Formative Lab Exercise #2:

2.1 Student_Performance_Data_Set (in-Class)

Task Overview

In this lab activity, your task is to build a Linear Regression model using a subset of data points from the Student Performance data. The following are the specific sub-tasks you must accomplish.

1. **Load the Data**: Load the provided Student Performance data in your Jupyter Notebook.
2. **Select a Unique Randomization Seed**: Select a unique integer that will serve as the seed for your randomization.
3. **Sample Train Data**: Randomly sample a subset of the data using the seed you have selected; *limit the sample to 30*. Ensure that your sampled data is representative of the population.
4. **Weight Update Function**: Build a weight update function following the Gradient Descent concept.
5. **Display the Values of Weights**: Print the values of weights at each iteration separated by individual cell.
6. **Plot the Value of Weights:** Display a line chart showing the variation of weight values per iteration. Per each weight, show an individual line chart of values against iteration.
7. **Build a Function for the Final Regression Model:** Create a function using the final regression model after all your iterations. Display the mathematical expression with all the final weights values multiplied by the input variables.
8. **Sample Test Data:** From the remainder of the original dataset, randomly sample another set of 30 observations NOT present in your training sample.
9. **Use the Regression Function for Prediction:** Use your built linear regression function to predict for the Target Variable in your test set.
10. **Calculate for Errors:** Calculate for the overall error between your model's prediction and the actual values in the test set.

The final deliverable will include your code implementation divided into sections according to above subtasks.

Dataset Overview

The Student Performance Dataset is a dataset designed to examine the factors influencing academic student performance. The dataset consists of 10,000 student records, with each record containing information about various predictors and a performance index.

The dataset aims to provide insights into the relationship between the predictor variables and the performance index. Researchers and data analysts can use this dataset to explore the impact of studying hours, previous scores, extracurricular activities, sleep hours, and sample question papers on student performance.

Dataset Attributes

The dataset consists of 5 input features and one target variable:

Input Features:

1. **Hours Studied:** The total number of hours spent studying by each student.
2. **Previous Scores:** The scores obtained by students in previous tests.
3. **Extracurricular Activities:** Whether the student participates in extracurricular activities (1 = Yes or 0 = No).
4. **Sleep Hours:** The average number of hours of sleep the student had per day.
5. **Sample Question Papers Practiced:** The number of sample question papers the student practiced.

Target Variable:

1. **Performance Index:** A measure of the overall performance of each student. The performance index represents the student's academic performance and has been rounded to the nearest integer. The index ranges from 10 to 100, with higher values indicating better performance.

2.2 Breast Cancer Data Set (Assignment)

Task Overview

In this lab activity, your task is to build a Logistic Regression model using a subset of data points from the Breast Cancer data. The following are the specific sub-tasks you must accomplish:

1. **Load the Data**: Load the Breast Cancer data in your Jupyter Notebook from the *scikit-learn* library using:

`load_breast_cancer`(*[, return_X_y, as_frame])	Load and return the breast cancer wisconsin dataset (classification).

2. **Select a Unique Randomization Seed**: Select a unique integer that will serve as the seed for your randomization.
3. **Sample Train Data**: Randomly sample a subset of the data using the seed you have selected; *limit the sample to 30*. Ensure that your sampled data is representative of the population.
4. **Weight Update Function**: Build a weight update function following the Gradient Descent concept.
5. **Display the Values of Weights**: Print the values of weights at each iteration separated by individual cell.
6. **Plot the Value of Weights:** Display a line chart showing the variation of weight values per iteration. Per each weight, show an individual line chart of values against iteration.
7. **Build a Function for the Final Regression Model:** Create a function using the final regression model after all your iterations. Display the mathematical expression with all the final weights values multiplied by the input variables.
8. **Sample Test Data:** From the remainder of the original dataset, randomly sample another set of 30 observations NOT present in your training sample.
9. **Use the Regression Function for Prediction:** Use your built linear regression function to predict for the Target Variable in your test set.
10. **Calculate for Errors:** Calculate for the overall error between your model's prediction and the actual values in the test set.

The final deliverable will include your code implementation divided into sections according to above subtasks.

The description of the features can also be seen on the library.

Important Notes

- **Code Execution**: Ensure your code runs successfully without errors. Submissions with non-functional code may receive lower grades.

Make sure both components are complete and follow the specified guidelines to ensure a smooth evaluation process.

Late Submission Policy

- Activity 2.1 is due within the class hours during lab time.
- Activity 2.2 is due one week after at 11:59PM of.
- Late submissions are accepted but will only be eligible for a maximum of 12 points (60%). All other criteria will be graded as specified above, but the final score will be capped.