HW6-Spandan Maaheshwari

Spandan Maaheshwari

2022-12-02

Problem 1 ->

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2

## Warning: package 'ggplot2' was built under R version 4.2.2

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tidytext)
```

```
## Warning: package 'tidytext' was built under R version 4.2.2
```

```
library(tokenizers)
```

```
## Warning: package 'tokenizers' was built under R version 4.2.2
```

```
library(glmnet)
```

```
## Warning: package 'glmnet' was built under R version 4.2.2

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 4.2.2

##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
## Loaded glmnet 4.1-6
```

```
data_tweets = read_csv("/Users/SPANDAN/DS 5110/twitter/twitter/realDonaldTrump-20201106.csv")
```

```
## Rows: 55090 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (2): text, device
## dbl  (3): id, favorites, retweets
## lgl  (2): isRetweet, isDeleted
## dttm (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(data_tweets, 10)
```

```
## # A tibble: 10 x 8
##         id text       isRet~1 isDel~2 device favor~3 retwe~4 date
##      <dbl> <chr>      <lgl>   <lgl>   <chr>    <dbl>   <dbl> <dttm>
##  1 9.85e16 Republica~ FALSE   FALSE   Tweet~      49     255 2011-08-02 18:07:48
##  2 1.23e18 I was thr~ FALSE   FALSE   Twitt~   73748   17404 2020-03-03 01:34:50
##  3 1.22e18 RT @CBS_H~ TRUE    FALSE   Twitt~       0    7396 2020-01-17 03:22:47
##  4 1.30e18 The Unsol~ FALSE   FALSE   Twitt~   80527   23502 2020-09-12 20:10:58
##  5 1.22e18 RT @MZHem~ TRUE    FALSE   Twitt~       0    9081 2020-01-17 13:13:59
##  6 1.22e18 RT @White~ TRUE    FALSE   Twitt~       0   25048 2020-01-17 00:11:56
##  7 1.32e18 "I'm runn~ FALSE   FALSE   Twitt~  149007   34897 2020-10-12 22:22:39
##  8 1.22e18 Getting a~ FALSE   FALSE   Twitt~  285863   30209 2020-02-01 16:14:02
##  9 1.32e18 https://t~ FALSE   FALSE   Twitt~  130822   19127 2020-10-23 04:52:14
## 10 1.32e18 https://t~ FALSE   FALSE   Twitt~  153446   20275 2020-10-23 04:46:53
## # ... with abbreviated variable names 1: isRetweet, 2: isDeleted, 3: favorites,
## #   4: retweets
```

```
summary(data_tweets)
```

```
##        id                  text             isRetweet       isDeleted
##  Min.   :1.698e+09   Length:55090       Mode :logical   Mode :logical
##  1st Qu.:4.531e+17   Class :character   FALSE:45755     FALSE:54050
##  Median :7.217e+17   Mode  :character   TRUE :9335      TRUE :1040
##  Mean   :7.844e+17
##  3rd Qu.:1.180e+18
##  Max.   :1.325e+18
##     device            favorites          retweets
##  Length:55090       Min.   :      0   Min.   :     0
##  Class :character   1st Qu.:     11   1st Qu.:    54
##  Mode  :character   Median :    154   Median :  2897
##                     Mean   :  25573   Mean   :  7917
##                     3rd Qu.:  40914   3rd Qu.: 12312
##                     Max.   :1869706   Max.   :408866
##       date
##  Min.   :2009-05-04 18:54:25.00
##  1st Qu.:2014-04-07 11:09:43.25
##  Median :2016-04-17 14:07:55.00
##  Mean   :2016-10-06 18:03:51.64
##  3rd Qu.:2019-10-05 03:20:56.00
##  Max.   :2020-11-06 17:38:17.00
```

Removing re-tweets

```
data_tweets$id <- format(data_tweets$id, scientific=F)

tidy_data <- data_tweets %>%
filter(isRetweet == FALSE)
```

Removing tweets without spaces

```
tidy_data <- tidy_data[-which(is.na(str_locate(tidy_data$text, " "))),]
```

Removing &amp, URLs, twitter user names and special characters

```
tidy_data$text <- gsub("(f|ht)(tp)(s?)(://)(.*)[.|/](.*)", " ", tidy_data$text)
tidy_data$text <- gsub("@\\w+", "", tidy_data$text)
tidy_data$text <- gsub("&amp", "", tidy_data$text)
tidy_data$text <- tolower(tidy_data$text)

tidy_data <- tidy_data %>%
rename(year = date)
tidy_data$year <- str_sub(tidy_data$year, 1, 4)
```

Removing variations on Donald Trump's name

