**Department of Computing**

**CS 471: Machine Learning**

**BESE 12AB**

**Lab 03: Logistic Regression**

**Date: 23 Feb 2024**

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

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

**Lab 03: Logistic Regression**

**Introduction**   
Logistic regression is a statistical technique utilized for binary classification, predicting the probability of an input belonging to a specific class by fitting a logistic function to the observed data. It's widely employed in various fields such as healthcare, finance, and marketing for its simplicity, interpretability, and effectiveness in handling categorical outcomes.

**Objective**

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

**Tools/Software Requirement**

Google Colab

**Description and tasks**

Please refer to class notes and perform the following tasks.

1. **Logistic regression single prediction:** Create a function that takes two vectors as input (weight and feature vector) and returns the logistic regression prediction for that input.
2. **Logistic regression vector prediction:** Create a function that takes a feature matrix and a weight vector as input and returns the logistic regression prediction vector for a batch of data.
3. **Logistic Loss:** Now create a function that takes a feature matrix, a weight vector and a label vector as input and returns the Logistic Loss for that batch. You can use the function from previous part here.
4. **Gradient of Logistic Loss:** Now create a function that takes a weight vector, a feature matrix and a label vector as input and returns the Gradient of Logistic Loss for that batch. You can call any functions from the previous parts here.
5. **Gradient Descent Algorithm:** Nowcreate a function that implements the gradient descent algorithm for logistic regression. The function should take a weight vector, a feature matrix, a label vector, learning rate and a stopping criterion as input and return the optimized weight vector. Implement both vanilla and stochastic gradient descent cases.
6. **Application of Logistic Regression:**

Apply logistic regression on the MNIST dataset to do multiclass classification. You can find details about this dataset at the following link: <https://www.kaggle.com/datasets/hojjatk/mnist-dataset>

Use the one vs rest approach discussed in class to do this task. Use 50,000 examples for training and the rest for testing. Give the accuracy, precision, recall and F-1 score for each class in a table.

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

# Load the MNIST dataset

mnist = fetch\_openml('mnist\_784', version=1, cache=True, parser='auto')

# Pandas data frame with feature vectors

X = mnist.data

# Labels

y = mnist.target

# Labels converted to integers

y = y.astype(int)

print("Feature vectors shape:", X.shape)

print("Labels shape:", y.shape)

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.