Report on "Implementing QDA and PCA on the MNIST Dataset"

Gaurav Deswal
MT23118

Introduction

This report mentions the assumptions, implementation details, and results obtained from applying Quadratic Discriminant Analysis (QDA) and Principal Component Analysis (PCA) on the MNIST dataset using Python programming language. The MNIST dataset for handwritten digit recognition contains 60,000 training samples and 10,000 test samples across 10 classes (digits 0-9).

Assumptions

Assumptions for Question 1 Python Code: q1.py

1. **mnist.npz** file is present in same folder as the source code **q1.py**
2. No of training samples and training labels for each sample are equal.
3. Make sure the following packages are installed on your Python Environment. I have also mentioned the package version used for development.
    a. numpy == 1.26.4
    b. matplotlib == 3.8.2

Assumptions for Question 2 Python Code: q2.py

1. **mnist.npz** file is present in same folder as the source code **q2.py**
2. No of training samples and training labels for each sample are equal.
3. **q1.py** must be present at the same level as **q2.py**.
4. **q1.py** must have the following functions defined and working correctly: **load_mnist_dataset, calculate_accuracy, calculate_class_wise_accuracy**
5. Make sure the following packages are installed on your Python Environment. I have also mentioned the package version used for development.
    a. numpy == 1.26.4
    b. matplotlib == 3.8.2

Approach

Question 1: QDA Implementation

The approach for QDA (q1.py) involved several steps:

1. **Data Loading and Visualization:** The MNIST dataset was loaded, and 5 samples from each class were visualized.
2. **Data Preparation:** Images were vectorized from a 28x28 format to a 784-dimensional vector to facilitate mathematical operations.
3. **Model Implementation:** The mean vector and covariance matrix for each class were computed based on the training data. Using these parameters, QDA was applied to classify the test dataset. The QDA score computation used the derived expression from the lecture.

4. **Result Analysis:** Accuracy and class-wise accuracy were reported.

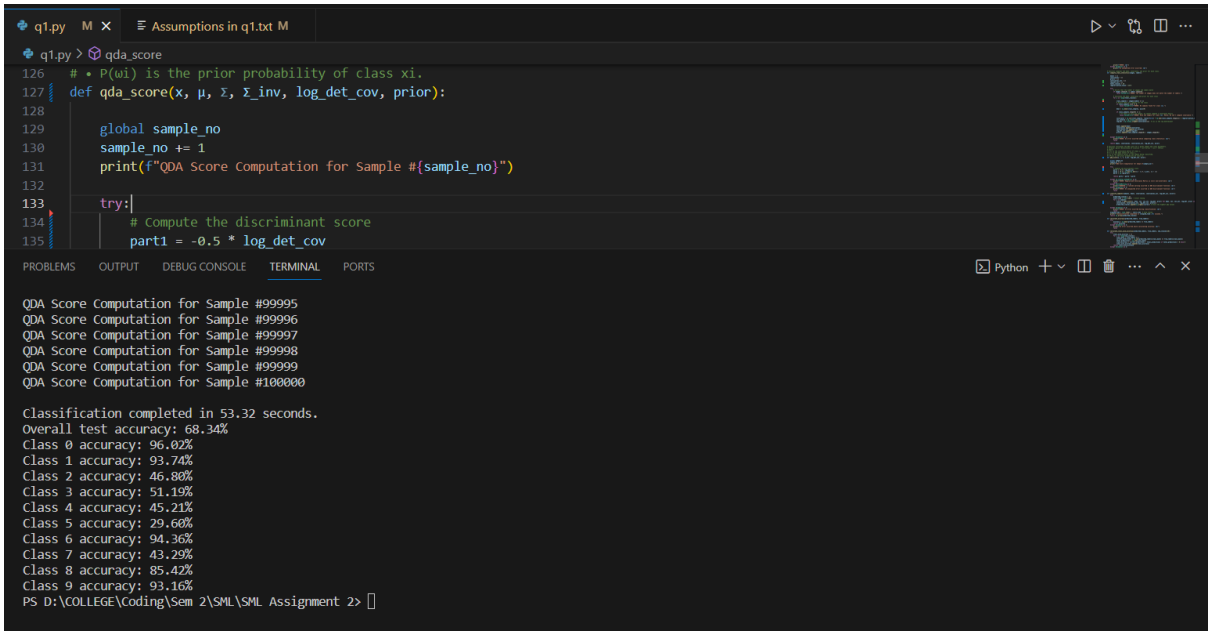
Question 2: PCA and QDA on Reduced Dimensions

