Lecture 2: R Markdown

Basic manipulations:

1. Create a RMarkdown HTML document in `RStudio` and "'knit'" it.
2. Create a new header of type 2 (i.e., `## Header Text`).
3. Perform a linear regression with "Sepal Length" as the response and "Sepal Width" as the explanatory variable from the `iris` dataset. Save the result.
4. Highlight the code using `monochrome` style.
5. Print the summary of the linear regression.
6. Include the QQ plot from the linear regression and change the points to filled dots.
7. Print the head of the `iris` dataset using `kable`.
8. Remove the period from the column labels. (Refer to this link).

More advanced manipulations:

1. Install `kableExtra`. Perform the examples shown in the slides with the `iris` dataset.
2. Using Mathpix, reproduce equation (6.1) from the paper available at https://arxiv.org/abs/math/(
3. Add the reference for the paper and cite it in the RMarkdown file.
4. Recreate your RMarkdown file as a Quarto document.

Lecture 3: GitHub

1. Create a GitHub repository for the RMarkdown file (`.Rmd`) you created in the previous exercise.
2. Edit the `README.md` file and push the `.Rmd` to the repository.
3. Collaborate with another person (Person A and Person B):
   - Ensure the repository is up-to-date. Person B modifies the `.Rmd` and pushes the changes, then Person A pulls the changes.
   - Ensure the repository is up-to-date. Person A modifies the 1st section of the `.Rmd`, while Person B modifies the 2nd section (no conflict). No push or pull in between. Person A commits and pushes, then Person B attempts to push and solve any merge conflicts.
   - Handle conflicts when both Person A and Person B modify the same section of the `.Rmd`.

Lecture 4: Data Structures

Using the following code:

```
set.seed(1)
A <- matrix(rnorm(20), ncol = 2)
B <- matrix(rnorm(20), ncol = 2)
```

1. What are the dimensions of A and B? Compute $A^T B$ and AB^T.
2. Combine A and B row-wise to create C.
3. Let D be a copy of C centered around the mean column-wise. The unbiased estimator of the covariance matrix of C is defined as:

$$\frac{1}{n-1} D^T D,$$

where n is the number of rows in D. Compute this quantity and compare it with `cov(C)`.

Lecture 5: Control Structures

Bootstrap

The bootstrap is a widely used method in statistics, introduced by Efron in 1979. It is simple to implement and versatile. We present the simplest form of bootstrap here:

1. Compute the statistic on the original sample: $\hat{\theta} = g(x_1, \dots, x_n)$.
2. Create a new sample $x_1^*, \dots, x_n^*$ by drawing data from the original sample *at random with replacement*. This new sample is called a *bootstrapped sample*.
3. Compute the statistic on the bootstrapped sample: $\hat{\theta}^* = g(x_1^*, \dots, x_n^*)$.
4. Repeat steps 2 and 3 a total of B times.
5. Compute the unbiased estimator of the variance:

$$\frac{1}{B-1} \sum_{b=1}^{B} (\hat{\theta}_b^* - \hat{\theta})^2.$$

Now, perform the following:

1. Load the `ToothGrowth` dataset and create two vectors of tooth lengths corresponding to the `OJ` and `VC` factors, respectively. Compute the mean of each vector.
2. Create a bootstrap distribution for each vector using $B = 10,000$ iterations and a `for` loop. Use the `sample` function to draw with replacement.
3. Using `ggplot2`, plot two histograms of the bootstrap distributions on the same plot.

Lecture 6: Functions

1. What does the following code return?

```
x <- 2
f1 <- function(x) {
  function() {
    x + 3
  }
}
f1(1)()
```

2. How would you usually write these expressions?

```
`+`(1, `*`(2, 3))
`*`(3, `+`(2, 1))
```

3. How could you improve the readability of this function call?

```
mean(, TRUE, x = c(seq(10), rep(NA, 3)))
```

4. Does the following code throw an error? If so, why?

```
f2 <- function(a, b) {
  a * 3
}
f2(3, stop("This is an error!"))
f2(stop("This is an error!"), 3)
```

5. Propose an infix function in R.