single_payer_system_1,universal_health_care_1,medicare_for_all_2,sing
```

```
suppressPackageStartupMessages(library(tidyverse))
merged_data <- bind_rows(medicare_for_all_1,single_payer_system_1,universal_health_care_1,medicare_for_a

write_excel_csv(merged_data, 'Data/merged_data.xlsx')
view(merged_data)
```

```
unique(merged_data$Language)
```

```
##  [1] "en"  "und" "ja"  "tl"  "es"  "ro"  "zh"  "sv"  "nl"  "pt"
```

```
library(tidytext)
library(tm)
```

```
## Loading required package: NLP
```

```
##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
library(dplyr)
library(NLP)
#https://stackoverflow.com/questions/36824296/r-remove-specific-word-in-a-txte-like-the-this

corpus <- Corpus(VectorSource(merged_data$Text))


corpus <- tm_map(corpus, removePunctuation)
```

```
## Warning in tm_map.SimpleCorpus(corpus, removePunctuation): transformation drops
## documents
```

```
corpus <- tm_map(corpus, removeNumbers)
```

```
## Warning in tm_map.SimpleCorpus(corpus, removeNumbers): transformation drops
## documents
```

```
corpus <- tm_map(corpus, tolower)
```

```
## Warning in tm_map.SimpleCorpus(corpus, tolower): transformation drops documents
```

```
corpus <- tm_map(corpus, removeWords, stopwords("english"))
```

```
## Warning in tm_map.SimpleCorpus(corpus, removeWords, stopwords("english")):
## transformation drops documents
```

```
corpus <- tm_map(corpus, stripWhitespace)
```

```
## Warning in tm_map.SimpleCorpus(corpus, stripWhitespace): transformation drops
## documents
```

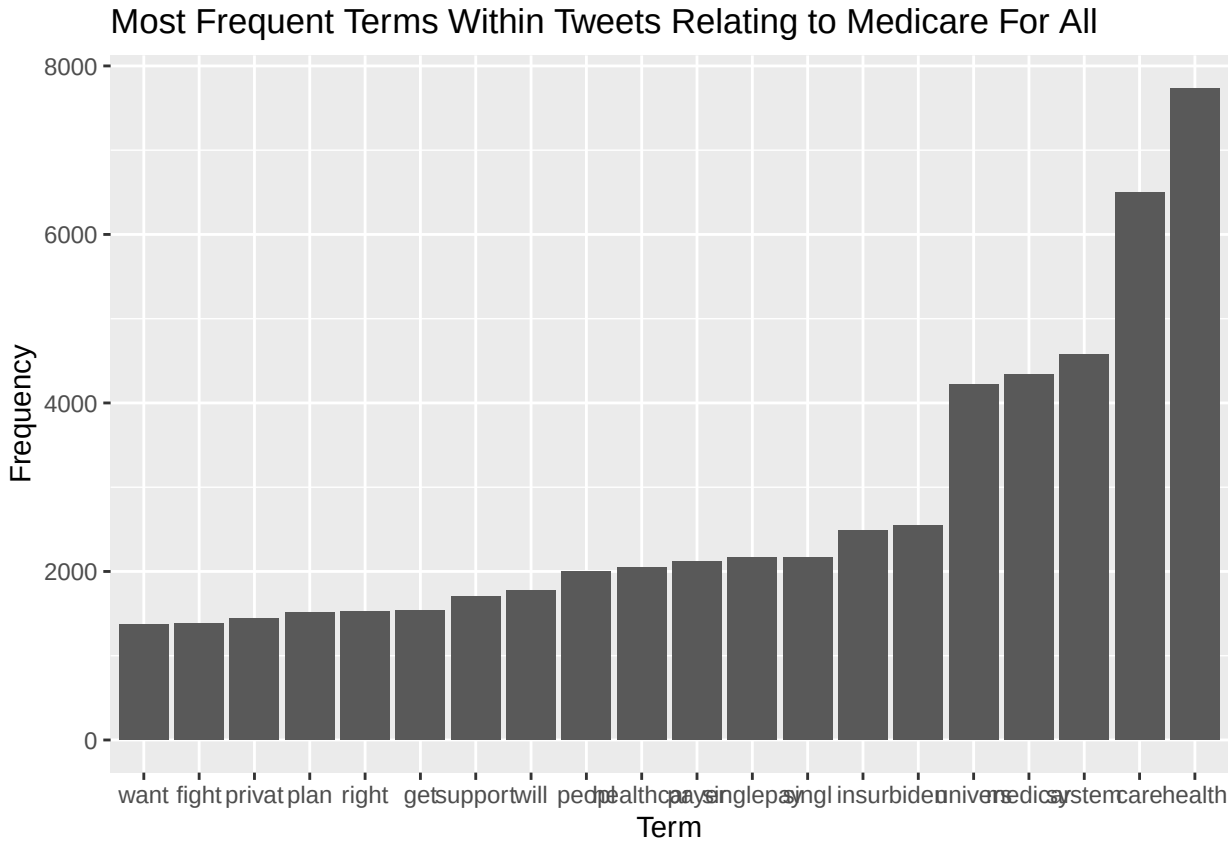
```
corpus <- tm_map(corpus, stemDocument)
```

