Week 1 Assignment Solution

Alicia R. Chen, Yining Sun

9/1/2021

```
# load packages
packages <- c("tidyverse","data.table", "lubridate", 'ggplot2')
lapply(packages, library, character.only = TRUE)
[[1]]
 [1] "forcats"   "stringr"   "dplyr"     "purrr"     "readr"     "tidyr"
 [7] "tibble"    "ggplot2"   "tidyverse" "stats"     "graphics"  "grDevices"
[13] "utils"     "datasets"  "methods"   "base"

[[2]]
 [1] "data.table" "forcats"    "stringr"    "dplyr"      "purrr"
 [6] "readr"      "tidyr"      "tibble"     "ggplot2"    "tidyverse"
[11] "stats"      "graphics"   "grDevices"  "utils"      "datasets"
[16] "methods"    "base"

[[3]]
 [1] "lubridate"  "data.table" "forcats"    "stringr"    "dplyr"
 [6] "purrr"      "readr"      "tidyr"      "tibble"     "ggplot2"
[11] "tidyverse"  "stats"      "graphics"   "grDevices"  "utils"
[16] "datasets"   "methods"    "base"

[[4]]
 [1] "lubridate"  "data.table" "forcats"    "stringr"    "dplyr"
 [6] "purrr"      "readr"      "tidyr"      "tibble"     "ggplot2"
[11] "tidyverse"  "stats"      "graphics"   "grDevices"  "utils"
[16] "datasets"   "methods"    "base"
```

Data Prep

Note that tweets data has many duplicates rows.

```
### Tweets data
# Load tweets data
tweets <- fread("IRA_tweets.csv")
tweets$Date <- as.Date(tweets$Date)
tweets <- unique(tweets)
```

You need to have filtered GTD data by the relevant years and country. Also need to use the islamist_groups dataset to add an indicator for islamist. Lastly, GTD data is not balanced so need to fill in 0's across the days where no event occured.

```
### GTD data
# Load and filter by Russia and 2015-2018
gtd <- fread("GTD.csv")
gtd <- filter(gtd, country_txt == "Russia")
gtd <- filter(gtd, iyear >= 2014)
gtd$Date <- as.Date(with(gtd, paste(iyear, imonth, iday,sep="-")), "%Y-%m-%d")

# Add indicator that a terrorist or islamist attack occured on these dates
gtd$terrorist <- 1
islamist_groups <- read_csv("islamist_groups.csv")
gtd$islamist <- ifelse(gtd$gname %in% c(islamist_groups$islamist_groups), 1,0)

# Balance GTD data
full_gtd <- gtd %>%
  select(Date, terrorist, islamist) %>%
  right_join(., data.frame(Date = unique(tweets$Date)))

full_gtd[is.na(full_gtd)] <- 0

# alternative code for this
# gtd %>%
#   select(Date, terrorist, islamist) %>%
#   merge(., tweets, all.y=T)
```

Some missing values in holidays data. After investigating, they are duplicates of other rows so I chose to remove those (there are other ways to address this).

```
### Russian holidays data
holidays <- fread("Russian_Holidays.csv")
holidays <- holidays[complete.cases(holidays), ]
holidays$holiday <- 1
holidays$MonthDay <- with(holidays, paste(Month, Day, sep="-"))
```

Constructing Panel

Note that holidays data also is unbalanced, so fill in 0's for non-holiday days once I construct the panel.

```
# Merge based on date
final_panel <- tweets %>%
  plyr::join(.,full_gtd) %>%
  mutate(MonthDay = paste(month(Date,label=TRUE), format(Date, "%d"), sep="-")) %>%
  plyr::join(., select(holidays, MonthDay, Religious:holiday)) %>%
  filter(Date <= as.Date("2018-06-30") & Date >= as.Date("2015-01-01"))
final_panel[is.na(final_panel)] <- 0
```

