22MAP500 Coursework

- Getting started
- Instructions
- Files to submit
- Assessment criteria
    - 1. Content & Layout [50 points]
    - 2. Reproducibility [10 points]
    - 3. Tidyverse syntax [10 points]
    - 4. Figure formatting [15 points]
    - 5. Data [5 points]
    - 6. Coding style [10 points]
    - Bonus points [5 points]

Getting started

1. Create a new RStudio project.
2. Within your project folder, create a folder `data`.
3. Download the two data sets from the submission point on Learn and save them in the `data` subfolder of your RStudio project folder.
4. Under no circumstances should you modify the downloaded files in `data`, neither manually nor through your code. Your R script should read the raw data directly from this folder.

Instructions

In this coursework, you will analyse data about bike rides undertaken by users of Capital Bikeshare (CaBi) – a publicly-owned bicycle sharing system that serves areas in and around Washington DC in the United States. More information can be found at https://ride.capitalbikeshare.com/ (https://ride.capitalbikeshare.com/) and https://ride.capitalbikeshare.com/system-data (https://ride.capitalbikeshare.com/system-data).

You will write a report about the provided data. The report will be in the form of an RMarkdown notebook. It should be structured into the following sections.

1. ~~Introduction~~
    - Motivate your work, i.e.
        - briefly describe the topic,
        - briefly explain why it is worth studying.
2. ~~Data~~
    - Describe the data, i.e.
        - briefly explain the source of the raw data,
        - summarise the variables contained in the data set after any steps needed for cleaning and tidying the data,
        - if you are excluding any observations from your analysis, justify this here.
3. Questions
    - State five *interesting* research questions raised by the data.
        - All five questions should be significantly qualitatively different, i.e. two questions should not just differ by swapping out one variable for another.
        - Remember that we can only deal with *descriptive* and *exploratory* questions!

- **Hint:** *Do read the website https://capitalbikeshare.com (https://capitalbikeshare.com) for further information about the scheme such as the pricing structure to help find interesting research questions. However, note that there has been a change: the website states that members can take unlimited rides of duration shorter than 45 minutes, but until 2021, this was limited to rides of duration shorter than 30 minutes.*
- For each question, explain how you operationalise it (see the slides from the Day-2 lecture if you don't remember what "operationalisation" means).

4. Analysis
   - Answer the five operationalised research questions.
     - Each answer should be supported by one or more *meaningful* visualisations (i.e. figures). In particular, "meaningful" means that the answer to the question should not just consist of one or two numbers (because in this case, one would not use a visualisation).
     - For each question, at least one of the visualisations should rely on data from more than one variable (these can be variables which already existed in the original data set or variables which you have newly created by transforming existing variables).
   - Over the course of answering your five questions, you should
     a. make use of at least six variables contained in the original data;
     b. use each of the following elements at least once:
        i. faceting;
        ii. meaningful annotation;
        iii. discrete colour or fill scales (i.e. discrete variable mapped to the colour or fill aesthetic);
        iv. continuous colour or fill scales (i.e. a continuous variable mapped to the colour or fill aesthetic).
5. Conclusion
   - Summarise your results.
   - Mention at least one further question raised by the results of your analysis.

The output should be a full-text report, i.e. not just bullet points, and written with enough detail and explanation that it can be easily(!) understood by someone who does not know the topic and has not seen the data set. No code or warning, error or other messages produced by R should be visible in the knitted (.html) document! You do not need to cite any other literature (but this is also not forbidden).

Files to submit

You will submit your coursework as a single R notebook (i.e. `.Rmd` file) which can be rendered ("knitted") to an `.html` document. Specifically, **submit all of the following**

- your R notebook (i.e. the `.Rmd` file),
- the rendered `.html` version of your notebook (in case there are any problems knitting your `.Rmd` during marking),
- any images (e.g. `.jpg` files) loaded by in your notebook (You do not need to include any images. However if you do, make sure to read the relevant instructions in the "Reproducibility" section below).

Do *not* include any identifying information such as your name or student ID in the submitted documents!

Do *not* submit the data set(s)!

Do *not* submit your entire RStudio project!

Do *not* compress your files in a `.zip` (or similar) archive!

Assessment criteria

To obtain full marks, your submitted R notebook must satisfy the following conditions. The points for Criteria 2–6 are pro-rated to the number of analyses performed in Section 4, e.g. if you only fully answer three out of five questions, then you can achieve at most 9 out of 15 points for figure formatting.

