Chaudhary_work_5a

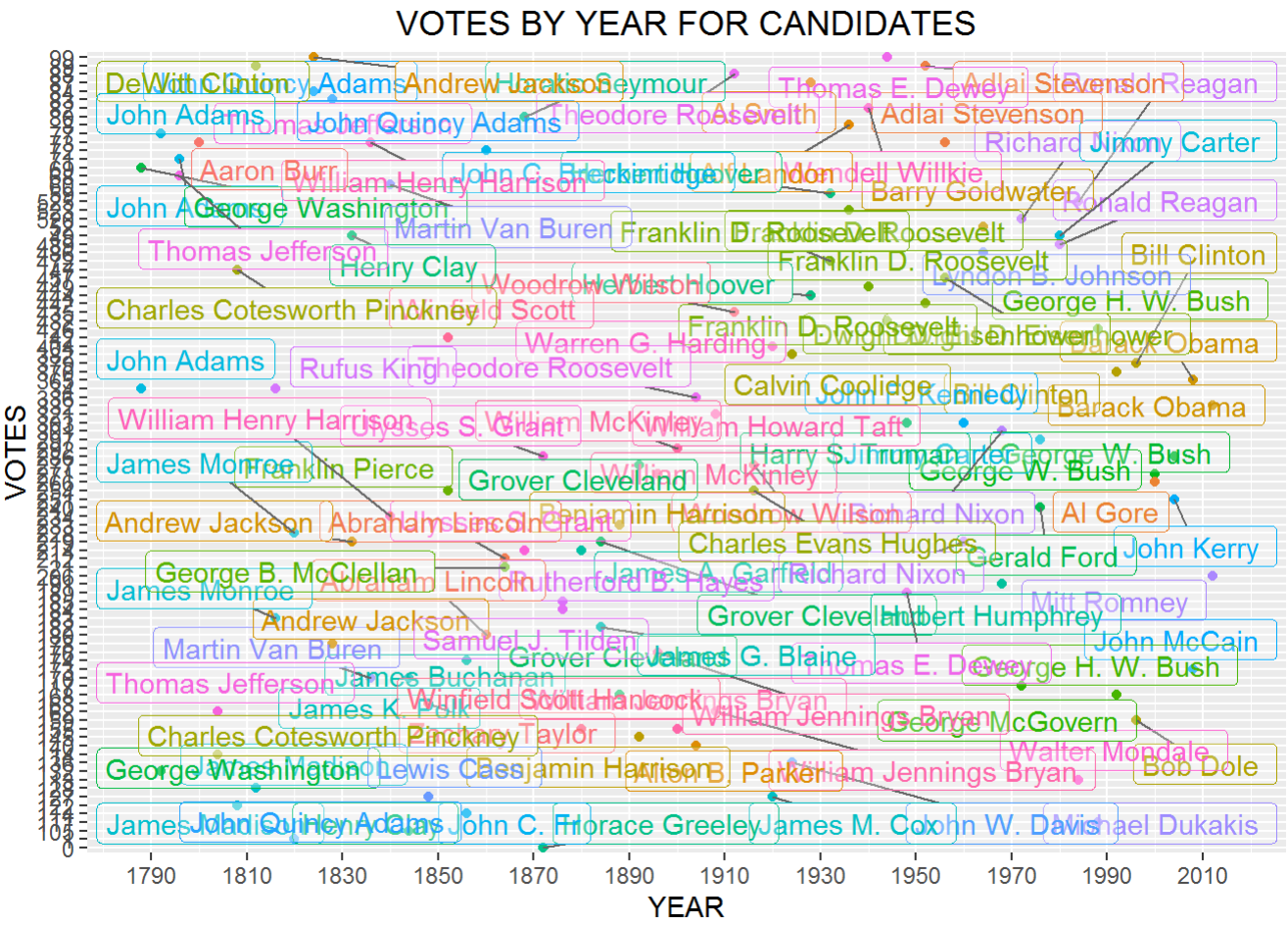
Jyoti Chaudhary

October 30, 2016

PROBLEM 1

Continuing the in-class assignment, take the electoral college table from the Wikipedia page on US presidential elections and create a tidy dataset. Each row should correspond to one election year and candidate, and the variables should be year, candidate, party, electoral votes, and whether or not the candidate won. Finally, use that tidy dataset to plot the electoral votes over time. That is, use year on the x-axis, electoral votes on the y-axis, and a dot for each candidate. Color the dots in some informative way.

```
url2 <- "https://en.wikipedia.org/wiki/United_States_presidential_election"

elections.list <-
  url2 %>%
  read_html() %>%
  html_nodes("table") %>%
  html_table(fill = TRUE)

elections <- elections.list[[3]]
names(elections) <- make.names(names(elections))

elections$Election.year <- as.integer(str_sub(elections$Election.year, end = 4))

elections <- elections %>% group_by(Election.year) %>% arrange(desc(Election.year))

colnames(elections) <- c("ORDER", "YEAR", "WINNER", "NON_WINNER")

elections1 <- elections %>% tbl_df %>% gather("WINNER/NON_WINNER", "CANDIDATE", 3:4)

#Remove the first row from elections1 dataset
elections1 <- tail(elections1,-1)

#Extract Party name into a separate column
elections1$PARTY <- str_extract(elections1$CANDIDATE, "\\([A-z -]+\\)") %>% str_replace_all("
[()]", "")

#Extract Vote count into a separate column
elections1$VOTES <- str_extract(elections1$CANDIDATE, "\\d+")

elections1$CANDIDATE <- str_extract(elections1$CANDIDATE, "[A-z. -]+")

##elections1$YEAR <- str_trim(elections1$YEAR)

## Plot electoral votes on Y-axis and YEAR on X-axis with colored dots for Candidates.
ggplot(data = elections1, aes(YEAR, VOTES, color = CANDIDATE)) + geom_point() + geom_line() +
  scale_x_continuous(breaks=seq(1750,2016,20)) +
  ggtitle("VOTES BY YEAR FOR CANDIDATES") + ggrepel::geom_label_repel(aes(label = CANDIDATE), al
pha = .25) + theme(legend.position = "none")
```



PROBLEM 2

2. For this problem you will need to install the readxl package
    a. Use ?read_excel to see how to use that function
    b. Copy the file “UN_MigrantStockByOriginAndDestination_2015.xlsx” to your folder for the assignment (and write your RMarkdown with the assumption that this file is in the same folder as the Rmd file)
