TA Session 11: Grouped Analysis and Combining Data Frames

Harris Coding Camp

Summer 2024

General Guidelines

You may encounter some functions we did not cover in the lectures. This will give you some practice on how to use a new function for the first time. You can try following steps:

1. Start by typing `?new_function` in your Console to open up the help page.
2. Read the help page of this `new_function`. The description might be a bit technical for now. That's OK. Pay attention to the Usage and Arguments, especially the argument `x` or `x,y` (when two arguments are required).
3. At the bottom of the help page, there are a few examples. Run the first few lines to see how it works.
4. Apply it in your questions.

It is highly likely that you will encounter error messages while doing this exercise. Here are a few steps that might help get you through it:

1. Locate which line is causing this error first.
2. Check if you have a typo in the code. Sometimes your group members can spot a typo faster than you.
3. If you enter the code without any typo, try googling the error message. Scroll through the top few links see if any of them helps.
4. Try working on the next few questions while waiting for help by TAs.

Background and data

First, follow the tweet thread and you'll see that Prof. Damon Jones, of Harris, gets that data and does some analysis. In this exercise, you're going to follow his lead and dig into traffic stop data from the University of Chicago Police Department, one of the largest private police forces in the world.

Download the data here. You can save the file directly from your browser using `ctrl + s` or `cmd + s`. Alternatively, you can read the csv directly from the internet using the link https://github.com/harris-coding-lab/harris-coding-lab.github.io/raw/master/data/data_traffic.csv

Next, we will examine data on turnout in U.S. presidential elections from McDonald and Popkin (2001).[1] The data from the original article have been updated through 2016. We are providing you with two data sets—`mcdonald1.csv` and `mcdonald2.csv`—both of which are available on Canvas.

[1] For more information on their study, see McDonald, Michael P. and Samuel L. Popkin. 2001. "The Myth of the Vanishing Voter." *American Political Science Review* 95(4): 963-974.

Warm-up

1. Open a new `Rmd` and save it in your coding lab folder; if you downloaded the data, move your data file to your preferred data location.
2. In your `Rmd`, write code to load your packages. If you load packages in the console, you will get an error when you knit because knitting starts a fresh R session.
3. Load `data_traffic.csv` and assign it to the name `traffic_data`. This data was scrapped from the UCPD website and partially cleaned by Prof. Jones.
4. Recall that `group_by()` operates silently. Below I create a new data frame called grouped_data.

```
grouped_data <-
  traffic_data %>%
    group_by(Race, Gender)
```

   a. How can you tell `grouped_data` is different from `traffic_data`?
   b. How many groups (Race-Gender pairs) are in the data? (This information should be available without writing additional code!)
   c. Without running the code, predict the dimensions (number of rows by number of columns) of the tibbles created by `traffic_data %>% summarize(n = n())` and `grouped_data %>% summarize(n = n())`.
   d. Now check you intuition by running the code.

5. Use `group_by()` and `summarize()` to recreate the following table.

```
#> # A tibble: 6 x 2
#>   Race                                    n
#>   <chr>                               <int>
#> 1 African American                     3278
#> 2 American Indian/Alaskan Native         12
#> 3 Asian                                 226
#> 4 Caucasian                             741
#> 5 Hispanic                              217
#> 6 Native Hawaiian/Other Pacific Islander    4
```

Moving beyond counts

1. Raw counts are okay, but frequencies (or proportions) are easier to compare across data sets. Add a column with frequencies and assign the new tibble to the name `traffic_stop_freq`. The result should be identical to Prof. Jones's analysis on twitter.

   Try on your own first. If you're not sure how to add a frequency though, you could google "add a proportion to count with tidyverse" and find this stackoverflow post. Follow the advice of the number one answer. The green checkmark and large number of upvotes indicate the answer is likely reliable.

2. The frequencies out of context are not super insightful. What additional information do we need to argue the police are disproportionately stopping members of a certain group? (Hint: Prof. Jones shares the information in his tweets.)[2]
3. For the problem above, your group members tried the following code. Explain why the frequencies are all 1.[3]

```
traffic_stop_freq_bad <-
traffic_data %>%
```

[2] To be fair, even with this information, this is crude evidence that can be explained away in any number of ways. One job of a policy analyst is to bring together evidence from a variety of sources to better understand the issue.

[3] Hint: This is a lesson about `group_by()`!

```
  group_by(Race) %>%
  summarize(n = n(),
            freq = n / sum(n))

traffic_stop_freq_bad
```

4. Now we want to go a step further.[4] Do outcomes differ by race? In the first code block below, I provide code so you can visualize disposition by race. "Disposition" is police jargon that means the current status or final outcome of a police interaction.