```
## Warning in tm_map.SimpleCorpus(corpus, stemDocument): transformation drops
## documents
```

```
doc_matrix <- TermDocumentMatrix(corpus)
doc_matrix_m <- as.matrix(doc_matrix)
doc_matrix_val <- sort(rowSums(doc_matrix_m), decreasing = T)
doc_matrix_df <- data.frame(word = names(doc_matrix_val), freq = doc_matrix_val)

top_20 <- head(doc_matrix_df, 20)
top_20
```

```
##                word freq
## health       health 7738
## care           care 6495
## system       system 4576
## medicar     medicar 4335
## univers     univers 4215
## biden         biden 2540
## insur         insur 2482
## singl         singl 2163
## singlepay singlepay 2160
## payer         payer 2118
## healthcar healthcar 2050
## peopl         peopl 1995
## will           will 1779
## support     support 1707
## get             get 1541
## right         right 1523
## plan           plan 1512
## privat       privat 1444
## fight         fight 1378
## want           want 1368
```

```
top_20%>%
  ggplot(aes(reorder(word,freq), freq)) + geom_bar(stat = "identity") + xlab("Term") + ylab("Frequency")
```




```
findAssocs(doc_matrix, terms = c('health','care','system'), corlimit = 0.25)
```

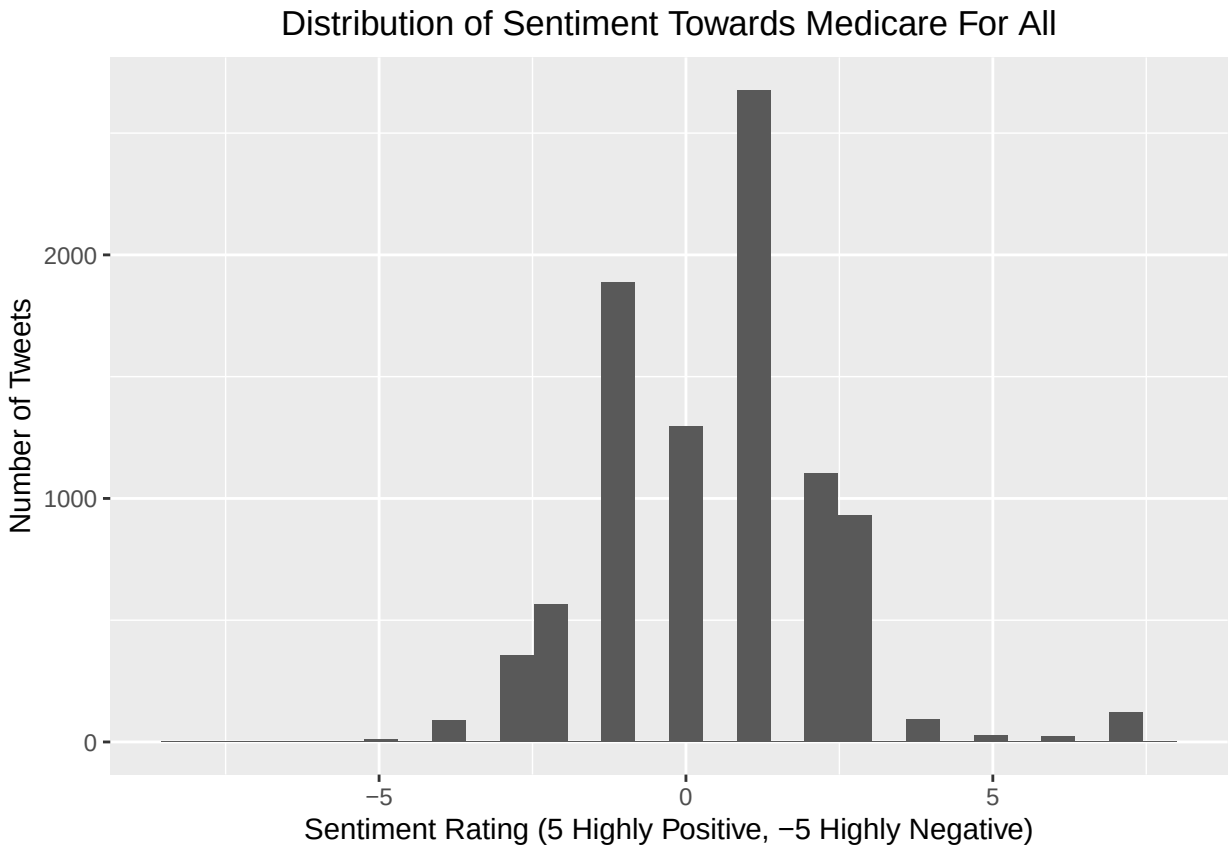
```
## $health
##            univers              privat                plan               insur
##               0.43                0.38                0.34                0.34
##              biden        trumpwarroom              scheme                path
##               0.34                0.33                0.33                0.32
##              ultim       governmentrun           socialist    factcheckdotorg
##               0.32                0.30                0.29                0.26
## httpstcomhntbiylxk          singlepay               fals
##               0.26                0.25                0.25
##
## $care
## univers
##    0.55
##
## $system
##     payer     singl singlepay    privat     insur
##      0.53      0.53      0.42      0.30      0.26
```

