ECE 411 Homework 2 (Fall 2022)
Instructor: Dr. Chau-Wai Wong
Material Covered: Python and R Basics

Problem 1 (20 points) [Python Basics] Python is the most popular programming language used by the machine learning community. In this problem, you will go through a Python tutorial to quickly learn its syntax in order to work on neural network problems that you will encounter in the near future. Note that this process is very different from (and much easier than) learning a programming language for the first time: You learn it by focusing on the special syntax that the "baseline" programming language you are good at does not have. Limit the time spend on this problem to one hour.

In this specific problem, we will use Google Colab to execute Python code. (You are feel to use other integrated development environments (IDEs) such as `PyCharm` in other problems.) An introduction on how to get started with the Colab environment can be found *here*. After opening the above link in your browser, click "Copy to Drive" to play the code in your own Google Drive. Below is *a quick tutorial* on using Python that includes examples on using common Python libraries such as `numpy` and `matplotlib`. (If you feel like to learn from another Python tutorial, feel free to follow that tutorial.) While going over the tutorial, you should note the following unique features in Python:

- Syntax wise: no "`end`" or "`}`"; use indent.
- List/set comprehension
- Built-in data structures such as list and dictionary
- The absence of `++`, `--`, `&&`, `||` operators
- `range()` function
- Exponentiation operator

What to submit: A self-designed Python coding cheat sheet of no more than one page that is tailored to yourself.

Problem 2 (20 points) [ML Environment and First Example] You will need to work on the task using one of the two languages: Python or R.

(a) Coding Environment Setup for Python. Option (*i*) Use a cloud interpreter of Python on `Google Colab`, which allows you to execute Python scripts through a web browser. Option (*ii*) Use locally installed Python interpreter. Download and install `Python` and an IDE such as `PyCharm`.

Coding Environment Setup for R. Install `R`, a statistical programming language, and `RStudio`, an integrated development environment (IDE) for `R`. I suggest you use `RStudio` since it allows you to complete the tasks more efficiently.

(b) Complete *ISLR-2.3 Lab: Introduction to R*. Please write a report, include source code, plots, and provide concise explanation for nontrivial commands and results. For example, what does `attach()` do? In what cases do we need to use `as.factor()`? What do various components of a boxplot mean? Additional hints:

- Data files such as `Auto.data` and `Auto.csv` can be downloaded under Data Sets from ISLR (1st edition)'s webpage: https://www.statlearning.com/resources-first-edition
- When data files are loaded, they should be placed in the same folder as displayed in the bottom-right panel of the `RStudio`.
- Try not to reuse a variable name to avoid difficult-to-debug issues. For example, `auto = na.omit(auto)` is bad. Try `auto = na.omit(auto_raw)` instead.
- To finish executing the `identify()` function, you need to click the "Finish" button at the top-right corner of the plot for which the function is called.
- Function `q()` for exiting R may not work in `RStudio`.

For Python users, please follow the text book's instructions while referring to *the "equivalence" Python code*. Note that it is not possible to create strict one-to-one correspondences between these two programming languages. The"equivalence" Python code is our best effort to replicate the key tasks in R.

Problem 3 (20 points, bonus) [College Dataset] Complete *ISLR-2.4.8* and write a report.

For Python users, please follow the text book's instructions while referring to *the "equivalence" Python code*, where you may find the sample code and the comments useful.

Problem 4 (20 points) [Deriving Least-Squares Estimator]

a) Write model $y_i = \beta_0 + \beta_1 x_i + e_i,\ i = 1, \ldots, n$ into the matrix–vector form. Clearly indicate what are $\underset{\sim}{y}, \mathbf{X}, \underset{\sim}{\beta}$, and $\underset{\sim}{e}$. Set up the cost function $J(\underset{\sim}{\beta})$ to be minimized. Take gradient with respect to $\underset{\sim}{\beta}$ and set it to zero. Express $\hat{\underset{\sim}{\beta}}$ in terms of $\mathbf{X}$ and $\underset{\sim}{y}$. What are the closed-form solution for $\hat{\beta}_0$ and $\hat{\beta}_1$. Is $\hat{\beta}_1$ consistent with that in b)?

b) We have derived in class the least-squares estimator for p predictors. Now, show through partial differentiation that the least-squares estimator for β_1 of model $y_i = \beta_0 + \beta_1 x_i + e_i$ is

$$\hat{\beta}_1 = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sum x_i^2 - n\bar{x}^2}. \tag{1}$$

Prove that the above expression is equivalent to

$$\hat{\beta}_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}. \tag{2}$$

Problem 5 (20 points) [Data-Driven Formula Search] It is known that the sum of the squares of n from $n = 0$ to $N - 1$ has a closed-form expression as follows:

$$\sum_{n=0}^{N-1} n^2 = a_0 + a_1 N + a_2 N^2 + a_3 N^3. \tag{3}$$

Given that a third-order polynomial is uniquely determined in terms of the values of the polynomial at four distinct points, derive a closed-form expression for this sum by setting up a set of linear equations and solving these equations for a_0, a_1, a_2, and a_3. Verify your solution against the formula $\sum_{k=1}^{n} k^2 = n(n+1)(2n+1)/6$.