Problem Set 3

Due February 27, 10:00 AM (Before Class)

Instructions

1. The following questions should each be answered within an R script. Be sure to provide many comments in the script to facilitate grading. Undocumented code will not be graded.
2. Work on git. Fork the repository found at https://github.com/domlockett/PDS-PS3 and add your code, committing and pushing frequently. Use meaningful commit messages – these may affect your grade.
3. You may work in teams, but each student should develop their own R script. To be clear, there should be no copy and paste. Each keystroke in the assignment should be your own.
4. If you have any questions regarding the Problem Set, contact the TAs or use their office hours.
5. For students new to programming, this may take a while. Get started.
6. You will need to install `ggplot2` and `dplyr` to complete this dataset.

`ggplot2`

1. Finish the exercise we started in class on 2/11/2020:
   - Alabama, Arkansas, California, Colorado, Maine, Massachusetts, Minnesota, North Carolina, Oklahoma, Tennessee, Texas, Utah, Vermont, and Virginia will all hold their primraries on March 3
   - You have been assigned to create a visulaization of the state of the race for this date.
   - You will make a plot to show this.
   - In addition to the kinds of issues discussed above
     - Change to the minimial theme
     - Figure out how to change the axis labels and legends beyond the defaults
   - Visit https://ggplot2.tidyverse.org/reference/
2. Finish the exercise we started in class on 2/13/2020:
   - Re-organize the dataset so that there is only one row for each candidate-state dyad
   - Feel free to limit this down to only the relevant candidates
   - Compare the size of this dataset to our original dataset using the object_size command.

`tidyverse`

3. Now you are going to combine two datasets in order to observe how many endorsements each candidate recieved using **only** `dplyr` functions. First, create two new objects `polls` and `Endorsements`:

```
library(fivethirtyeight)
library(tidyverse)
polls <- read_csv('https://jmontgomery.github.io/PDS/Datasets/president
_primary_polls_feb2020.csv')
Endorsements <- endorsements_2020
```

   - Change the `Endorsements` variable name `endorsee` to `candidate_name`
   - Change the `Endorsements` dataframe into a `tibble` object.

- Filter the `poll` variable to only include the following 6 candidates: Amy Klobuchar, Bernard Sanders,Elizabeth Warren, Joseph R. Biden Jr., Michael Bloomberg, Pete Buttigieg **and** subset the dataset to the following five variables: `candidate_name, sample_size, start_date, party, pct`
- Compare the candidate names in the two datasets and find instances where the a candidates name is spelled differently i.e. Bernard vs. Bernie. Using only `dplyr` functions, make these the same across datasets.
- Now combine the two datasets by candidate name using `dplyr` (there will only be five candidates after joining).
- Create a variable which indicates the number of endorsements for each of the five candidates using `dplyr`.
- Plot the number of endorsement each of the 5 candidates have. Save your plot as an object `p`.
- Run the following code: `p + theme_dark()`. Notice how you can still customize your plot without rerunning the plot with new options. Save this plot in your forked repository
- Now, using the knowledge from the last step change the label of the X and Y axes to be more informative, add a title, and use your favorite theme. Save the plot in your forked repository.

Text-as-Data

4. For this assignment you will be analyzing Tweets from President Trump for various characteristics.

```
library(tidyverse)
#install.packages('tm')
library(tm)
#install.packages('lubridate')
library(lubridate)
#install.packages('wordcloud')
library(wordcloud)
tweets <- read_csv('https://politicaldatascience.com/PDS/Datasets/trump_tweets.csv')
```

- First separate the `created_at` variable into two new variables where the date and the time are in separate columns. **Then** report the range of dates that is in this dataset.
- Using `dplyr` subset the data to only include original tweets (remove retweets) and show the text of the President's **top 5** most popular and most retweeted tweets. (Hint: The `match` function can help you find the index once you identify the largest values.)
- Remove extraneous whitespace, remove numbers and punctuation, convert everything to lower case and remove the standard english stop words and include the following as stop words: c("see", "people",'new','want','one','even','must','need','done','back','just','going', 'know', 'can','said','like','many','like','realdonaldtrump').
- Now create a `wordcloud` to visualize the top 50 words the President uses in his tweets. Use only words that occur at least three times. Save the plot into your forked repository.
- Create a *document term matrix* called `DTM` that includes the argument `control = list(weighting = weightTfIdf)`
- Finally, report the 50 words with the the highest tf.idf scores using a lower frequency bound of .8.