```
df_sentiment <- data_frame(text = character(), positive = double(), negative = double())
for(index in seq_along(corpus)){
  tweet <- corpus[[index]]$content
  tokens <- data_frame(text = tweet) %>% unnest_tokens(word,text)
  tokens%>%
     inner_join(get_sentiments("bing"))%>%
    count(sentiment)%>%
    spread(sentiment, n, fill = 0) -> rating
```

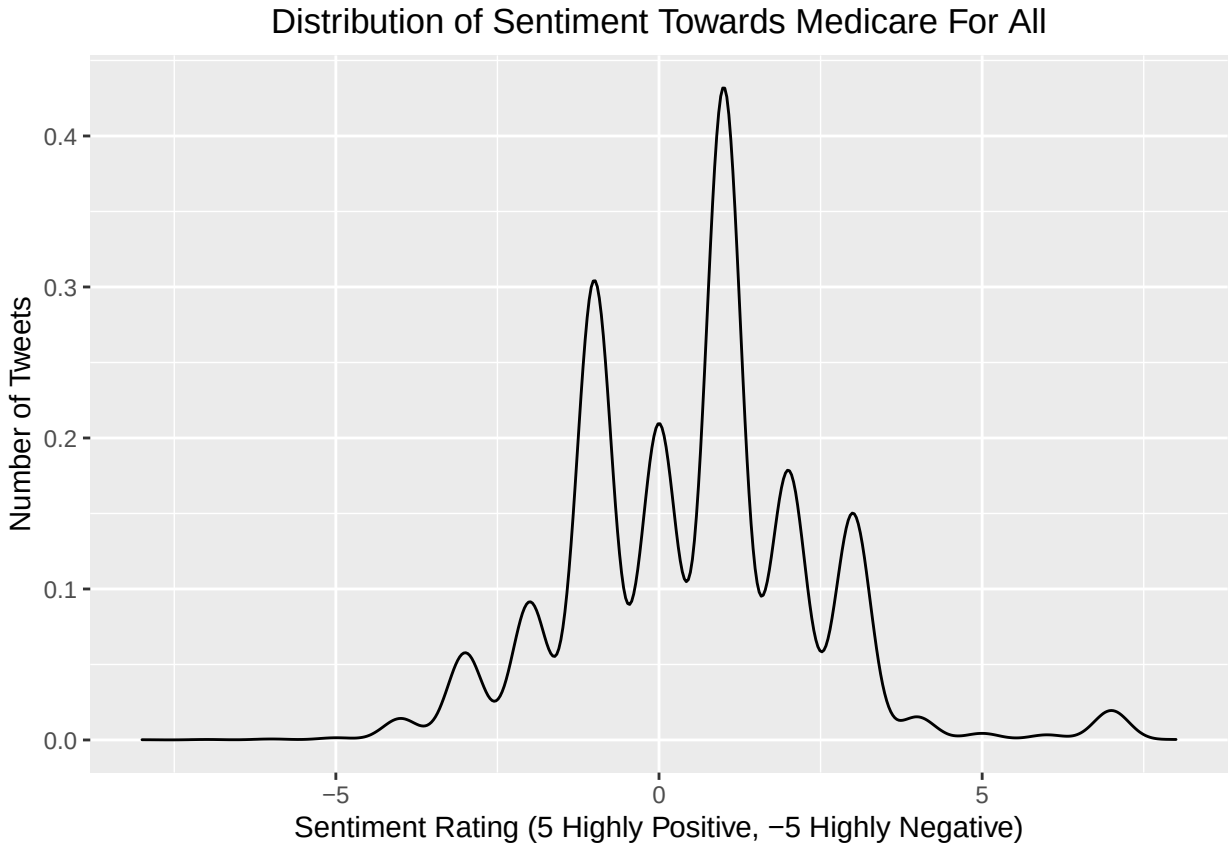
```
  if (ncol(rating) == 0){
    next()
  }
  if (ncol(rating) == 1){
    var_1 = names(rating[1])
    if(var_1 == 'positive'){
      rat = rating$positive
      df_sentiment%>%
        add_row(positive = rat, text = tweet) -> df_sentiment
    }
    if(var_1 == 'negative'){
      rat = rating$negative
      df_sentiment%>%
        add_row(negative = rat, text = tweet) -> df_sentiment
    }
  }

  if (ncol(rating) == 2) {
    df_sentiment%>%
      add_row(negative = rating$negative, positive = rating$positive, text = tweet) -> df_sentiment

  }


}
```

```
df_sentiment%>%
  replace_na(list(positive = 0, negative = 0, text = "Blank")) -> df_sentiment

df_sentiment%>%
  mutate(total = positive - negative) -> df_sentiment

write_excel_csv(df_sentiment, 'Data/sentiment_scores.xlsx')
```

```
df_sentiment%>%
  ggplot(aes(total)) + labs(title = "Distribution of Sentiment Towards Medicare For All", x = 'Sentimen

par(mfrow = c(2,2))
fig_1 + geom_histogram()
```




```
fig_1 + geom_density()
```


Distribution of Sentiment Towards Medicare For All
Number of Tweets
0.0
0.1
0.2
0.3
0.4
−5
0
5
Sentiment Rating (5 Highly Positive, −5 Highly Negative)