```
a <- 'donald*'
donald <- str_subset(tidy_data$text,a)

b <- 'trump*'
trump <- str_subset(tidy_data$text,b)

tidy_data$text <- gsub("donald","dt", tidy_data$text)
tidy_data$text <- gsub("realdonaldtrump", "dt", tidy_data$text)
tidy_data$text <- gsub("trump","dt", tidy_data$text)
tidy_data$text <- gsub("donaldtrump","dt", tidy_data$text)
tidy_data$text <- gsub("realdonal","dt", tidy_data$text)
tidy_data$text <- gsub("donal","dt", tidy_data$text)
tidy_data$text <- gsub("donaldTrump","dt", tidy_data$text)
tidy_data$text <- gsub("DonaldTrump","dt", tidy_data$text)
tidy_data$text <- gsub("dt", "", tidy_data$text)
tidy_data <- tidy_data[!(tidy_data$text == ""), ]
```

Removing stop words

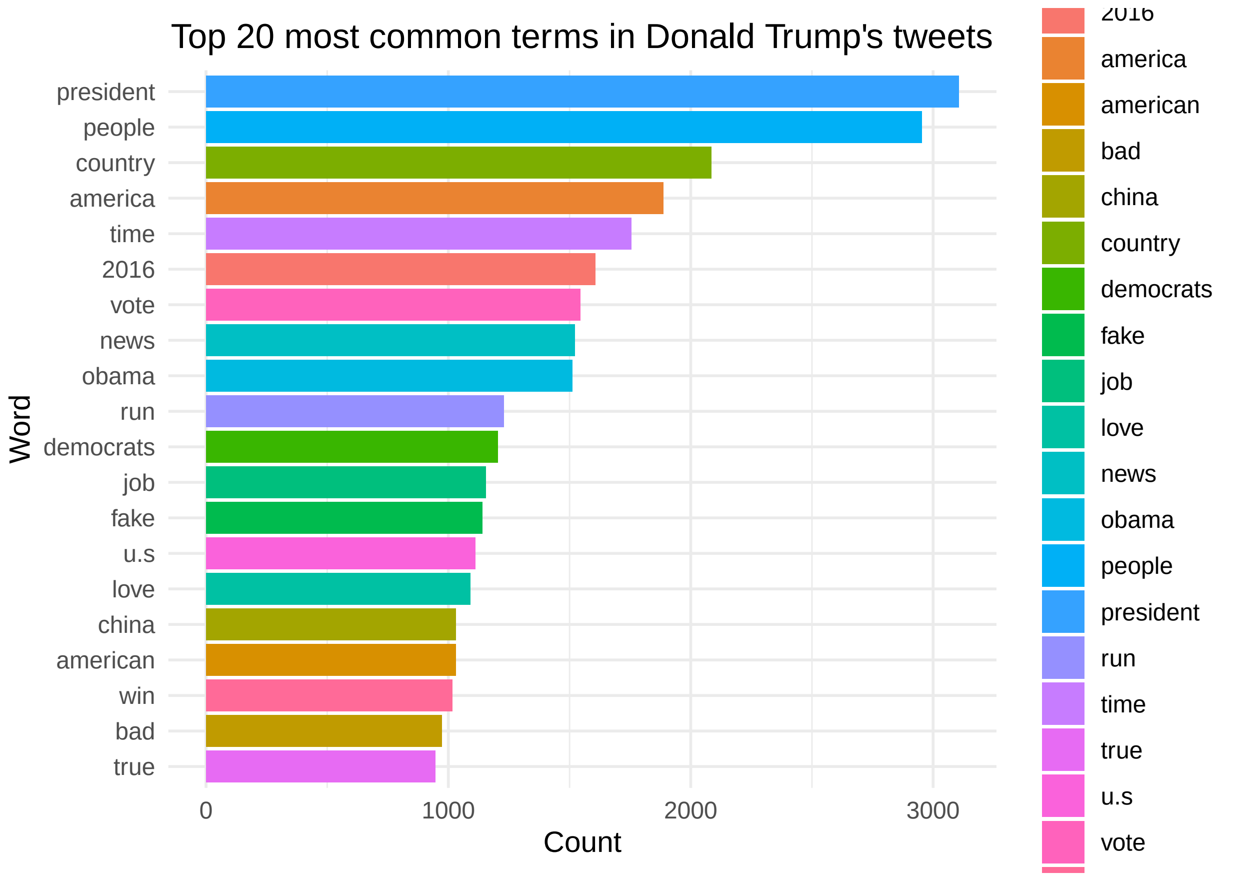
```
tidy_data <- unnest_tokens(tidy_data, output = "word", input = text)
tidy_data <- anti_join(tidy_data, stop_words, by="word")

