Applied Statistical Programming - ggplot

2/28/2022

Write the R code to answer the following questions. Write the code, and then show what the computer returns when that code is run. Thoroughly comment your solutions.

You have until the beginning of class 3/2 at 10:00am to complete the assignment below. You may use R, but not any online R documentation. Submit the Rmarkdown and the knitted PDF to Canvas. Have one group member submit the activity with all group members listed at the top.

Figuring out the Competition

You've been hired by a campaign to do some data analysis during the primary stage of an election. The campaign wants to understand competitiveness of certain candidates under different general election scenarios. You will plot some summary features of the provided `primaryPolls` data using `ggplot()`.

The data is associated with 2020 Democratic primary elections. Polling results for 38 states are provided. You will create a visualization of the state of the race using this data. For three states of your choosing, generate a summary figure that visualizes the support for each candidate in that state. Each plot must include:

- a title,
- a subtitle,
- labeled axes,
- a legend for the candidates, and
- a source attribution to the GitHub URL for the data.

```
# Remove eval=FALSE to have this code block run.

# Load library dependencies
library(dplyr)
library(tidyr)
library(readr)

# Define path to the data
dataURL <- "https://jmontgomery.github.io/PDS/Datasets/president_primary_polls_feb2020.csv"
# Load the data
primaryPolls <- read_csv(dataURL)
```

```
## Rows: 16661 Columns: 33
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (21): state, pollster, sponsors, display_name, pollster_rating_name, fte...
```

```
## dbl  (8): question_id, poll_id, cycle, pollster_id, pollster_rating_id, samp...
## lgl  (3): internal, tracking, nationwide_batch
##
## 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.
```

```
# Format the date
primaryPolls$start_date <- as.Date(primaryPolls$start_date, "%m/%d/%y")
```

NOTE 1: primaryPolls's unit of analysis (i.e., each observation/row) is on the candidate level. Notable variables:

- **start_date**: Survey's initiation date.
- **end_date**: Survey's termination date.
- **party**: Political party of candidate (this polling data *does* include some Republican candidates).
- **candidate_name**: name of candidate.
- **pct**: estimated proportion of the population that supports the candidate.

STEP 1: Fix errors in and consolidate the data.

- Drop Republican candidates.

```
unique(primaryPolls$party)
```

```
## [1] "DEM" "REP"
```

```
primaryPolls <- primaryPolls[which(primaryPolls$party == "DEM"), ]
unique(primaryPolls$party)
```

```
## [1] "DEM"
```

```
# Only Democratic candidates remain.
```

- Fix Julian Castro's name. I think the accents didn't load into the original dataset, so I don't include the accent.

```
unique(primaryPolls$candidate_name[primaryPolls$answer == "Castro"])
```

```
## [1] "Juli<cc><c1>n Castro"
```

```
sum(is.na(primaryPolls$answer))
```

```
## [1] 0
```

```
sum(is.na(primaryPolls$candidate_name))
```

```
## [1] 0
```

```
primaryPolls[primaryPolls$answer == "Castro", "candidate_name"] <- "Julian Castro"

unique(primaryPolls$candidate_name[primaryPolls$answer == "Castro"])
```

```
## [1] "Julian Castro"
```

```
# candidate_name now uses all unique values.
```

- Ensure that there are a manageable number of Democratic candidates.

```
length(unique(primaryPolls$candidate_name))
```

```
## [1] 59
```

```
# Holy cow! Pollsters really included everyone and their mom in their polls, didn't they?
# We can't have 59 Democratic candidates on our graphs!
```

Step 2: Separate out data for each state.

* New Hampshire.

```
newhampshire <- primaryPolls[which(primaryPolls$state == "New Hampshire"),]
```

* Nevada.

```
nevada <- primaryPolls[which(primaryPolls$state == "Nevada"),]
```

* South Carolina.

```
southcarolina <- primaryPolls[which(primaryPolls$state == "South Carolina"),]
```