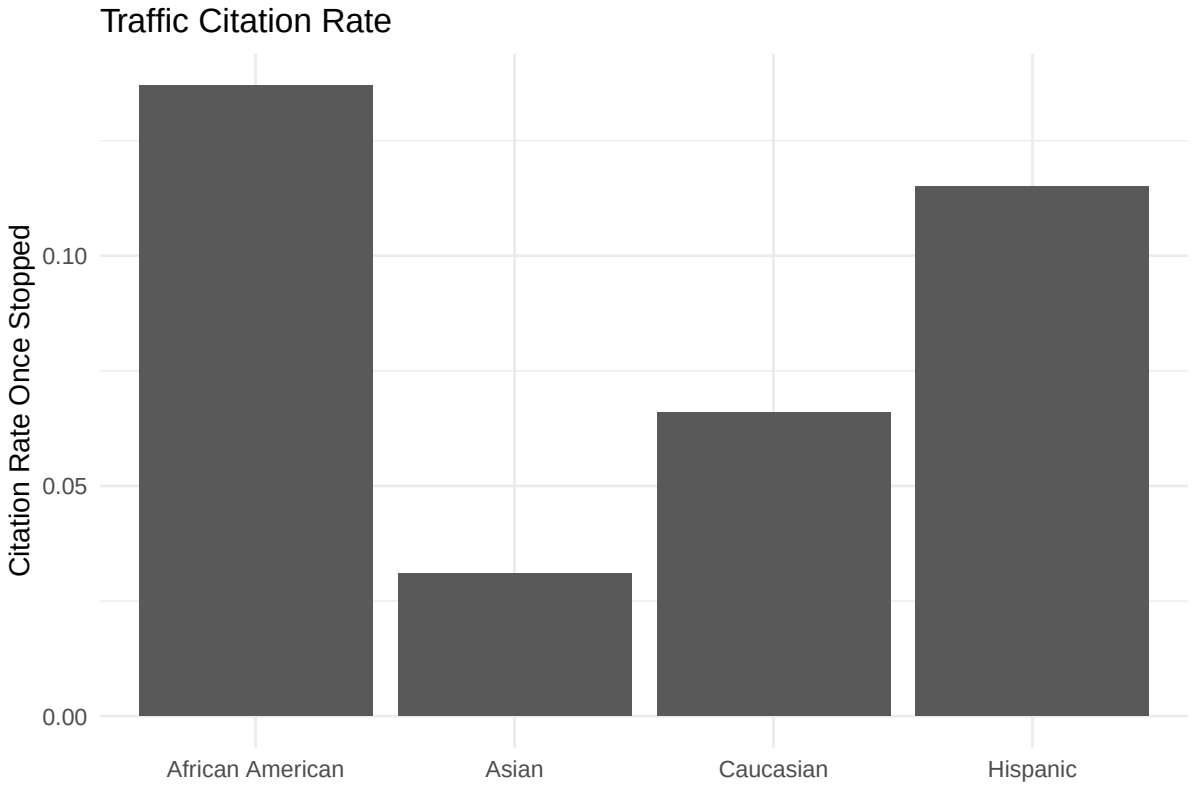
```
citation_strings <- c("citation issued", "citations issued", "citation  issued" )

arrest_strings <- c("citation issued, arrested on active warrant",
                "citation issued; arrested on warrant",
                "arrested by cpd",
                "arrested on warrant",
                "arrested",
                "arrest")

disposition_by_race <-
    traffic_data %>%
      mutate(Disposition = str_to_lower(Disposition),
             Disposition = case_when(Disposition %in% citation_strings ~ "citation",
                                     Disposition %in% arrest_strings ~ "arrest",
                                     TRUE ~ Disposition)) %>%
      count(Race, Disposition) %>%
      group_by(Race) %>%
      mutate(freq = round(n / sum(n), 3))


disposition_by_race %>%
  filter(n > 5, Disposition == "citation") %>%
  ggplot(aes(y = freq, x = Race)) +
  geom_col() +
  labs(y = "Citation Rate Once Stopped", x = "", title = "Traffic Citation Rate") +
  theme_minimal()
```

[4]The analysis that follows is partially inspired by Eric Langowski, a Harris alum, who was also inspired to investigate by the existence of this data (You may have seen Prof. Jones retweet him at the end of the thread.)



Let's break down how we got to this code. First, I ran `traffic_data %>% count(Race, Disposition)` and noticed that we have a lot of variety in how officers enter information into the system.[5] I knew I could deal with some of the issue by standardizing capitalization.

a. In the console, try out `str_to_lower(...)` by replacing the `...` with different strings. The name may be clear enough, but what does `str_to_lower()` do?[6]

After using `mutate` with `str_to_lower()`, I piped into `count()` again and looked for strings that represent the same `Disposition`. I stored terms in character vectors (e.g. `citation_strings`). The purpose is to make the `case_when()` easier to code and read. Once I got that right, I added frequencies to finalize `disposition_by_race`.