head(tidy_data, 10)
```

```
## # A tibble: 10 x 8
##    id                    isRetweet isDeleted device  favor~1 retwe~2 year  word
##    <chr>                 <lgl>     <lgl>     <chr>     <dbl>   <dbl> <chr> <chr>
##  1 "  98454970654916608" FALSE     FALSE     TweetD~      49     255 2011  repu~
##  2 "  98454970654916608" FALSE     FALSE     TweetD~      49     255 2011  demo~
##  3 "  98454970654916608" FALSE     FALSE     TweetD~      49     255 2011  crea~
##  4 "  98454970654916608" FALSE     FALSE     TweetD~      49     255 2011  econ~
##  5 "1234653427789070336" FALSE     FALSE     Twitte~   73748   17404 2020  thri~
##  6 "1234653427789070336" FALSE     FALSE     Twitte~   73748   17404 2020  city
##  7 "1234653427789070336" FALSE     FALSE     Twitte~   73748   17404 2020  char~
##  8 "1234653427789070336" FALSE     FALSE     Twitte~   73748   17404 2020  north
##  9 "1234653427789070336" FALSE     FALSE     Twitte~   73748   17404 2020  caro~
## 10 "1234653427789070336" FALSE     FALSE     Twitte~   73748   17404 2020  thou~
## # ... with abbreviated variable names 1: favorites, 2: retweets
```

Top 20 most common terms in Donald Trump's tweets:

```
tidy_data %>%
count(word, sort=TRUE) %>%
top_n(20) %>%
ggplot(aes(x=reorder(word, n), y=n, fill = word)) +
  geom_col() + coord_flip() + labs(x="Word", y="Count",
    title="Top 20 most common terms in Donald Trump's tweets") +
       theme_minimal() + theme(plot.title = element_text(hjust = 0.5))
```

```
## Selecting by n
```



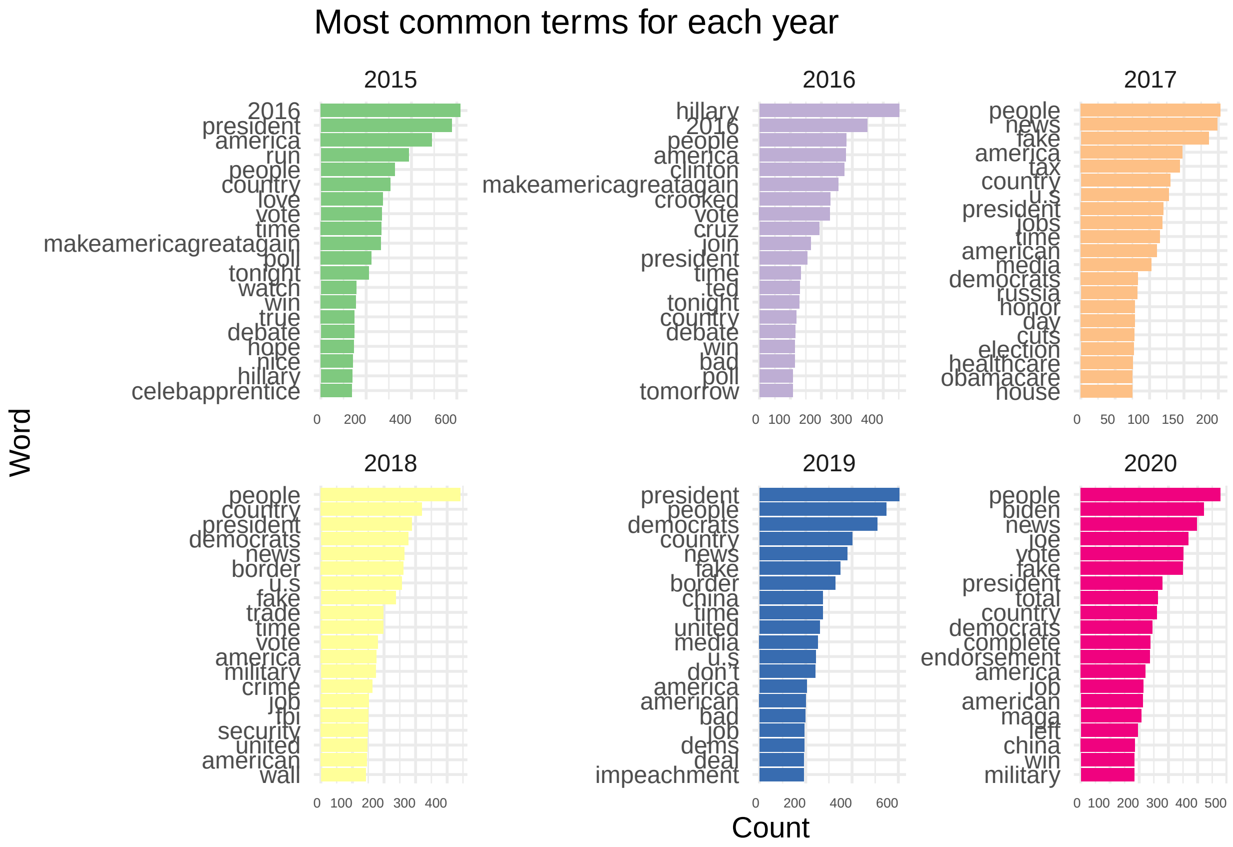
Explanation for Problem 1:

The most common term used in Donald Trump's tweets is president

Problem 2 ->