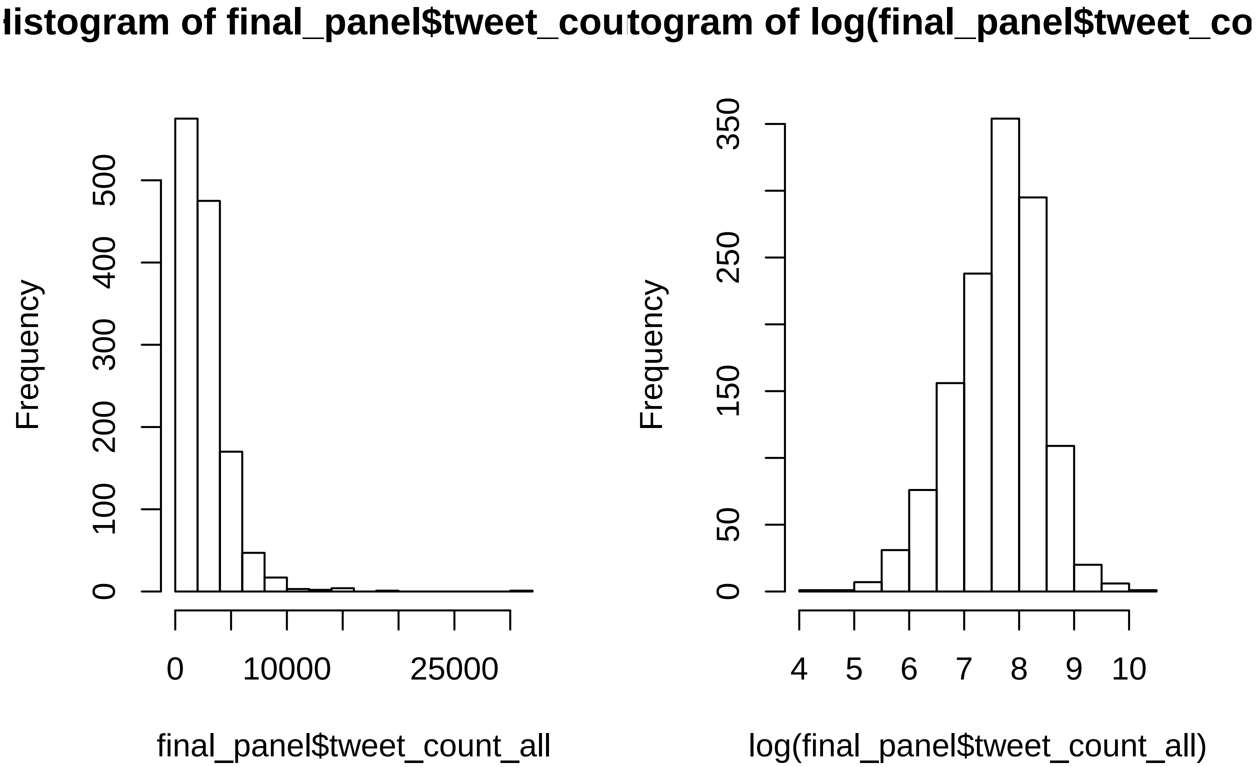
```
cat("Total observations:", nrow(final_panel))
Total observations: 1295
colSums(final_panel[,c('terrorist', 'islamist', 'Religious', 'Public', 'Political', 'holiday')])
terrorist  islamist Religious    Public Political   holiday
      121        36        12        20         0        43
```

