Assignment 02

Statistical Computing and Empirical Methods

A word of advice

Think of the SCEM labs like going to the gym: if you pay for gym membership, but instead of working out you use a machine to lift the weights for you, you won't grow any muscle.

*ChatGPT, DeepSeek, Claude and other GenAI tools can provide answers to most of the questions below. Before you try that, please consider the following: answering the specific questions below is not the point of this assignment. Instead, the questions are designed to give you the chance to develop a better understanding of estimation concepts and a certain level of **statistical thinking**. These are essential skills for any data scientist, even if they end up using generative AI - to write an effective prompt and to catch the common (often subtle) errors that AI produces when trying to solve anything non-trivial.*

A very important part of this learning involves not having the answers ready-made for you but instead taking the time to actually search for the answer, trying something, getting it wrong, and trying again.

So, make the best use of this session. The assignments are not marked, so it is much better to try them yourself even if you get incorrect answers (you'll be able to correct yourself later when you receive feedback) than to submit a perfect, but GPT'd solution.

Before starting the assignment it is recommended that you watch the video lectures corresponding to Week 2.

This assignment will be done in Rstudio. Please have it open before you start.

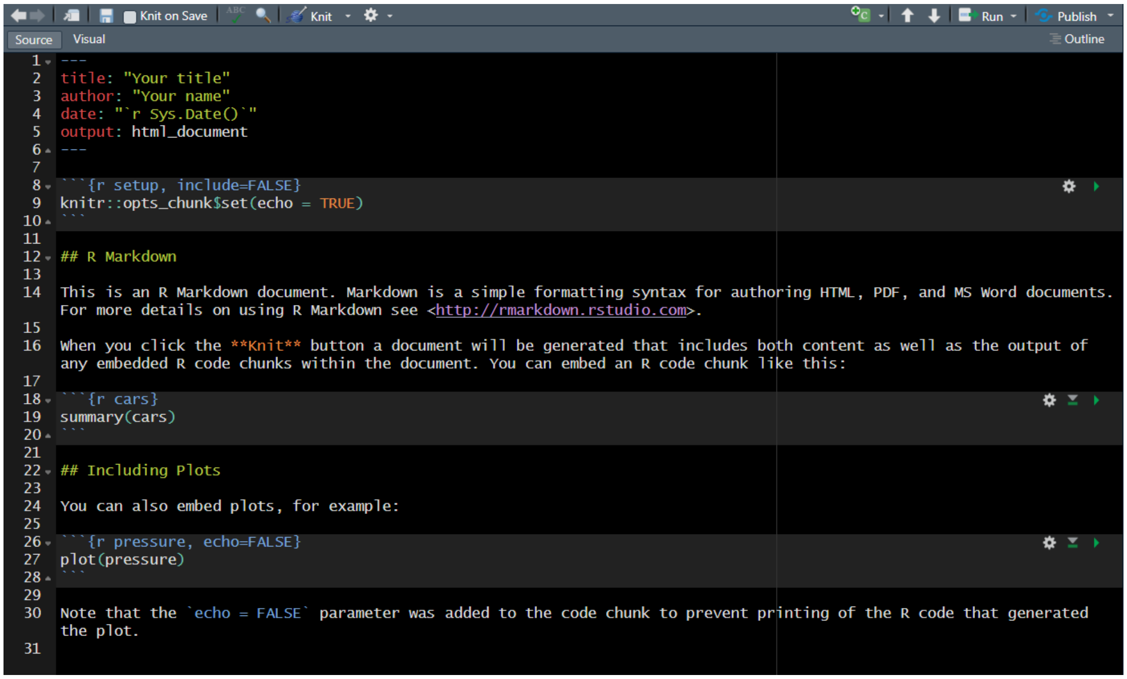
You don't need to submit this assignment.

Introduction

Create an R Markdown for your assignment

It is recommended that you create a single R Markdown document to include your solutions. Use headings, created using markdown header codes such as `### Q1`, `###Q2` etc., to indicate your answer to each item.