```
tidy_data <- tidy_data %>%
        filter(year >= 2015)

tidy_data %>%
    group_by(year) %>%
    count(word, year, sort=TRUE) %>%
    top_n(20) %>%
    ggplot(aes(x=reorder_within(word, n, year), y=n, fill=year)) +
    geom_col(show.legend=FALSE) + facet_wrap(~year, scales="free") +
    coord_flip() + labs(x="Word", y="Count",
    title="Most common terms for each year", fill="Year") +
    scale_fill_brewer(palette="Accent") + scale_x_reordered() +
    theme_minimal() +
    theme(axis.text.x = element_text(angle=0,hjust=1,vjust=0.5,size=5))
```

```
## Selecting by n
```



Explanation for Problem 2 ->

In the bulk of popular words during the past few years, “People” has been referenced. We discover that “2016” was the most frequently used word in 2015 due to the presidential elections in that year.

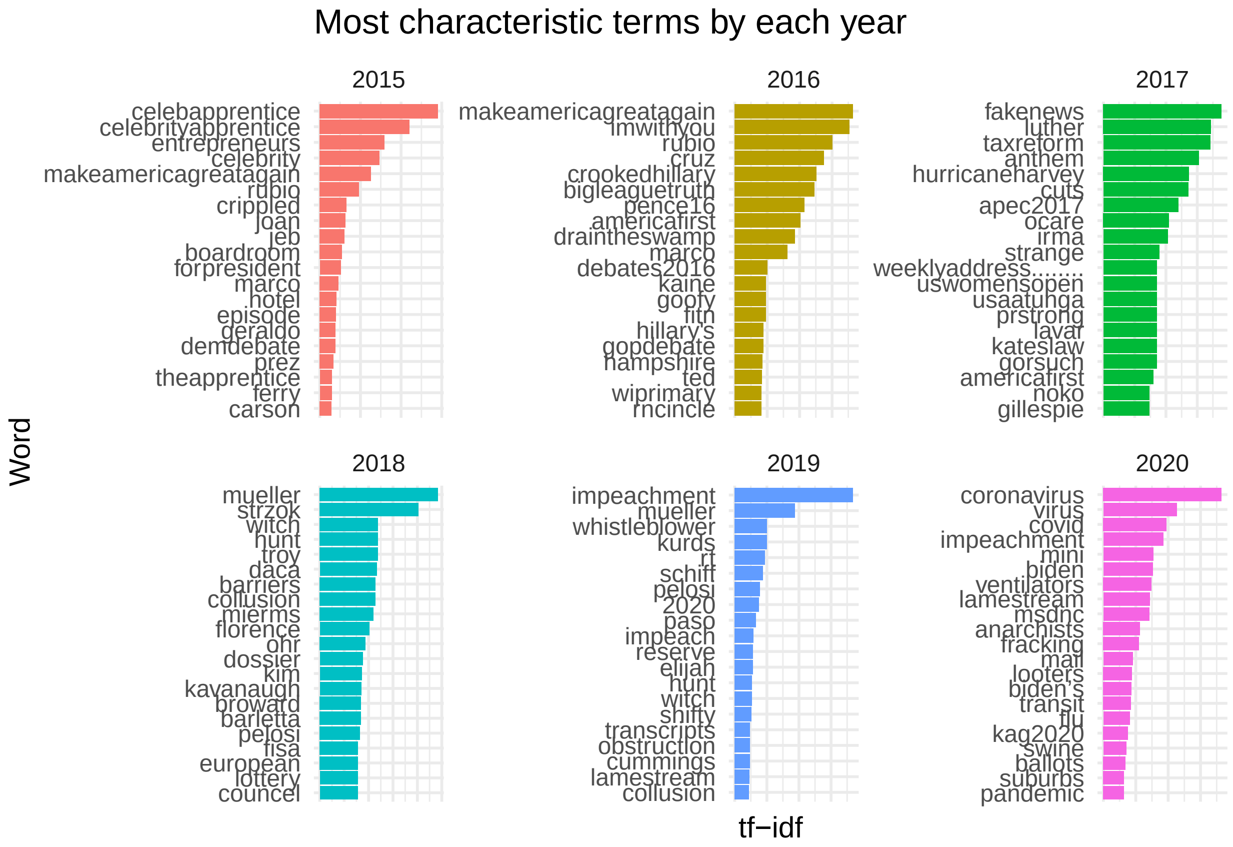
The word “hillary” has increased in usage during Hillary Clinton’s presidential campaign in 2016. Similar to today, Joe Biden is the second most often used word in 2020.

Year 2019 has seen a lot of use of the word “President” due to the fact that 2020 is the year of the elections.

It makes sense that “People” was the most frequently used term in the other two years because those were the years between the elections and during Trump’s presidency.

Problem 3 ->

```
trump_tf_idf <- tidy_data %>%
count(year, word, sort=TRUE) %>%
bind_tf_idf(term=word, document=year, n=n)
trump_tf_idf %>%
  group_by(year) %>%
  top_n(20, wt=tf_idf) %>%
  ggplot(aes(x=reorder_within(word, tf_idf, year),
  y=tf_idf, fill=factor(year))) +
  geom_col(position="dodge", show.legend=FALSE) +
  coord_flip() + facet_wrap(~year, scales="free") +
  labs(x="Word", y="tf-idf", title="Most characteristic terms by each year", fill="Year") +
  scale_x_reordered() +  scale_y_continuous(labels=NULL) + theme_minimal() +
  theme(axis.text.x = element_text(angle=0,hjust=1,vjust=0.5,size=5))
```



Explanation for Problem 3:

Because of the 2016 presidential election, we can see that the phrases “celebapprentice” and “making America Great Again” were used in 2015 and 2016, respectively. The word “fakenews” was used the most in 2017.

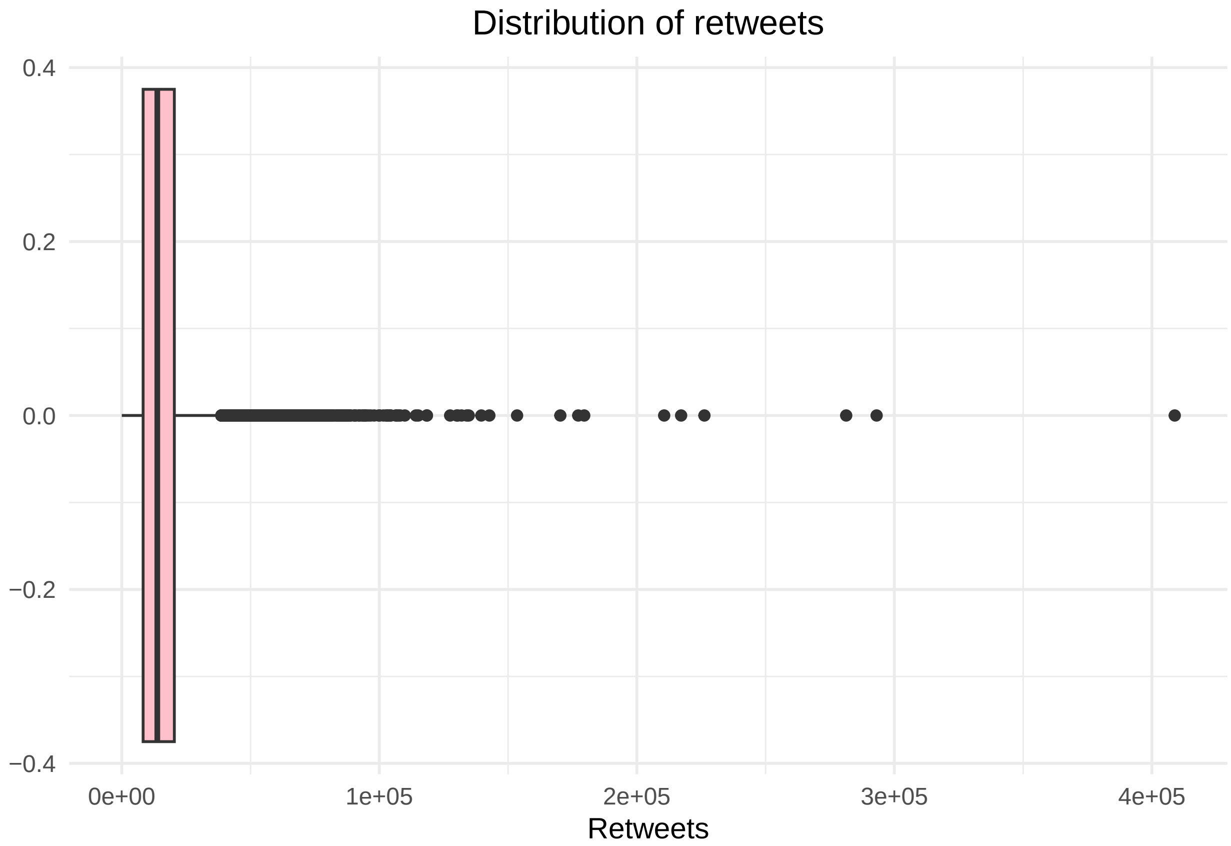
Robert Muller’s Russian investigation started in 2018, hence we consider Muller to be the first word of the year. Mueller is the second word of 2019, as the study was finished in December 2018.

Trump was the subject of an impeachment probe in 2019. Impeachment is the first word we come across in the year 2019.

In the year 2020, the COVID-19 Coronavirus is frequently mentioned. We hear several different terms related to the epidemic and health in the year 2020.

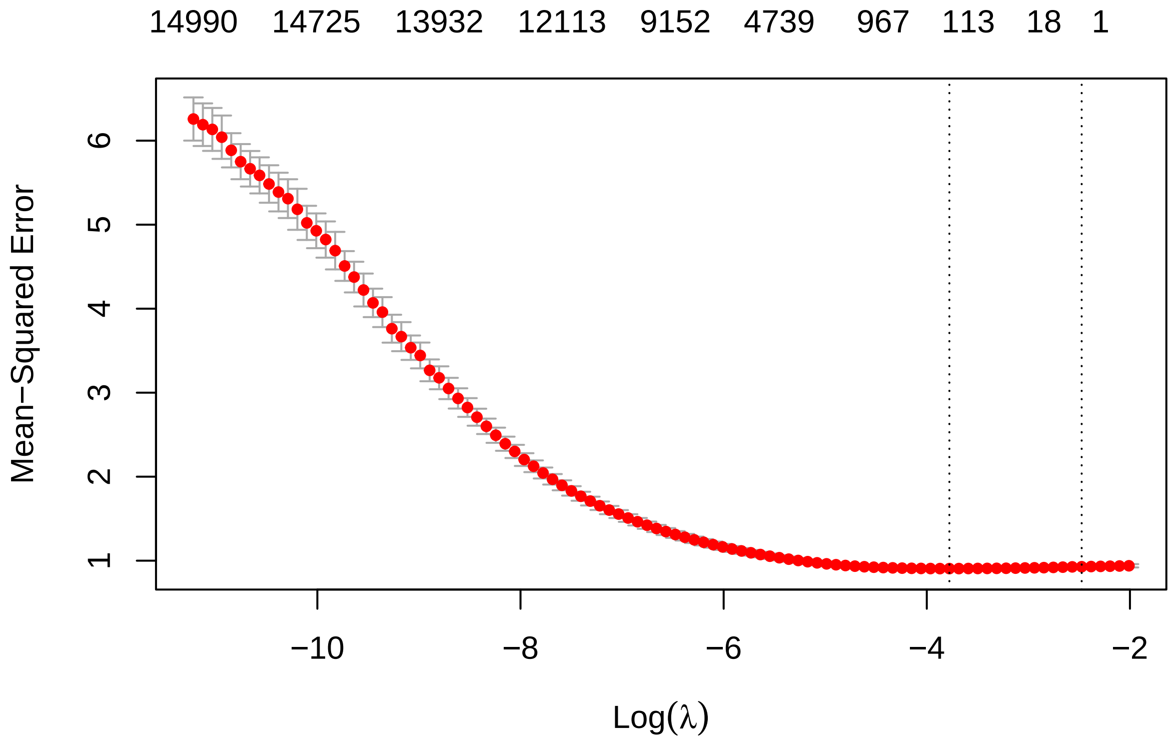
Problem 4 ->