Descriptive Statistics

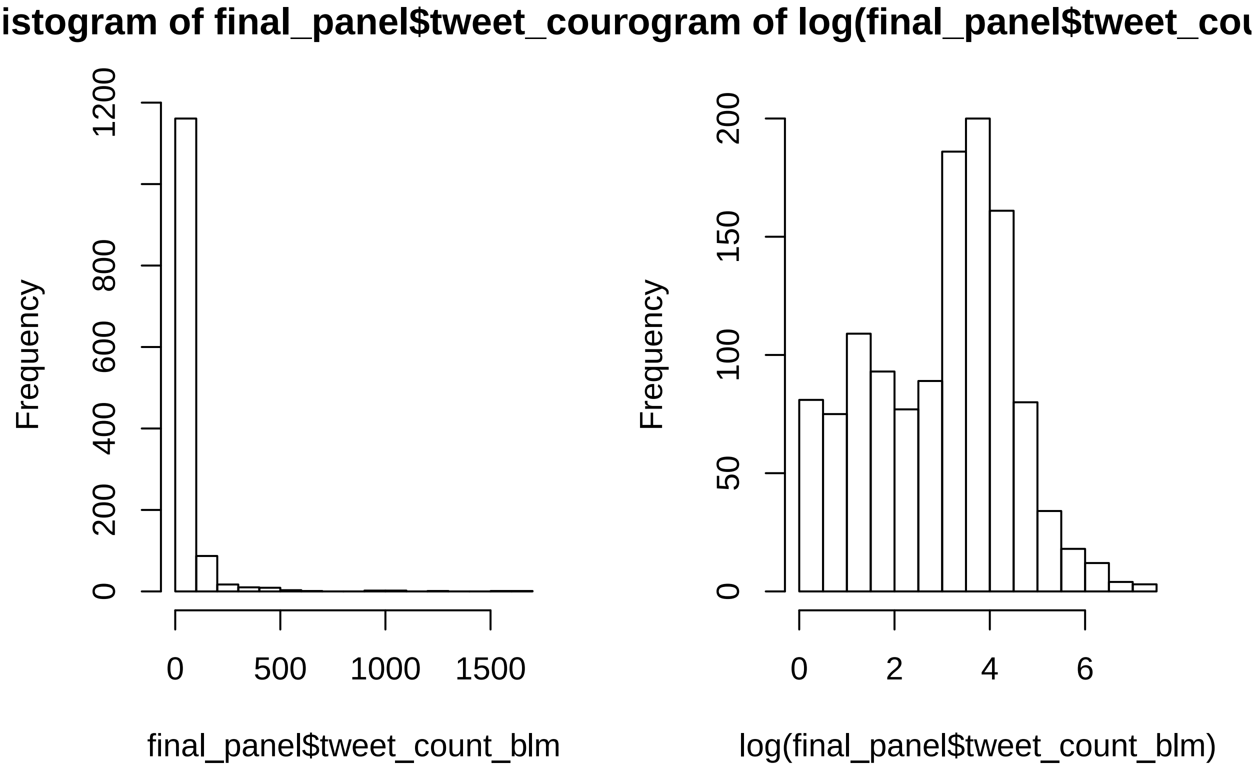
```
describe <- function(data, column){
  variable <- data[,get(column)]
  stats <- data.frame(variable = column,
                      length = length(na.omit(variable)),
                      mean = mean(na.omit(variable)),
                      median = median(na.omit(variable)),
                      min = min(na.omit(variable)),
                      max = max(na.omit(variable)),
                      sd = sd(na.omit(variable)))
  return(stats)
}

descriptives <- list()
# to get the three tweet count variables
for(i in 3:5){
  col <- colnames(final_panel)[i]
  stats <- describe(final_panel, col)
  descriptives[[i-2]] <- stats
}

df <- do.call("rbind",descriptives)
df
           variable length       mean median min   max        sd
1 tweet_count_islam   1295   98.16139     61   0  3343  143.4002
2   tweet_count_blm   1295   49.00386     23   0  1604  110.7262
3   tweet_count_all   1295 2659.38919   2206  82 30298 2133.1289
```

There are lots of different ways to do (b) through (d). This is just one example:

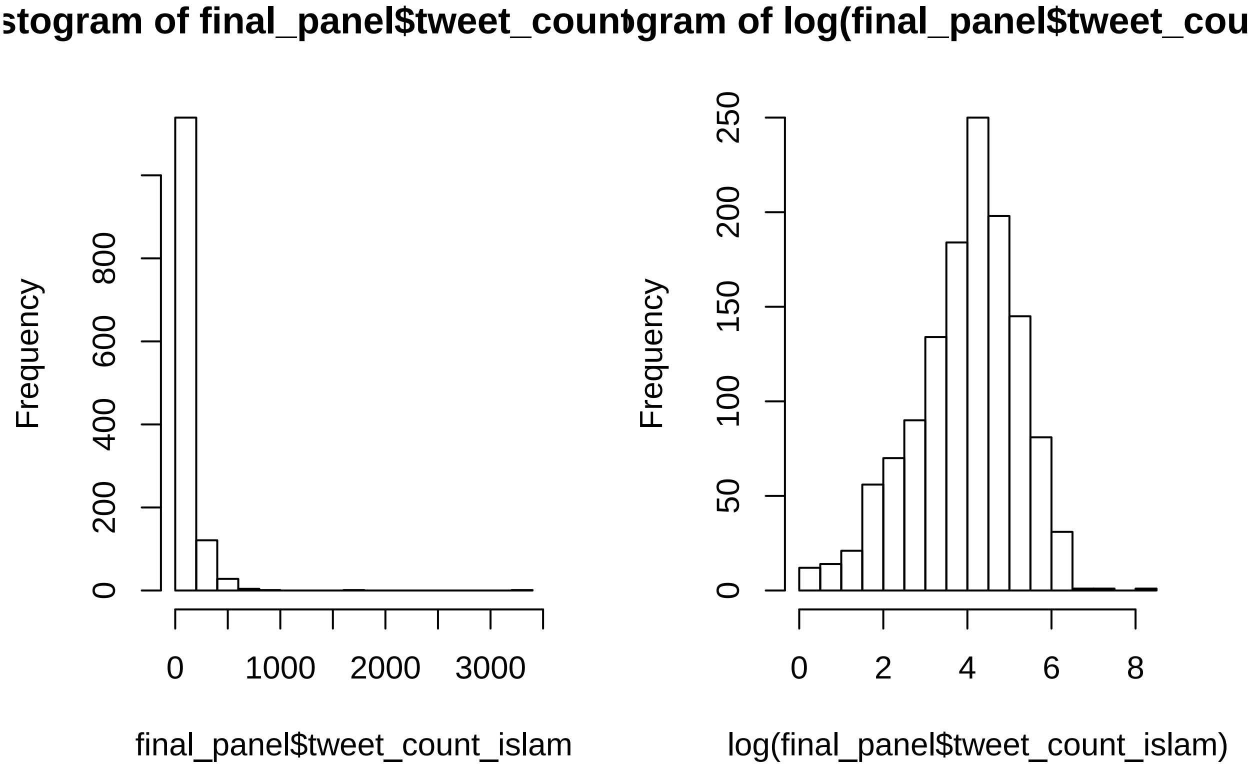
```
par(mfrow=c(1,2))
hist(final_panel$tweet_count_all)
hist(log(final_panel$tweet_count_all))
```