    c. Use read_excel to load the sheet named “Table 16” into R. You’ll need both the sheet and the skip options.
    d. Convert the dataset to a tidy dataset, where the columns are origin, destination, and number of migrants. The dataset should have only countries, not regions or other categories
    e. Determine the 10 pairs of countries with the largest migrant flow.

```
## Reading excel file in stock_file1 dataframe
stock_file1 <- read_excel("UN_MigrantStockByOriginAndDestination_2015.xlsx", sheet = "Table 16",
 skip=15, na="", col_names = TRUE, col_types = NULL)

## Adding column names to first 5 dataframe columns
colnames(stock_file1)[1:5] = c("ORDER", "DESTINATION", "NOTES", "CODE", "TYPE_OF_DATA")

## using pipe to create another dataframe stock_file. Country code is filtered to include rows t
hat has less than 900 (I believe < 900 has all the countries). Filtered all the 'NA' on Migrants
 column. Arranged in desc order of Migrant values.
stock_file <- stock_file1 %>% filter(stock_file1$CODE < 900) %>% gather("ORIGIN", "MIGRANTS", 7:
240) %>%           filter(!is.na(MIGRANTS)) %>% select(ORIGIN, DESTINATION, MIGRANTS) %>% arrang
e(desc(MIGRANTS))

## 10 pair of countries with the larget migrant flow
head(stock_file, 10)
```

```
## # A tibble: 10 × 3
##                ORIGIN                                    DESTINATION
##                 <chr>                                          <chr>
## 1              Mexico                      United States of America
## 2               India                          United Arab Emirates
## 3  Russian Federation                                       Ukraine
## 4             Ukraine                            Russian Federation
## 5          Bangladesh                                          India
## 6         Other South                      United States of America
## 7          Kazakhstan                            Russian Federation
## 8  Russian Federation                                    Kazakhstan
## 9         Afghanistan                   Iran (Islamic Republic of)
## 10              China China, Hong Kong Special Administrative Region
## # ... with 1 more variables: MIGRANTS <dbl>
```