```
tidy_data <- tidy_data %>%
filter(year >= 2016)
df_data <- left_join(tidy_data, trump_tf_idf, by=c("year","word")) %>%
select(c("id","retweets","word","n"))
df<- df_data %>%
group_by(id) %>%
summarise(retweets = mean(retweets))

df %>%
   ggplot() +
   geom_boxplot(aes(x = (retweets)), fill = "pink")+
   labs(x = "Retweets", title="Distribution of retweets")+
   theme_minimal()+
  theme(plot.title = element_text(hjust = 0.5))
```



The above graph is right skewed and not normal. Hence for using glmnet we transform the data by using log1p

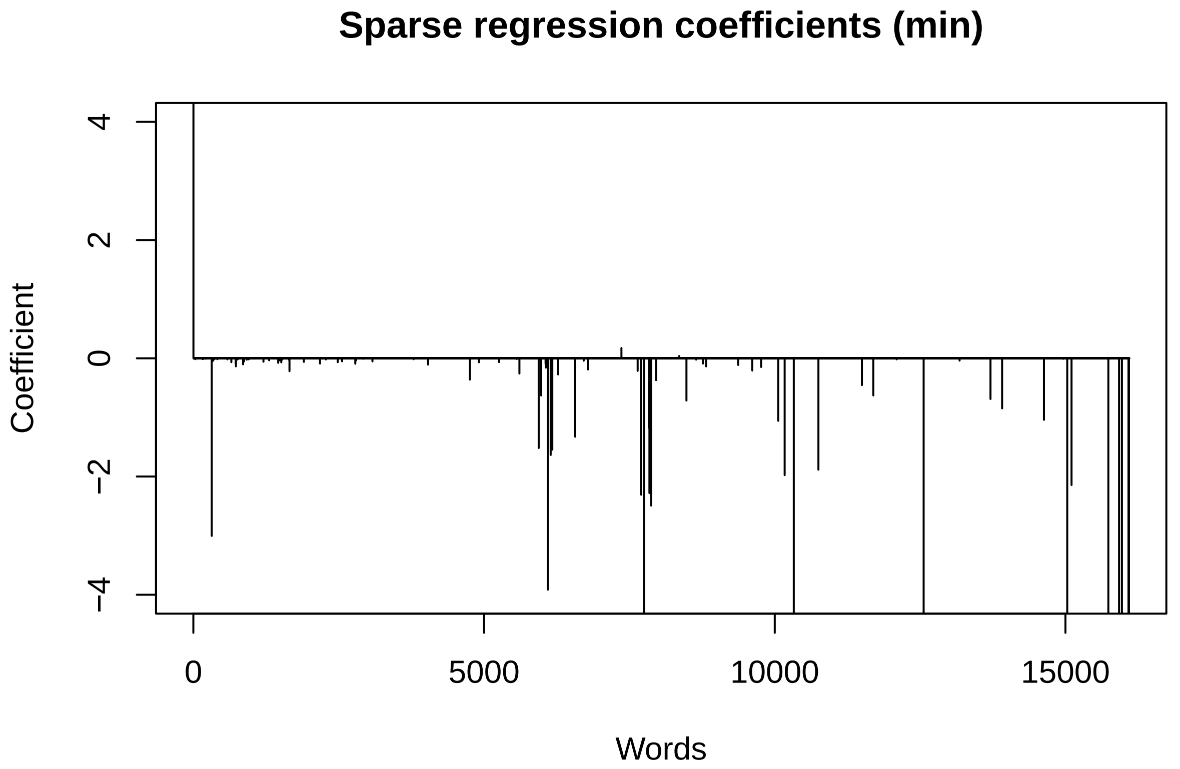
```
X <- cast_sparse(data = df_data, row = id, column = word, value = n)
Y <- as.matrix(log1p(df$retweets))
set.seed(1234)
cvfit <- cv.glmnet(X,Y, family = "gaussian")
plot(cvfit)
```



```
c1 <- coef(cvfit, s="lambda.min")
sum(c1 != 0)
```

```
## [1] 190
```

```
plot(c1, type= 'h', ylim=c(-4, 4),
xlab="Words", ylab="Coefficient",
main="Sparse regression coefficients (min)")
```



