Prep2S24

YourNameGoesHere

2024-02-08

Reminder: Prep assignments are to be completed individually. Upload a final copy of the .Qmd and renamed .pdf to your private repo, and submit the renamed pdf to Gradescope before Sunday, Feb. 11th at midnight (11:59 pm is what Gradescope shows).

Reading

The associated reading for the week is Chapter 4, Chapter 5, Chapter 6 (skip 6.4) and Sections 8.3 and 8.4. This reading explores major functions in wrangling data, including reshaping data. There are many commands here to learn about - do your best to develop a sense of what they each do, and we will build on that by using them for the rest of the semester. You don't need to memorize them all.

Remember, I recommend you code along with the book examples. You can try out the code yourself - just be sure to load the mdsr package and any other packages referenced. You can get the code in R script files (basically, files of just R code, not like a .Rmd or .Qmd) from the book website.

1 - Some basics

Many different data wrangling commands are covered in these chapters. Identify the command you'd use for each of the operations below.

part a - Add the average of 3 variables to the data set as a new variable.

Solution:

part b - Keep only 4 columns of a data frame in a new data set.

Solution:

part c - Choose observations that match a particular category of a categorical variable to keep in a new data set.

Solution:

part d - Combine two data sets based on common variables where all rows from the first are returned, along with any matches in the second.

Solution:

2 - NYC Flights

In Section 5.1, the `flights` and `airlines` tables within the `nycflights13` package are joined together.

> part a - Recreate the `flights_joined` dataset from Section 5.1, being sure to *glimpse* the data in the Console (or via the code chunk) to verify the join worked.

Solution:

> part b - Now, starting from `flights_joined`, create a new dataset `flights_short` that does the following:

- creates a new variable, `distance_km`, which is distance in kilometers (note that 1 mile is about 1.6 kilometers)
- keeps only the variables: `name`, `flight`, `arr_delay`, and `distance_km` and
- keeps only observations where the distance is less than 480 kilometers (300 miles).

Solution:

> part c - Using the functions introduced in Section 4.1.4, compute the number of flights (call this `N`), the average arrival delay (call this `avg_arr_delay`), and the average distance in kilometers (call this `avg_dist_km`) among these flights with distances less than 480 km (i.e. working off of `flights_short`), grouping by the carrier name. Sort the results in descending order based on `avg_arr_delay`. Save the results in a tibble object called `delay_summary`, and display the table.

Solution:

> part d - Rename the four columns in the `delay_summary` data table to `Airline`, `"Total flights under 480 km"`, `"Average arrival delay (mins)"` and `"Average distance (km)"`, respectively, then use `kable(booktabs = TRUE, digits = 0)` to make the final table output in the pdf close to publication quality.

Solution:

3 - Baby names - Variant of 6.2.5 example

part a - Working with the `babynames` data in the **babynames** package, create a dataset `recent_names` that only includes years 2003 to 2017 (giving us the most recent 15 years of data).

Solution:

part b - Following the code presented in Section 6.2.5, create a dataset called `recentnames_summary` that summarizes the total number of people in recent history (years 2003 to 2017) with each name, grouped by sex.

Solution:

part c - Now, following the third and fourth code chunks presented in Section 6.2.5, reshape or *pivot* the summary data from *long* format to *wide* format. Only keep observations where more than 8,000 babies have been named in each sex (`M` and `F`), and find the smaller of the two ratios `M / F` and `F / M` to identify the top three sex-balanced names (and only the top three!). Save the wide data as `recentnames_balanced_wide`. Display the table.

Solution:

part d - Finally, use `pivot_longer()` to put the dataset back into *long* form. Call this dataset `recentnames_balanced` and display the table.

Solution:

4 - Ethics

Each subsection of Section 8.4 discusses an ethical scenario and ends with one or more questions. Consider the subsection 8.4.6 on "Reproducible spreadsheet analysis".

Write two or three sentences reflecting on how using RMarkdown would help avoid some of the issues described in this scenario, or at least make them easier to spot.

Solution: