

Applied Statistical Programming - Spring 2022

Problem Set 3

Due Wednesday, March 16, 10:00 AM (Before Class)

Instructions

1. The following questions should each be answered within an Rmarkdown file. Be sure to provide many comments in your code blocks to facilitate grading. Undocumented code will not be graded.
2. Work on git. Continue to work in the repository you forked from <https://github.com/johnsontr/AppliedStatisticalProgramming2022> and add your code for Problem Set 4. Commit and push frequently. Use meaningful commit messages because these will affect your grade.
3. You may work in teams, but each student should develop their own Rmarkdown file. To be clear, there should be no copy and paste. Each keystroke in the assignment should be your own.
4. For students new to programming, this may take a while. Get started.

tidyverse

Your task in this problem set is to combine two datasets in order to observe how many endorsements each candidate received using only `dplyr` functions. Use the same Presidential primary polls that were used for the in class worksheets on February 28 and March 2.

First, create two new objects `polls` and `Endorsements`. Then complete the following.

- 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 using `ggplot()`. Save your plot as an object `p`.

- Rerun the previous line as follows: `p + theme_dark()`. Notice how you can still customize your plot without rerunning the plot with new options.
- Now, using the knowledge from the last step change the label of the X and Y axes to be more informative, add a title. Save the plot in your forked repository.

```
#Use rename function to change variable name
endorsements <- rename(Endorsements, candidate_name=endorsee)
#as_tibble convert the data frame into tibble
endorsements <- as_tibble(endorsements)
#Endorsements is a tibble
class(endorsements)
```

```
## [1] "tbl_df"      "tbl"        "data.frame"
```

```
#Use filter to subset the polls data and use select to only store the variables we are interested
subset_polls <- polls %>% filter(candidate_name %in% c("Amy Klobuchar", "Bernard Sanders", "Elizabeth Warren"))
distinct(subset_polls, candidate_name) #Unique values in candidate_name
```

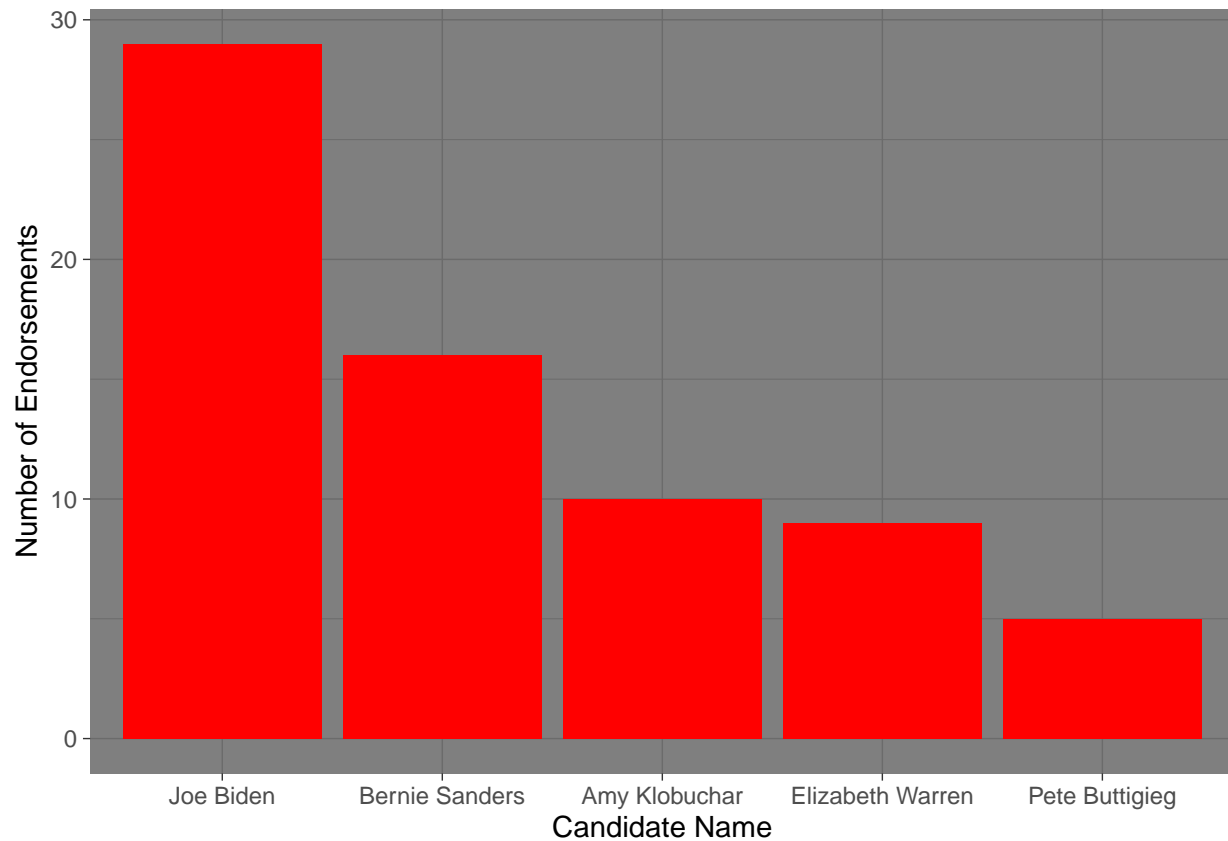
```
## # A tibble: 6 x 1
##   candidate_name
##   <chr>
## 1 Bernard Sanders
## 2 Pete Buttigieg
## 3 Joseph R. Biden Jr.
## 4 Amy Klobuchar
## 5 Elizabeth Warren
## 6 Michael Bloomberg
```

```
#In the same object as before, change names that are spelled differently in endorsements
subset_polls <- subset_polls %>% mutate(new_name = case_when(subset_polls$candidate_name == "Bernard Sanders" ~ "Bernard Sanders",
                                                             subset_polls$candidate_name == "Joseph R. Biden Jr." ~ "Joe Biden",
                                                             TRUE ~ as.character(subset_polls$candidate_name))) %>% rename(candidate_name=new_name)
#Use inner_join to merge two tibbles/data frame
join <- inner_join(endorsements, subset_polls, by="candidate_name")
distinct(join, candidate_name)
```

```
## # A tibble: 5 x 1
##   candidate_name
##   <chr>
## 1 Joe Biden
## 2 Bernie Sanders
## 3 Amy Klobuchar
## 4 Elizabeth Warren
## 5 Pete Buttigieg
```

```
#Filter the five candidates that we are interested. Apply summarise function to collapse and count the number of endorsements
endorsements_count <- endorsements %>% filter(candidate_name %in% c("Amy Klobuchar", "Bernie Sanders", "Elizabeth Warren", "Pete Buttigieg", "Joe Biden"))
join <- left_join(join, endorsements_count, by= "candidate_name")
#Use ggplot to create the plot and customize it
```

```
library(ggplot2)
p <- ggplot(data=endorsements_count , aes(x=reorder(candidate_name, -count), y=count)) +
  geom_bar(stat="identity", fill="red")+ labs(x="Candidate Name", y="Number of Endorsements")
p + theme_dark()
```



```
ggsave("plot.pdf", p)
```

```
## Saving 6.5 x 4.5 in image
```

