MIE 223 Data Science
Lab and Assignment 01:
More on Data Manipulation Tools for Data Science with Python

In this lab and assignment, you will learn and gain hands-on experience on the use of Numpy for numerical computation and Pandas for data manipulation.

- Programming language: Python (Google Colab Environment)
- Due Date: Posted in Syllabus

Marking scheme and requirements: You are required to provide appropriate answers to the questions in the Jupyter notebook named `Assignment_01.ipynb` and commit all changes to your assignment repository.

This assignment has *4 points* in total and the point allocation is shown below:

- Solution to Questions (2 points):
  - Q1: 0.5 points
  - Q2(a): 0.5 points
  - Q2(b): 0.5 points
  - Q2(c): 0.5 points
- Quiz (2 points)

What/how to submit your work:

- All your code should be included in the provided notebook named `Assignment_01.ipynb`.
- Commit and push your work to your github repository in order to submit it. Your last commit and push before the assignment deadline will be considered to be your submission. You can check your repository online to make sure that all required files have actually been committed and pushed to your repository.
- A link to create a personal repository for this assignment is posted on QUERCUS.

Notes that you should pay attention to

1. All your code should be written in the provided notebook named `Assignment_01.ipynb`.
2. All functions must *return* the specified return type.
3. Commit and push your work to your GitHub repository in order to submit it. Your last commit and push before the assignment deadline will be considered to be your submission. You can check your repository online to make sure that all required code has actually been committed and pushed to your repository.
4. Your code should show all required outputs asked in the assignment questions. If any of the assignment question asks you to explain your outputs, please use markdown cells to clearly explain them.
5. As a backup plan, also upload a PDF copy of your assignment notebook to GitHub. This is useful especially when GitHub cannot render some images or outputs.
6. Please note the plagiarism policy in the syllabus.

1 Main Assignment

Please answer the questions below in the provided notebook named `Assignment_01.ipynb` using Google Colab.

Q1. Simple Linear Regression Using Numpy

(a) Given a data $D = \{x, y\}$ where x and y are the two variables (features) in the data, we can use x to predict y by fitting a regression line that minimizes the error between our predicted value $\hat{y}$ and the actual value y. The regression line is defined by:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 \cdot x_i \;\; \forall\, i \in \{1, 2, ..., n\}$$

Where

n is the number of observations (rows) in the data D

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$\bar{x}$, $\bar{y}$ is the mean of x and y respectively.

Complete the **fit_regression_line** function (in the **Assignment_01.ipynb** notebook) that takes two 2-D numpy arrays (x and y, each of shape n by 1) and returns a 1-D numpy array of length 2 such that the first element is $\hat{\beta}_0$, and the second is $\hat{\beta}_1$. You are to only use arithmetic operators (e.g +, -, *, /) and Numpy methods for your solution. Answers should **not** use comprehensions or loops. Use the provided codes (in **Assignment_01.ipynb**) to calculate and report the Root Mean Squared Error (RMSE), R-squared score (R^2), and to generate a plot of the regression line.

Q2. Pandas

(a) Using Pandas, convert each **numeric** column in the **iris_df** dataset provided in the **Assignment_01.ipynb** notebook to its z-score values (z). The z-score values should be calculated as follows:

$$z_{rc} = \frac{x_{rc} - \mu_c}{\sigma_c} \, \forall \, r \in \{1, 2, ..., R\}, c \in \{1, 2, ..., C\}$$

where:

R is the number of observations (rows) in the dataset.
C is the number of **numeric** features (columns) in the dataset.
z_{rc} is the z-score value of row r for **numeric** column c
μ_c is the sample mean value for **numeric** column c
σ_c is the sample standard deviation value for **numeric** column c

Note that **Lab_00.ipynb** notebook provides the Pandas method for calculating standard deviation. Please do not hard code the column names, but use for-loop to explicitly loop over the **numeric** column names when calculating the z-scores per column. Output the first five rows (using **.head**).

(b) For each row in the Pandas DataFrame returned in **Q2(a)**, use Pandas to find the average z-score value. Do this by creating a new column named **mean_zscore** that represents the average of the z-score values of all the numeric columns in each row. Output the first six rows after sorting the resulting dataframe by **mean_zscore** in ascending order.

(c) Using Pandas, find the maximum absolute value of the average z-score values (calculated in **Q2(b)**) for each **species** in the dataset. **Output** the resulting Pandas Series sorted in descending order of its values (the maximum absolute value of the average z-score values). The solution should be written in a single line of code. (*Hint*: you can use **apply** method, **lambda** function, and **sort_values** method to achieve this).