```
cvfit
```

```
##
## Call:  cv.glmnet(x = X, y = Y, family = "gaussian")
##
## Measure: Mean-Squared Error
##
##      Lambda Index Measure      SE Nonzero
## min 0.02288    20  0.9057 0.02373     189
## 1se 0.08416     6  0.9282 0.02000       2
```

Explanation for Problem 4:

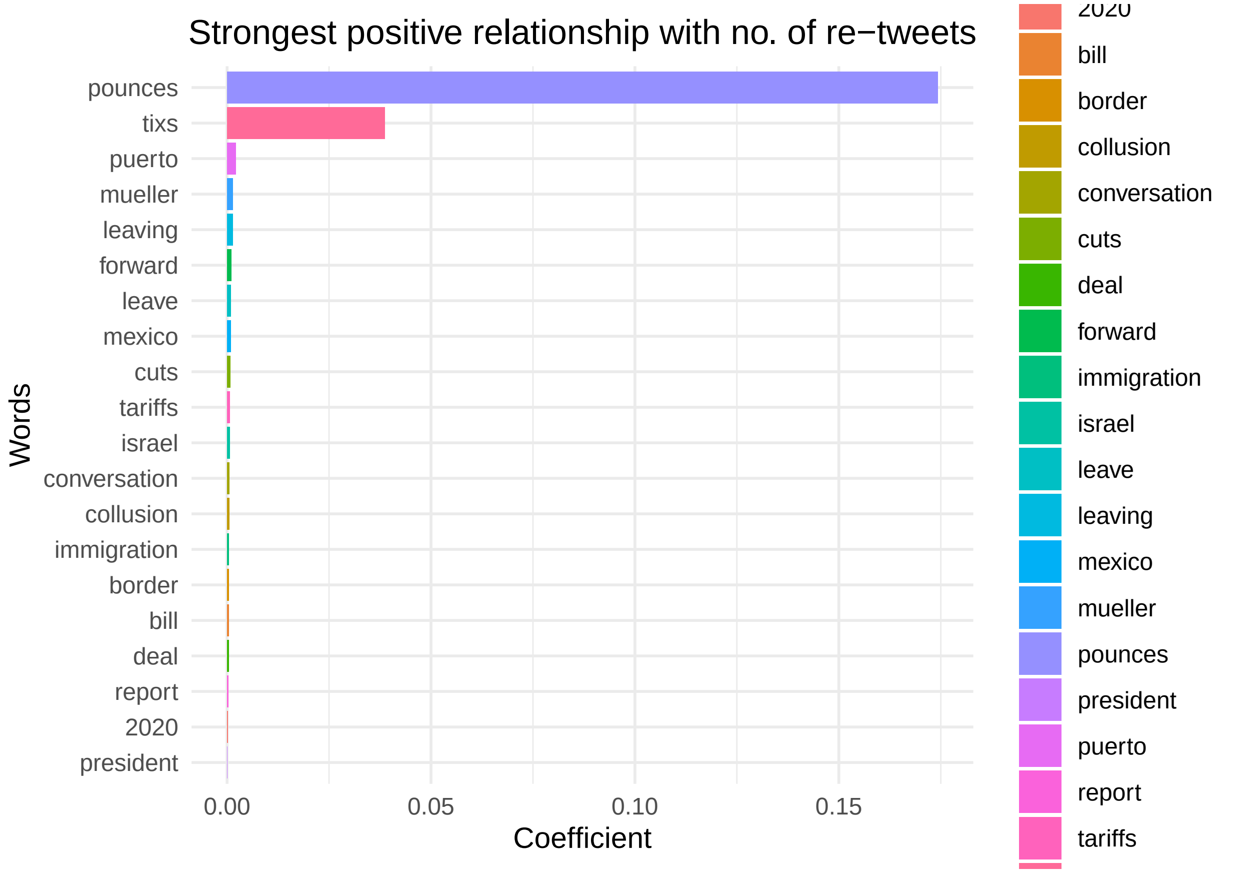
The minimum value of lambda is 0.02288 and the number of non zero coefficients are 189

Problem 5 ->

```
coeff_vars <-rownames(c1)[which(c1 >0)]
coeff <- c1[which(c1>0)]
model <- as.data.frame(coeff_vars)
model$coeff <- coeff
model <- model %>%
filter(coeff_vars != "(Intercept)")
model <- model[order(model$coeff, decreasing = TRUE),]
model %>%
top_n(20) %>%
```

```r
ggplot(aes(y = reorder(coeff_vars, coeff), x = coeff, fill=coeff_vars)) +
geom_col() +
scale_x_continuous(labels = scales::label_comma()) +
labs(x = "Coefficient", y = "Words",
title = "Strongest positive relationship with no. of re-tweets")+
theme_minimal()+
theme(plot.title = element_text(hjust = 0.5))
```

```
## Selecting by coeff
```



Explanation for Problem 5:

Looking at the top 20 strongest positive words with number of tweets, we see that pounces is the strongest positive word with the highest number of retweets.

Out of these 20, five words including pounces, tixs, puerto, mueller, and leaving—have higher coefficients.

All of the other words have low coefficient values and a less favourable correlation with the amount of retweets.