Text-as-Data with tidyverse

For this question you will be analyzing Tweets from President Trump for various characteristics. Load in the following packages and data:

```
library(tidyverse)
library(tm)
```

```
## Loading required package: NLP
```

```
##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##   annotate
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##   date, intersect, setdiff, union
```

```
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
library(readr)
trump_tweets_url <- 'https://politicaldatascience.com/PDS/Datasets/trump_tweets.csv'
tweets <- read_csv(trump_tweets_url)
```

```
## Rows: 32974 Columns: 6
```

```
## -- Column specification -----
## Delimiter: ","
## chr (3): source, text, created_at
## dbl (2): retweet_count, favorite_count
## lgl (1): is_retweet
```

```
##
## 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.
```

- First separate the `created_at` variable into two new variables where the date and the time are in separate columns. After you do that, 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.)
- Create a *corpus* of the tweet content and put this into the object **Corpus** using the `tm` (text mining) package. (Hint: Do the assigned readings.)
- Remove extraneous whitespace, remove numbers and punctuation, convert everything to lower case and remove 'stop words' that have little substantive meaning (the, a, it).
- 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. Display the plot with words in random order and use 50 random colors. 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.

```
#Use the separate function to separate the two values in two variables. Then use the as.Date function to
s_tweets <- tweets %>% separate(created_at, into = c("sdate", "stime"), sep = " ") %>% mutate(dates = as.Date(sdate))
#Report the ranges of dates with summarise function and min and max arguments
s_tweets %>% summarise(min = min(dates),
                        max = max(dates))
```

```
## # A tibble: 1 x 2
##   min      max
##   <date>   <date>
## 1 2014-01-01 2020-02-14
```

```
#Use slice_max and filter to show the top 5 most popular and retweeted tweets.
top5_rt <- s_tweets %>% filter(is_retweet==FALSE) %>% slice_max(retweet_count, n = 5)
top5_fav <- s_tweets %>% filter(is_retweet==FALSE) %>% slice_max(favorite_count, n = 5)
knitr::kable(top5_rt$text)
```

```
x
```

```
#FraudNewsCNN #FNN https://t.co/WYUnHjjUjg
TODAY WE MAKE AMERICA GREAT AGAIN!
Why would Kim Jong-un insult me by calling me "old" when I would NEVER call him "short and fat?"
Oh well I try so hard to be his friend - and maybe someday that will happen!
AAPRockyreleasedfromprisonandonhiswayhometotheUnitedStatesfromSweden.ItwasRockyWeekgethomeASAPAAP!
Such a beautiful and important evening! The forgotten man and woman will never be forgotten again. We
will all come together as never before
```

```
knitr::kable(top5_fav$text)
```

```
x
```

```
AAPRockyreleasedfromprisonandonhiswayhometotheUnitedStatesfromSweden.ItwasRockyWeekgethomeASAPAAP!
https://t.co/VXeKiVzpTf
All is well! Missiles launched from Iran at two military bases located in Iraq. Assessment of casualties &
damages taking place now. So far so good! We have the most powerful and well equipped military
anywhere in the world by far! I will be making a statement tomorrow morning.
MERRY CHRISTMAS!
Kobe Bryant despite being one of the truly great basketball players of all time was just getting started in
life. He loved his family so much and had such strong passion for the future. The loss of his beautiful
daughter Gianna makes this moment even more devastating...
```

```
s_tweets$text <- gsub(" ?(f|ht)tp(s?):/(.*)[.][a-z]+", "", s_tweets$text) #Remove URLs from the beginning
#Use Vcorpus function to create a corpus of Trump's tweets
Corpus_trump <- VCorpus(VectorSource(s_tweets$text))
inspect(Corpus_trump[[1]])
```

```
## <<PlainTextDocument>>
## Metadata: 7
## Content: chars: 120
##
## RT @DailyCaller: 'Why Would I Not:' Chiefs' Bashaud Breeland Looking Forward To WH Visit After Super
```



```
dtm <- removeSparseTerms(dtm, .9)
#Find the 50 words with highest tf.idf
dtm <- as.matrix(dtm)
```