Homework 2: Data Cleaning and Summarizing

Overview

This homework is aimed toward practicing using functions for cleaning and summarizing data.

Relevant Module Topics:

- Data frame manipulation and wrangling
- Data subsets and summaries

Requirements

- Do not create any variables outside of `{dplyr}` functions.
- Do not create unnecessary objects.
- Create code what will execute from beginning to end and be reproducible.
- All problems can be answered using functions from the listed modules.
- Focus on code readability: use single spacing between operators, one pipe per line, prioritize lower casing, adding sectional guidance notes, etc.

Assignment Trade-offs

I prefer to make homework assignments that supplement the team project and therefor expect your question responses to represent reasonable attempts to work with your data. If your work does not evidence reasonable work toward the project, on a case-by-case basis, I reserve the right to require that you work with data other than that for your team.

AI Usage

I assume that you are completing readings and referring to modules before consulting outside sources. Grading will be based on this assumption and solutions provided in course materials rather than those obtained from online or AI sources. If you reference course materials to help you, you should experience few, if any, coding errors and should not need to use an AI. Course website content as well as homework question content cannot be submitted to an AI in order to obtain coded solutions. If, however, you have difficulty using functions beyond the course examples or you need to troubleshoot an error your code is throwing and you cannot figure out whey, you may use an AI to help you if you document how you used it. As always, understand that you are still responsible for understanding the code you submit as your work.

Problems

Use your `dataviz-exercsies` RStudio project, not your Team project. In order to move your project along, you are encouraged to a data file from your team project if you copy that file to your `dataviz-exercsies` project. If for some reason you are unprepared to use data from your team project, use **the cms-top-all-time-2023-swim.csv file**.

1. Create a Data Cleaning Script

Start with the `starter_script.R` to create a data-cleaning version named something like: `your_name_data_cleaning.R`. Save the script in the proper project directory. Complete the header content, add your authorship, etc. so that you practice creating clean readable scripts.

2. Clean Data

Note: Responses will differ as projects may require different responses.

Your script should include the following steps:

- *Load Libraries*: Load libraries needed for your cleaning. Be mindful of any function masking based on loading order.
- *Read Data*: Provide code to read the data file using `{here}` for your path.
- *Clean the Data*: Perform any necessary data cleaning tasks (e.g., convert character to numeric, convert numeric to character, rename variable columns to better names, remove missing values from vectors, convert data types, add new variables that might be relevant or helpful (e.g., additional metrics). Add comments to explain your coding steps. Think `select()`, `filter()`, `mutate()`, and `relocate()`.
- *Save Data*: Write the cleaned data as `data_cleaned.Rds` in the `data/processed/` directory.

3. Create a Data Summary Script

Create `your_name_data_summary.R` from the `starter_script.R` to summarize data. Save the script in the proper directory. Complete the header content, add your authorship, etc.

4. Summarize Cleaned Data

Whereas some visualization present all data, other visualizations present aggregated forms of data. Think about your project data and identify ways that you might present aggregated summaries of data (e.g., totals/sums, counts/instances, means, medians, minimum values, maximum values, variance metrics, etc.). Consider ways that might be relevant for grouping your data for building those summaries.

Your script should include the following steps:

- *Load Libraries*: Load libraries needed for your cleaning/summarizing. Be mindful of any function masking based on loading order.

- *Read Data*: Provide code to read your cleaned data file using `{here}` for your path.
- *Group, Summarize, and Saving Data*: Consider **2** ways for which you might aggregate and summarize your data. For each approach, document the goal so that the script section is clear, perform the grouping summary, and save an `.Rds` file of the summarized data frame using a file name appropriate for your given goal. As good practice, please follow naming conventions provided in the team project sample materials and agreed upon by your team.

5. Summarize Using `across()`

Group, Summarize, and Saving Data: *If you did not already use* `across()` in your 2 data-summary approaches add code to your script in order to perform a data summary by pairing `summarize()` and `across()`. This pair of functions is extremely useful when you need to *cleanly* summarize a single metric across many variables or summarize a single variable across many metrics. Document your goal and save the data frame with an appropriate name.

6. Git and GitHub

Stage, Commit, and Push your two `.R` files to your remote.

Upload your two `.R` files to: https://ln5.sync.com/dl/a038628f0/wwfifjxk-f7rfshin-rkedi3y8-77f9zaii