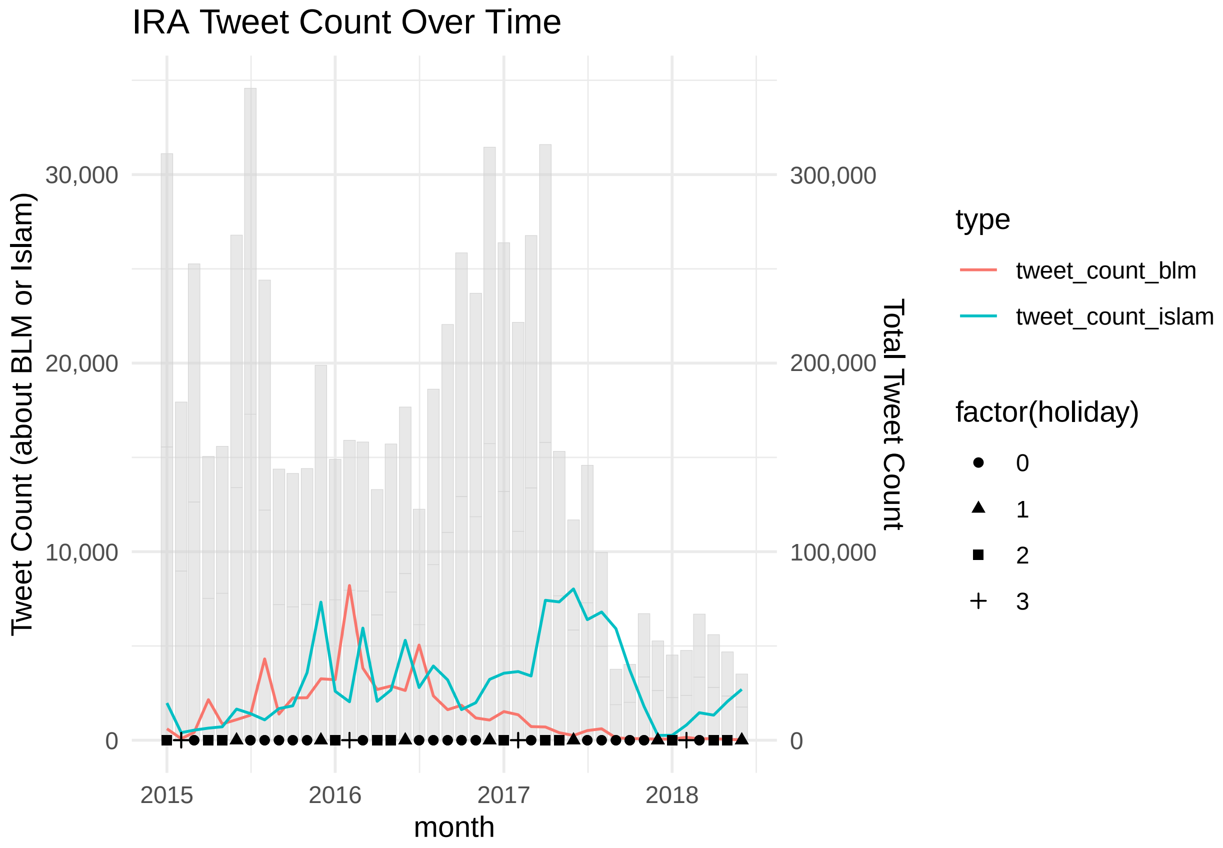
```
par(mfrow=c(1,2))
hist(final_panel$tweet_count_blm)
hist(log(final_panel$tweet_count_blm))
```



```
par(mfrow=c(1,2))
hist(final_panel$tweet_count_islam)
hist(log(final_panel$tweet_count_islam))
```



```r
byMonth <- final_panel %>%
  group_by(month = cut(Date, "month")) %>%
    summarise(tweet_count_all = sum(tweet_count_all),
              tweet_count_islam = sum(tweet_count_islam),
              tweet_count_blm = sum(tweet_count_blm),
              holiday = sum(holiday))
long <- gather(byMonth, type, tweet_count, tweet_count_islam:tweet_count_blm)
long$month <- as.Date(long$month)

coeff = 10
ggplot(long, aes(x=month)) +
  geom_bar(aes(y=tweet_count_all/coeff), stat = "identity",alpha=0.5,fill='lightgrey',color='lightgrey'
  geom_line(aes(y=tweet_count, group=type, color=type)) +
  theme_minimal() +
  scale_y_continuous(
    # Features of the first axis
    name = "Tweet Count (about BLM or Islam)",
    labels = scales::comma,
    # Add a second axis and specify its features
    sec.axis = sec_axis(~.*coeff, name="Total Tweet Count", labels = scales::comma)) +
  ggtitle("IRA Tweet Count Over Time") +
  geom_point(aes(x = month, y = holiday, shape = factor(holiday)))
```


IRA Tweet Count Over Time
Tweet Count (about BLM or Islam)
Total Tweet Count
month
type
tweet_count_blm
tweet_count_islam
factor(holiday)
● 0
▲ 1
■ 2