# Problem Set #02 Statistics 506

Due: Sept 26, 10am on Canvas

## Instructions

- **Review the attribution of sources discussions.**
- Submit the output of your Quarto document on Canvas. The output should include a link to your GitHub repository for this assignment.
- Unless otherwise explicitly stated, all problems should be solved in **R**.
- Your output file should include all code required to solve the problem. Code folding may be useful to make the output more readable.
- Use a consistent and readable code style for your code. Lack of a consistent and readable code style will negatively affect your grade.
- Some of these exercises may require you to use commands or techniques that were not covered in class or in the course notes. You can use the web as needed to identify appropriate approaches. Part of the purpose of these exercises is for you to learn to be resourceful and self sufficient. Questions are welcome at all times, but please make an attempt to locate relevant information yourself first.

### New instructions

These are modifications to the instructions from the first problem set. They will move into the general instructions in the next problem set.

- Be sure to properly document any functions you write using roxygen, and add comments as appropriate to make it clear what you are doing.
- If submitting an HTML file, please make sure to make it self-contained.

## Problem 1 - Dice Game

Let's play a dice game. It costs $2 to play. You roll a single 6-sided die.

- On a roll of 2, 4, or 6, you win the amount on the roll (e.g. a roll of 4 wins $4).
- On a roll of 1, 3, or 5, you lose.

We're going to implement this in different ways. Each function takes in as input the number of dice to roll, and each function returns your total winnings or loses. E.g.

```
> play_dice(10)
[1] 4

> play_dice(10)
[1] -6
```

a.

- Version 1: Implement this game using a loop over the die rolls.
- Version 2: Implement this game using built-in R vectorized functions.
- Version 3: Implement this by collapsing the die rolls into a single `table()`. (Hint: Be careful indexing the table - what happens if you make a table of 5 dice rolls? You may need to look to other resources for how to solve this.)
- Version 4: Implement this game by using one of the "`apply`" functions.

b. Demonstrate that all versions work. Do so by running each a few times, once with an input a 3, and once with an input of 3000.

c. Demonstrate that the four versions give the same result. Test with inputs 3 and 3000. (You may need to add a way to control the randomization.)

d. Use the *microbenchmark* package to clearly demonstrate the speed of the implementations. Compare performance with a low input (100) and a large input (10000). Discuss the results

e. Do you think this is a fair game? Defend your decision with evidence based upon a Monte Carlo simulation.

## Problem 2 - Linear Regression

Download the cars data set available at https://corgis-edu.github.io/corgis/csv/cars/. The goal is to examine the relationship between horsepower and highway gas mileage.

a. The names of the variables in this data are way too long. Rename the columns of the data to more reasonable lengths.

b. Restrict the data to cars whose Fuel Type is "Gasoline".

c. Fit a linear regression model predicting MPG on the highway. The predictor of interest is horsepower. Control for:

   - The torque of the engine
   - All three dimensions of the car
   - The year the car was released, as a categorical variable.

   Briefly discuss the estimated relationship between horsepower and highway MPG. Be precise about the interpretation of the estimated coefficient.

d. It seems reasonable that there may be an interaction between horsepower and torque. Refit the model (with `lm`) and generate an interaction plot, showing how the relationship between horsepower and MPG changes as torque changes. Choose reasonable values of horsepower, and show lines for three different reasonable values of torque.

   (Hint: If you choose to use the *interactions* package for this, look at the `at =` argument to help with how year comes into play - choose a reasonable single value for year.

e. Calculate $\hat{\beta}$ from d. manually (without using `lm`) by first creating a proper design matrix, then using matrix algebra to estimate $\beta$. Confirm that you get the same result as `lm` did prior.

## Problem 3 - Stata

Repeat problem 2 parts a. through d. in Stata.

**Important**: Repeating part e. (manually estimating $\hat{\beta}$) in Stata is optional . You can choose to repeat e. for minor extra credit.

Note: Quarto and Stata don't play together well, especially if you're working on Stata remotely. For this problem, and this problem only, you can write a .Do file to answer this question, then include its input and output directly into your submission, by copying the Stata output window and including a non-evaluated chunk, e.g.:

````
```stata
. sysuse auto
(1978 automobile data)

. summarize mpg

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
         mpg |         74     21.2973    5.785503         12         41
```
````

```
```

to produce

```
. sysuse auto
(1978 automobile data)

. summarize mpg

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
         mpg |         74     21.2973    5.785503         12         41
```

Do not just include a single giant Stata chunk; still split it up with discussion as appropriate.

For part e., you should save the graph from Stata (as .png or similar, not Stata's .gph format) and include as an image in your submission. (You can use `graph export` or do it manually.)

If you're feeling extremely adventurous, Stata has a similar functionality to RMarkdown in Stata's dynamic documents. I have a GitHub repository demonstrating how to mix Quarto and dyndoc here. No extra credit will be given for this. If you do explore this, I'd be very curious to hear your experiences about it! Please reach out.