For PCA implementation (q2.py), the steps were as follows:

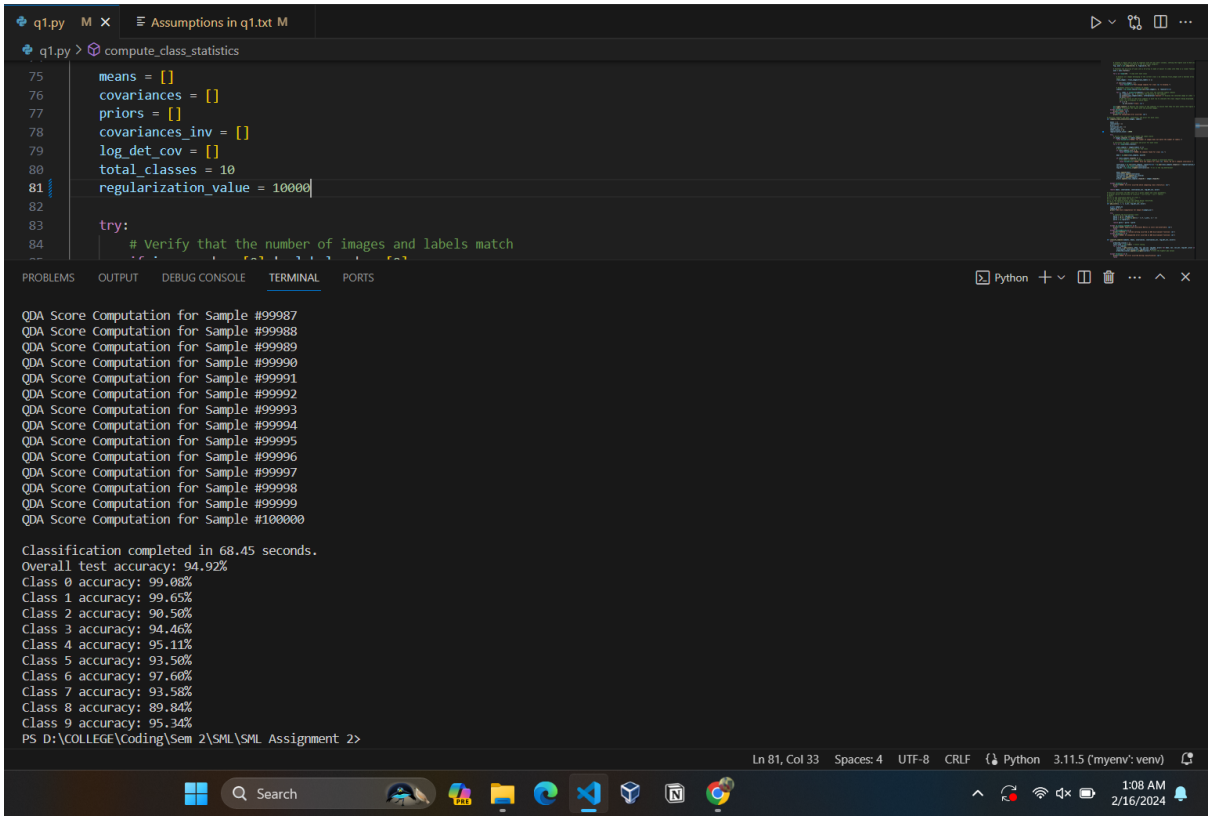
1. **Data Matrix Creation:** A data matrix X of 784x1000 was created by selecting 100 samples from each class, followed by **mean removal** to centralize the data.
2. **PCA Application:** PCA was applied to the centralized X, involving covariance computation, eigenvalue, and eigenvector derivation, and sorting them in descending order to form matrix U. The dataset was then projected onto the PCA space to obtain Y, and X_recon was calculated to check the reconstruction quality through MSE.
3. **Dimensionality Reduction and QDA Application:** The test set was projected onto spaces defined by p=5,10,20 principal components. QDA, as implemented in q1.py, was then applied to these reduced dimensions.
4. **Result Analysis:** Accuracy and class-wise accuracy were reported.