It is good practice to use R Markdown to organise your code and results. In RStudio, simply go to `File - New File - R Markdown`, add the title (e.g., "Assignment 02") and the author (you), and you will have a new file pre-populated with some text and code chunks. It should look somewhat like the figure below:



~~~
---
title: "Your title"
author: "Your name"
date: "`r Sys.Date()`"
output: html_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents.
For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of
any embedded R code chunks within the document. You can embed an R code chunk like this:

```{r cars}
summary(cars)
```

## Including Plots

You can also embed plots, for example:

```{r pressure, echo=FALSE}
plot(pressure)
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated
the plot.

~~~

Leave the headers (everything at the top between the triple-dash lines) and the *r setup* code block unchanged. Everything else will be deleted or edited by you (the examples provided in the template can help you structure your own assignment, but make sure your solution only has your own work)

Part I: Data wrangling

Load packages

You will need to load some packages to complete this assignment, namely `dplyr`, `nycflights13` and `gapminder`. If they haven't been installed in your computer, please use `install.packages()` to install them first. Then add this to an R code block at the start of your R markdown:

```
library(dplyr)
library(nycflights13) # For the NYC flight data
library(gapminder)    # For global socio-economic data
```

1. Data Wrangling

This part is mainly about data wrangling. Basic concepts of data wrangling can be found in one of the video lectures of Week 2.

1.1 Select and Filter

The *flights* dataset is loaded automatically when you load the `nycflights13` package. This is a large dataset (with over 300,000 rows and 19 variables) containing the data for all flights that departed from any of the three New York City airports (JFK, LGA, or EWR) in 2013. Use `?flights` and read the documentation of this dataset before you continue.

(Q1) Use a combination of `select()` and `filter()` to create a data frame called `large_delays_JFK` from the `flights` dataset, containing only:

1. Rows representing flights with the following characteristics:
- Departing from **JFK** airport.
- Departure delay is **greater than or equal to 60 minutes**.
2. Columns:
- `year`, `month`, `day`
- `dep_delay` (departure delay in minutes)
- `arr_delay` (arrival delay in minutes)
- `carrier` (airline code)

Use the pipe operator `%>%` to simplify your code. Show the first 5 rows of the resulting data frame.

(Q2) How many **variables** does the `large_delays_JFK` data frame have? How many **observations**?

1.2 Arrange

(Q3) Sort `large_delays_JFK` by **arrival delay** in **descending** order (i.e., largest to smallest) . Print the top 5 rows.

1.3 Join and Rename

Build the following lookup table mapping carrier codes to airline names:

```
airline_lookup <- data.frame(
  carrier = c("AA", "DL", "UA", "B6", "WN"),
  airline_name = c("American Airlines", "Delta", "United Airlines",
                   "JetBlue", "Southwest Airlines"))
```

(Q4) Use `left_join()` to merge `airline_lookup` into `large_delays_JFK`, replacing carrier codes with full airline names. Store this as a new variable `X`

(Q5) Rename the variable `airline_name` in data frame `X` to simply `airline`, then use a combination of `select()` and `print()` to display the first 7 rows of only the `airline_name`, `dep_delay`, and `arr_delay` columns.

(Q6) Does it matter whether you use `left_join()` vs. `inner_join()` in this case? Can you think of a situation in which it would make a difference? (e.g., in a real data analysis project)

1.4 Mutate

Assume that we are interested in a **"delay ratio"** variable, defined as `delay_ratio = arr_delay / dep_delay`.

(Q7) Using `mutate()`, compute this ratio for all flights in `large_delays_JFK`. Use `select()` to keep only `airline_name` and `delay_ratio`, and arrange the dataset by descending value of the delay ratio. Show the top 10 rows.

1.5 Summarise and Group By

(Q8) Use `group_by()` to group the joined dataset by airline name and create a summary table with:

1. Number of flights (`n_flights`)
2. Average departure delay (`avg_dep_delay`)
3. Median arrival delay (`median_arr_delay`)
4. The maximum delay ratio (`max_delay_ratio`)

(Q9) Count the number of **missing values** for each delay-related column by airline.

Part II: A quick recap of probability

- Use plain text + LaTeX for math notation; R code is optional.
- A quick guide to writing math in R markdown using LaTeX can be found here.
- For a slightly more complete LaTeX cheat sheet, check here.

(Q10) Write down the definition of a random experiment, event and sample space. This question aims to help you recall the basic concepts before completing the subsequent tasks.

(Q11) Consider a random experiment of rolling a fair 6-sided die twice. Give an example of what is an *event* in this random experiment. Also, can you write down the *sample space* as a set? What is the *total number of different events* in this experiment? Is the empty set considered as an event?

Suppose that samples of the glass used to manufacture smartphone screens are analyzed for scratch resistance (ScR) and shock resistance (ShR). The results from 100 samples are summarized below:

```
##          ShR:high ShR:low
## ScR:high       70       9
## ScR:low        16       5
```

Let A denote the event that a sample has high shock resistance, and let B denote the event that a sample has high scratch resistance.

(Q12) Determine the number of samples in $A \cap B$, A', and $A \cup B$. (note: A' can be read as *"not A"*, and represents the complement of A)

(Q13) If a sample is selected at random, what is the probability that its *scratch resistance is high* AND its *shock resistance is high*?

(Q14) If a sample is selected at random, what is the probability that its *scratch resistance is high* OR its *shock resistance is high*?

(Q15) Consider the event that a sample has high scratch resistance and the event that a sample has high shock resistance. Are these two events mutually exclusive?

(Q16) Let A and B denote the events as described in **Q12**. Determine the following probabilities: * $P(A)$ * $P(B)$ * $P(A|B)$ * $P(B|A)$

(Q17) Are events A and B independent? Why?

(Q18) Software to detect fraud in consumer phone cards tracks the number of metropolitan areas where calls originate each day. It is found that 1% of the legitimate users originate calls from two or more metropolitan areas in a single day. However, 30% of fraudulent users originate calls from two or more metropolitan areas in a single day. The proportion of fraudulent users in the population is 0.01%. If the same user originates calls from two or more metropolitan areas in a single day, what is the probability that the user is fraudulent?

Some performance indicators commonly used to assess the performance of machine learning models for binary classification can be easily defined in probabilistic terms:

Sensitivity (also known as *true positive rate* or *recall*): probability that a model correctly classifies a *positive* observation, $P(prediction = +|class = +)$

Specificity (also known as *true negative rate*): probability that a model correctly classifies a *negative* observation, $P(prediction = -|class = -)$

Accuracy: probability that a model correctly classifies an observation regardless of class, $P(prediction = -|class = -)P(class = -) + P(prediction = +|class = +)P(class = +)$

Although useful, these indicators are often not very useful in the context of *imbalanced classification*, when one of the classes is much more common than the other.

Imagine that a model is trained for the problem of detecting fraudulent credit card transactions. Most card transactions are legitimate, and only a minority of transactions (e.g., 0.05%) are fraudulent. If we assume that the event "transaction is fraudulent" is our event of interest (the "positive" event) then this means $P(+) = 0.0005$.

(Q19) A *default classifier* (also known as a *zero-rule model*) is a hypothetical model that predicts everything as the majority class - often useless, but good as a baseline. Calculate the Sensitivity, Specificity and Accuracy for a zero-rule classifier in the credit card fraud detection example above.

(Q20) Imagine that you develop a model with 99.9% sensitivity and 99% specificity for the credit card fraud detection example above. If your model predicts a transaction as fraud (i.e., *prediction* = +), calculate the probability that the transaction is indeed fraudulent, $P(class = +|prediction = +)$. (note: this is another common classification metric known as *Precision* or *positive predictive value, PPV*).

OPTIONAL

Imagine the following scenario. You are an Adventurer exploring a dungeon, and you get to a hall with a Wizard and three closed gates (call them G1, G2 and G3). The Wizard tells you that one of the gates leads to a treasure, and the remaining two lead to deadly traps. The rules of the dungeon are such that:

1. The Adventurer need to choose one of the three gates.
2. After they choose, the Wizard will then reveal one of the other two gates which he knows has a trap behind it.
3. After this, the Adventurer can either:
    (a) Walk through the gate they initially selected.
    (b) Switch their choice and walk through the other unopened gate.

For clarity, you can make the following assumptions:

- The treasure is assigned to one of the gates at random with equal probability for each door.
- The assignment of the treasure and the initial choice by the Adventurer are independent.

- Once the initial choice is made, the Wizard always reveals a door which (a) has a trap behind it and (b) is not your initial choice. If there is more than one such door (i.e. if the initial choice corresponded to the door with a treasure behind it) the Wizard chooses one of the other two at random, with equal probability.

To formalise this problem, consider the following events for $i = 1{,}2{,}3$:

- A_i denotes the event "Treasure is behind the i^{th} gate".

- B_i denotes the event "Adventurer initially chooses the i^{th} gate".

- C_i denotes the event "Wizard opens the i^{th} gate to reveal a trap".

(Q21) Consider a situation in which the Adventurer initially chooses the first gate (B_1) and then the Wizard opens the second gate to reveal a trap (C_2). What is $P(A_3|B_1, C_2)$? What does this suggest about a good strategy?
(Tip): you can solve this mathematically or, alternatively, by constructing a table with all possibilities and using it to calculate the proportions of success.

(Q22) Can you build a simulation to estimate the probability of getting the treasure (a) with the initial choice, and (b) by switching choice?