**Department of Computing**

**CS 471: Machine Learning**

**BESE 12AB**

**Lab 2: Linear Regression**

**Date: 15 Feb 2024**

**Time: 09:00 AM-01:00 PM & 02:00 PM-5:00 PM**

**Instructor: Dr. Hashir Kiani & Dr. Daud Asif Abdullah**

**Lab 02: Linear Regression**

**Introduction**

Linear regression is a fundamental statistical method used for modeling the relationship between a dependent variable and one or more independent variables. It assumes that the relationship between the variables can be approximated by a linear function. The goal of linear regression is to find the "best-fitting" line (or hyperplane in higher dimensions) that minimizes the sum of the squared differences between the observed values and the values predicted by the linear model. This line is characterized by its slope (weight) and intercept (bias). Linear regression is widely used in various fields, including economics, finance, biology, and engineering, to analyze and predict relationships between variables.

**Objective**

This lab is designed to deepen your understanding of Linear Regression through practical exercises in Google Colab.

**Tools/Software Requirement**

Google Colab

**Description and task**

Perform the following tasks.

1. **Linear regression single prediction:** Create a function that takes two vectors as input (weight and feature vector) and returns the linear regression prediction for that input.
2. **Linear regression vector prediction:** Create a function that takes a matrix and a vector as input (weight vector and feature matrix) and returns the linear regression prediction vector for the whole training set.
3. **Mean Squared Error:** Now create a function that takes a vector of predictions and a vector of actual values as input and returns the Mean Squared Error.
4. **MSE Gradient:** Now create a function that takes a vector of predictions and a vector of actual values as input and returns the Gradient of Mean Squared Error.
5. **Gradient Descent Algorithm:** Create a function that implements the gradient descent algorithm. The function should take a weight vector, gradient of loss and a stopping criterion as input and return the optimized weight vector. Implement both vanilla and stochastic gradient descent cases.
6. **Application of linear regression:**

Apply linear regression on the California housing dataset. You can find details about this dataset at the following link: <https://developers.google.com/machine-learning/crash-course/california-housing-data-description>

The dataset can be loaded by using the following lines of code:

from sklearn import datasets

california\_housing = datasets.fetch\_california\_housing()

X = california\_housing.data  # Feature matrix

y = california\_housing.target  # Target values (median house values)

Note: Code everything from scratch in this task. You can use any functions that you have developed in the previous parts here. Apply both vanilla and stochastic gradient descent and compare the results in both cases. Use different batch sizes for stochastic gradient descent. Give your results in a tabular form and provide your analysis on which approach of gradient descent works better.

**Deliverable**

Students are required to upload the lab task solution in. ipynb format on LMS.