Results

Question 1:

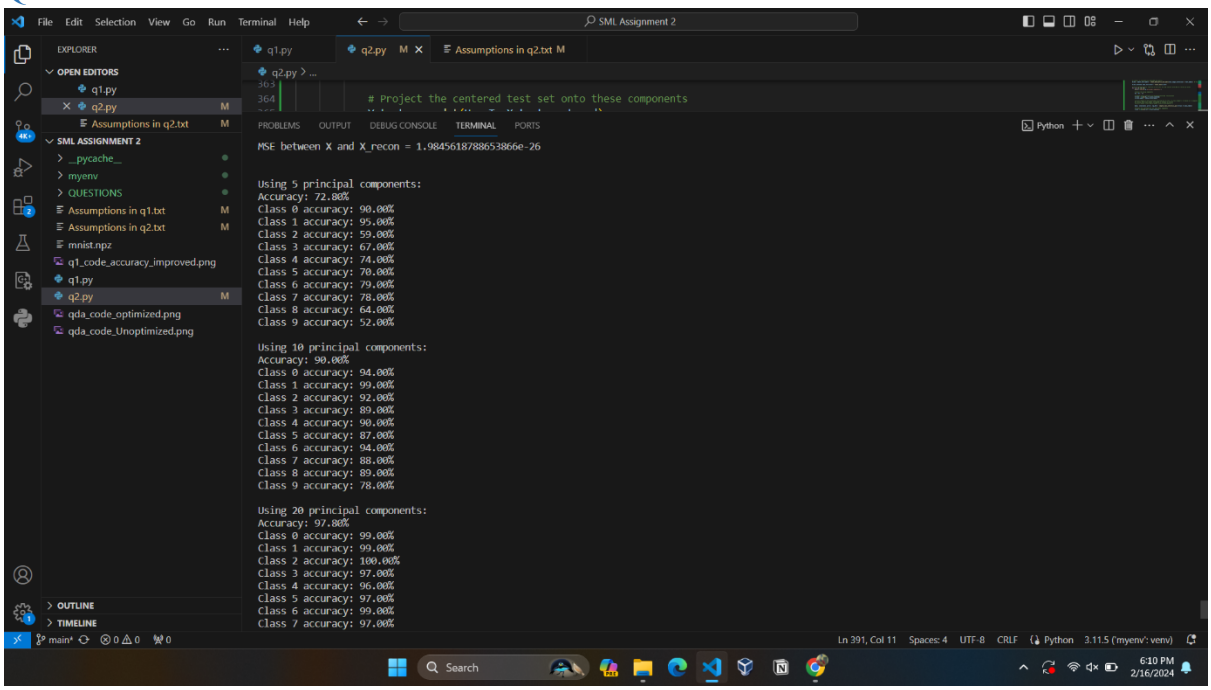


Initial Accuracy = 68.34%



Final Accuracy = 94.92%

Question 2:



For p = 5, Accuracy = 72.80%

For p = 10, Accuracy = 90.00%

For p = 20, Accuracy = 97.80%