Assignment 1
MTL766: Multivariate Statistical Methods

Due Date: 11:59pm Friday, 25th August, 2023

Instructions

Please adhere to the following guidelines while completing this assignment:

1. Your submission should consist of a single Jupyter notebook containing all code implementations and corresponding explanations in Markdown format. A sample submission is provided in the assignment kit.
2. For each question, provide a concise description that outlines your approach.
3. Ensure that your code is thoroughly documented with comments to explain key steps and logic.
4. Evaluation will consider both the quality of your submitted work and your ability to articulate your code's functionality.
5. All code must be independently authored, without any reference to external solutions.
6. While working on this assignment, you're permitted to utilize external libraries like `scipy`, `numpy` and `pandas`.
7. Additionally, feel free to harness the capabilities of Python's built-in system packages and standard modules. These modules include, but are not limited to, `os`, `random`, and `math`.

Your adherence to these instructions will contribute to a comprehensive evaluation of your assignment.

Feel free to address any further questions or concerns you may have on the assignment by posting them in the Microsoft Teams assignment channel.

Submission will be on Gradescope.

Questions [15 marks]

These questions use the `Student Performance` dataset, which simulates student performance data. The dataset consists of 100 datapoints representing different students. The variables in the dataset are:

- **StudyHours:** The number of hours each student studied.
- **PreviousScore:** The student's score in a previous assessment.
- **PracticeTests:** The number of practice tests the student took.
- **Age:** The age of the student.
- **FinalScore:** The final exam score achieved by the student, rounded to two decimal places and constrained within the range of 0 to 100.

This dataset will be used for regression analysis to explore the factors influencing a student's final exam score.

Q1 **Bivariate Regression Implementation [1 mark]**

Implement a Python function that performs bivariate regression on a given dataset. The function should take two lists as inputs, one for the independent variable (X) and another for the dependent variable (Y). The function should return the slope (β_1) and the y-intercept (β_0) of the regression line.

Q2 **Scatter Plot and Regression Line [1 mark]**

Write a Python function that reads a data frame containing two columns of data (X and Y) and visualizes the data points as a scatter plot. Additionally, the function should calculate and plot the regression line over the scatter plot. Label the plot appropriately with a title, axes labels, and a legend for the regression line.

Q3 **Coefficient of Determination (r-squared) [1 mark]**

Create a Python function that calculates the coefficient of determination (r-squared) for a given dataset and its corresponding regression line (β_0 and β_1). The function should take three lists as inputs: X, Y, and the predicted Y values from the regression line. The function should return the r-squared value.

Q4 **r-squared Analysis for Independent Variables [3 marks]**

For each independent variable in the `Student Performance` dataset, calculate the coefficient of determination (r-squared) by performing bivariate regression with the **FinalScore** column as the dependent variable. Identify the independent variable with the highest r-squared value.

Q5 **Residuals and Residual Plot for the Best Variable [1 mark]**

Implement a Python function that reads a data frame containing two columns of data (X and Y), performs bivariate regression for the identified best independent variable, and then calculates the residuals for each data point. The script should plot the residuals against the predicted Y values in a residual plot. Add appropriate labels and a title to the plot.

Q6 **Polynomial Regression for the Best Variable [2 marks]**

Implement a Python function that performs polynomial regression for the identified best independent variable. The function should take the independent variable (X) and the dependent variable (Y) as inputs, along with the degree of the polynomial. Return the coefficients of the polynomial regression model.

Q7 **Prediction and Confidence Intervals for the Best Variable [3 marks]**

Create a Python function that reads a data frame containing the identified best independent variable and the **FinalScore** column. Perform bivariate regression and then use the regression line to make predictions for new X values. Calculate the confidence intervals for the coefficients.

Q8 **Multiple Linear Regression [3 marks]**

Extend the implementation of bivariate regression to handle multiple independent variables. Create a Python function that performs multiple linear regression on the `Student Performance` dataset. Use **StudyHours**, **PreviousScore**, **PracticeTests**, and **Age** columns as independent variables (X) and the **FinalScore** column as the dependent variable (Y). Return the regression coefficients for each independent variable.