5. To make the graph, I first tried to get all the disposition data on the same plot.

```
disposition_by_race %>%
  ggplot(aes(y = freq, x = Race, fill = Disposition)) +
  geom_col()
```

By default, the bar graph is stacked. Look at the resulting graph and discuss the pros and cons of this plot with your group.

6. I decided I would focus on citations only and added the `filter(n > 5, Disposition == "citation")` to the code.[7] What is the impact of filtering based on `n > 5`? Would you make the same choice? This question doesn't have a "right" answer. You should try different options and reflect.

[5] Try it yourself!
[6] This code comes from the `stringr` package. Checkout `?str_to_lower` to learn about some related functions.
[7] Notice that I get the data exactly how I want it using `dplyr` verbs and then try to make the graph.

Voter Participation Trends in the United States

In this part of the exercise, we will examine trends over time in voter turnout in U.S. elections. Electoral participation is considered an extremely important indicator of democratic performance (e.g., Powell 1982). As a result, declines in voter participation within a polity are often concerning to scholars and political observers. For instance, Rosenstone and Hansen (1993), commenting on a decrease in turnout, noted that the "decline of citizen involvement in government has yielded a politically engaged class that is not only growing smaller and smaller but is also less and less representative.''

For this data exercise, we will examine data on turnout in U.S. presidential elections from McDonald and Popkin (2001).[8] The data from the original article have been updated through 2016. We are providing you with two data sets—`mcdonald1.csv` and `mcdonald2.csv`—both of which are available on Canvas.

The `mcdonald1.csv` data set contains the following variables:

Name	Description
`year`	Election year
`votes_higho`	Votes cast for the highest office on the ballot (in thousands)
`vap`	Voting-age population living in the U.S. (in thousands)

The `mcdonald2.csv` data set contains the following variables:

Name	Description
`year`	Election year
`noncit_pop`	Non-citizen population living in the U.S. (in thousands)
`overseas_el`	Overseas eligible population (in thousands)
`felon_inel`	Population ineligible due to felony conviction (in thousands)

1. Read the `mcdonald1.csv` data set into R. We will separately examine voting patterns in presidential and midterm election years. So, create a variable in your data set called `midterm`, which is coded = 0 for presidential election years and coded = 1 for midterm election years.[9] *Hint:* the `seq()` function and the `%in%` operator in R will likely prove useful.

Scholars traditionally measured turnout by examining the number of votes cast in an election as a share of the voting-age resident population. Create such a variable in your data set and call it `turnout_vap`.

a. Based on the variable you just created, what was the average turnout rate in presidential elections for this time period? What was the average turnout rate in midterm elections? Is turnout generally higher in midterm or presidential elections?

b. Calculate the average turnout rate for each of the following time periods: 1952–1968, 1972–1988, and 1992–2016.

c. Graph the turnout rate over time for presidential elections only. Make sure that the axes are labeled and that your plot has an appropriate title.

d. In 1-2 sentences, briefly describe the over-time patterns in presidential turnout as a share of the voting-age resident population.

[8] For more information on their study, see McDonald, Michael P. and Samuel L. Popkin. 2001. "The Myth of the Vanishing Voter." *American Political Science Review* 95(4): 963-974.

[9] The data sets you are provided contain only presidential and midterm elections. Presidential elections occur every four years (e.g., $1948, 1952, \ldots, 2016$), and midterm elections occur every four years too (e.g., $1950, 1954, \ldots, 2018$).

2. The main insight from McDonald and Popkin (2001) is that using the voting-age population (VAP) living in the U.S. as the denominator for a turnout measure is problematic. Specifically, the VAP includes non-citizens and felons who are not eligible to vote in these elections, and it excludes citizens residing overseas who are eligible to vote. Read the `mcdonald2.csv` data set into R and merge/join it into the data set you have been analyzing. Then, create a new turnout rate variable called `turnout_vep`, which is turnout as a share of the voting-eligible population (VEP). The voting-eligible population accounts for ineligible non-citizens and felons as well as eligible citizens residing overseas.

a. What was the average turnout rate (based on your new VEP measure) in presidential elections for this time period? What was the average turnout rate in midterm elections?

b. Calculate the average turnout rate (again, using the VEP measure) for each of the following time periods: 1952–1968, 1972–1988, and 1992–2016.

c. Create a plot in which you graph two separate times series on the same plot. The first series is presidential turnout as a share of VAP, which you plotted for Part 1, and the second series is presidential turnout as a share of VEP. When using base R, in addition to the `plot()` command, you should also use the `points()` command to graph the second series on the same plot. Finally, use `text()` to label each series on the graph. As always, make sure that the axes are labeled and that your plot has an appropriate title.

d. Briefly describe over-time patterns in presidential turnout as a share of the voting-eligible population. In what ways, if any, are these over-time patterns different than the over-time patterns based on the VAP measure? Are you now more or less concerned about the health of U.S. democracy compared to your assessment after Part 1?