1. Content & Layout [50 points]

- Half the marks are awarded for content and layout as instructed above, i.e. for writing an ~~introduction~~ **[1]** and ~~conclusion~~ **[1]**, for ~~explaining the data set~~ **[3]**, for developing suitable research questions and explaining how these are operationalised **[15]**, and for providing ~~answers~~ to the questions **[30]**.

~~2. Reproducibility [10 points]~~

- Your notebook must be able to be "knit" on another computer which is running the latest versions of R and RStudio (with only those R packages mentioned in the lecture notes and computer labs) which has access to the same data files organised in the same folder structure as mentioned above. In particular,
  - your project folder must contain a folder `data` which holds the original data files (and only these!) as instructed in "Getting started" above,
  - your notebook must specify the paths to these data using relative – not absolute – paths via the `here()` function,
  - your notebook must not load any *additional* data sets,
  - inclusion of images (e.g. `.jpg` files) is allowed as long as these are submitted alongside the `.Rmd` file and as long as the notebook loads these via relative file paths from a subfolder `figures` within the RStudio project folder as discussed in Chapter 1 of the R lecture notes,
  - any data wrangling/data cleaning must be done via the R code inside your notebook,

~~3. Tidyverse syntax [10 points]~~

- All importation, cleaning, transformation and visualisation of data sets must be achieved using **tidyverse** packages and syntax such as those taught Chapters 4 to 9. In particular:
  - Data sets must be stored in `tibble` objects rather than base R `data.frame` (or `data.table`) objects.
  - Use **tidyverse** verbs such as `filter()`, `select()`, `group_by()`, `pull()`, `mutate()`, `slice()`, `rename()`, `count()`, `summarise()`, `*_join()`, `pivot_longer()`, `pivot_wider()`, `slice()`, `slice_*()` (this list is not exhaustive!) to clean or transform data frames. These should be connected using the "pipe" operator `%>%` whenever this is appropriate.
  - Use `read_*()` rather than `read.*()` to import data sets.
  - Avoid the use of obsolete commands, i.e. those which are labelled as "lifecycle retired" in the documentation. For instance, do not use the obsolete commands `gather()` and `spread()` (instead use `pivot_longer()` and `pivot_wider()` which were taught in the lecture notes).
  - All visualisations must be created using **ggplot2** commands and syntax.
- Use of R packages other than those taught in the lectures is only allowed if they provide functionality that cannot be achieved using the packages and commands mentioned in the lecture notes **and only subject to approval by the lecturer**.

~~4. Figure formatting [15 points]~~

- You must use meaningful plot types (e.g. do not plot time-series data as a scatterplot).
- If you use colours in your figures, these must not be redundant and the colour scheme must be appropriate.
- All axes and legends must be appropriately labeled using words that are understandable to someone who has not seen your code. In particular, avoid all but common abbreviations in figures unless

absolutely necessary.
- Axis labels must not be overlapping.
- The figures should have a consistent look (e.g. when using colour, the same variables should be represented by the same colours throughout the report and, e.g., you should avoid having `theme_dark()` in one figure and `theme_grey()` in another).

~~5. Data [5 points]~~

- After cleaning, the columns of your data frame(s) should have appropriate data type, i.e. numbers must be stored in numeric (i.e. `integer` or `double`) columns and dates in `date` columns.
- After cleaning, all variables in your data frame(s) should have appropriate and correctly formatted names (remember that the `clean_names()` command from the **janitor** package mentioned in Chapter 6 is very helpful in this regard).
- Check each data set for obvious data-entry errors (e.g. if a measurement is indicated as having been taken in the year 2023); exclude such points during data cleaning and make a brief note of this in accompanying text in the “Data” section.

~~6. Coding style [10 points]~~

- Code layout and naming conventions for variables and functions must follow the style guidelines from Section 2.1–2.4 and 2.6 of Chapter 2 and Section 1.4.6 of Chapter 4 of the lecture notes.
- You must use “snake_case” for naming objects and files and avoid spaces in file names.
- Lines of code (this includes comments!) must not be longer than 80 characters (i.e. no longer than the thin vertical line in RStudio).
- You must add meaningful comments to your code unless it is self-explanatory in the eyes of a person familiar with base R and any of the packages used in the lecture notes (as discussed in Section 2.5 of Chapter 2 of the lecture notes).

~~Bonus points [5 points]~~

For the Analysis in Section 4, you can obtain up to 5 bonus points for conducting more sophisticated analyses (the total number of points is capped at 100).