Course Project - Logistic regression

Jaeyoung Lee

11/29/2020

```
## [1] "Summary of data"
```

```
##  retweet_count       party               score
##  Min.   :     1   Length:12515164    Min.   :-28.30769
##  1st Qu.:    85   Class :character   1st Qu.: -0.30769
##  Median :   906   Mode  :character   Median :  0.00000
##  Mean   :  6840                      Mean   :  0.03608
##  3rd Qu.:  5431                      3rd Qu.:  0.41026
##  Max.   :625495                      Max.   : 33.38462
```

```
## [1] "Summary of logistic regression coefficients"
```

```
##   (Intercept)     retweet_count           score
##  Min.   :0.8465   Min.   :-1.634e-05   Min.   :-0.68466
##  1st Qu.:1.0491   1st Qu.:-6.950e-06   1st Qu.:-0.34276
##  Median :1.0990   Median :-4.887e-06   Median :-0.27249
##  Mean   :1.0995   Mean   :-4.684e-06   Mean   :-0.27241
##  3rd Qu.:1.1532   3rd Qu.:-3.029e-06   3rd Qu.:-0.20221
##  Max.   :1.3437   Max.   : 1.615e-05   Max.   : 0.03723
```

```
## [1] "Summary of exponenetiated coefficients"
```

```
##   (Intercept)    retweet_count     score
##  Min.   :2.331   Min.   :1     Min.   :0.5043
##  1st Qu.:2.855   1st Qu.:1     1st Qu.:0.7098
##  Median :3.001   Median :1     Median :0.7615
##  Mean   :3.012   Mean   :1     Mean   :0.7655
##  3rd Qu.:3.168   3rd Qu.:1     3rd Qu.:0.8169
##  Max.   :3.833   Max.   :1     Max.   :1.0379
```

The raw data has 24201654 observations. To reduce the size of data and to fix the right tail distribution of *Re-tweet counts*, the observations which have zero counts are excluded. Also, to control the multicollinearity, redundant variables are not selected for constructing a model.

From the raw data, 12515164 observations are used which *Re-tweet counts* are non-zero. Since the observations are still too large, it is necessary to use some methods handling the large data. To speed up computation and deal with large data, a re-sampling method is used which is similar to Bootstrap. Unlike Bootstrap, we sample without replacement and the sample size of each sample is smaller than the original.

The algorithm for the re-sampling method is as follows:

1. For j = 1, . . . , $M = 1000$,
   (a) Obtain sample of size $n = 1000$.
   (b) Fit logistic regression from the jth sample.
   (c) Store the coefficients of the model.

From the $M = 1000$ samples, we can obtain a matrix of logistic regression coefficients.

From the result, we can build a logistic regression model as follows:

$$\log \frac{P(Y = Republicans)}{P(Y = Democrats)} = 1.0995 - 4 \times 10^{-6} Retweet - 0.27241 Score \tag{1}$$

where the baseline is *Democrats*, and the odds is defined as the proportion of Republicans to Democrats.

From the model, we can notice that the input *Re-tweet counts* is not significant. In other words, there is quite little effect of *Re-tweet counts* to the model. In addition, under fixed *Re-tweet counts*, the odds of Republicans is $exp(1.0995 - 0.27241 \times Score) = 3.012 \times exp(-0.27241 \times Score)$. Therefore, as $Score$ increases as one, the odds of Republicans decreases 23.84%. In other words, we can say that increase of the sentiment score of the tweets leads to the decrease of the odds of Republicans, and reversely, the increase of the odds of Democrats.

```r
knitr::opts_chunk$set(echo = FALSE)
library(data.table)
library(tidyverse)
library(bit64)
library(lubridate)
# Load data
tweetdata <- fread('uselection_tweets_1jul_11nov.csv')

# Select variables of interest
# Remove redundant varaiables (Negativity, Positivity) b/c there is score
tweetdata_select <- tweetdata %>% select(`Retweet-Count`, PartyName, Score)
names(tweetdata_select) <- c('retweet_count', 'party', 'score')

saveRDS(tweetdata_select, 'tweetdata_select.RDS')

# Load RDS data
twitter_data <- readRDS('tweetdata_select.RDS')

# Remove observations with zero retweet_count
# Left observations mentioned one of two parties only: Republicans or Democrats
twitter <- twitter_data %>% filter(retweet_count != 0) %>%
  filter(party == 'Republicans' | party == 'Democrats')

##### Multinomial Logistic regression with bootstrap-like method #####
# Sample from the data
set.seed(20202021)
n <- 1000 # Sample size
M <- 1000   # Number of iteration

logit_coefficients <- NULL # Coefficients from M logit models
# Generate M coefficients
for(i in 1:M){
  # Sample from data with size n
  index <- sample(1:nrow(twitter), size = n)
  twitter_sample <- twitter[index,]

  # Response variable : Democrats, Republicans
  logit_fit <- nnet::multinom(party~retweet_count + score,
                              data = twitter_sample, trace = FALSE)
  logit_coefficients <- rbind(logit_coefficients, summary(logit_fit)$coefficients)

}

# Save RData
save.image('logit_model.RData')


# Load RData
load('logit_model.RData')

# Summary of data used
print('Summary of data')
summary(twitter)
```

```r
# Summary of logistic regression coefficients
print('Summary of logistic regression coefficients')
summary(logit_coefficients)

# Exponentiate regression coefficients
print('Summary of exponenetiated coefficients')
summary(exp(logit_coefficients))
```