BILKENT UNIVERSITY

2023- 2024 FALL SEMESTER

CS464 Homework 2 Report

13.12.2023

Instructor: Ayşegül Dündar Boral

Mennan Gök

22003074

Question 1

Q.1.1

- The proportion of variance explained for the first 10 principal components:

  0.09704665

  0.07095925

  0.0616909

  0.0538942

  0.04868798

  0.04312232

  0.0327193

  0.02883896

  0.0276203

  0.02357001

- Total variance explained by the first 10 principal components: 0.48814985156059265.

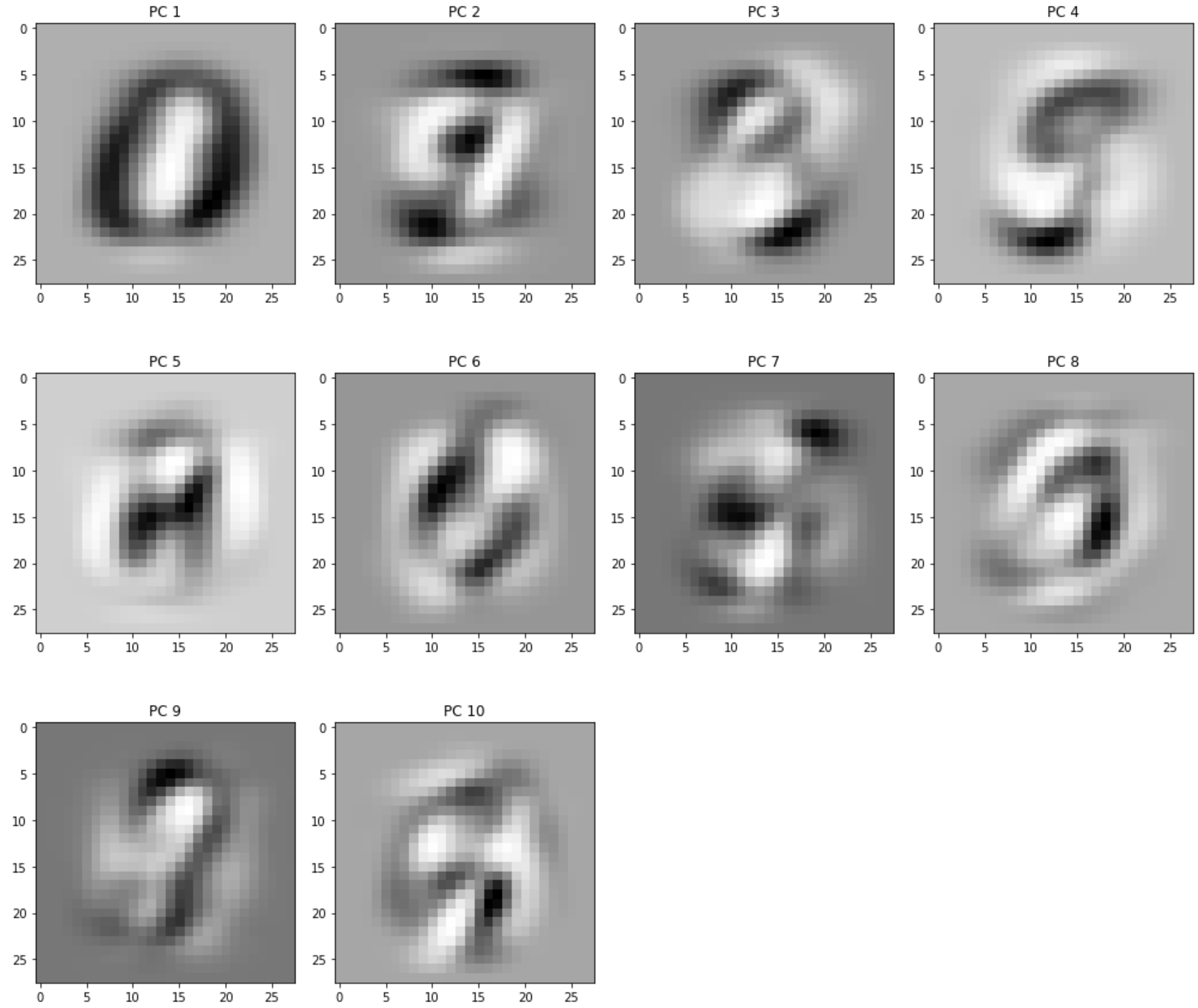
Using only 10 principal components out of 784, we were able to explain almost half of the variance in the data. While this shows how powerful the PCA algorithm is for dimensionality reduction, we lose slightly more than half of the variance in the data which is an important point to consider. With that said, we may consider using some more principal components to reach a balance between the portion of variance explained and the complexity of the new data.

Q.1.2

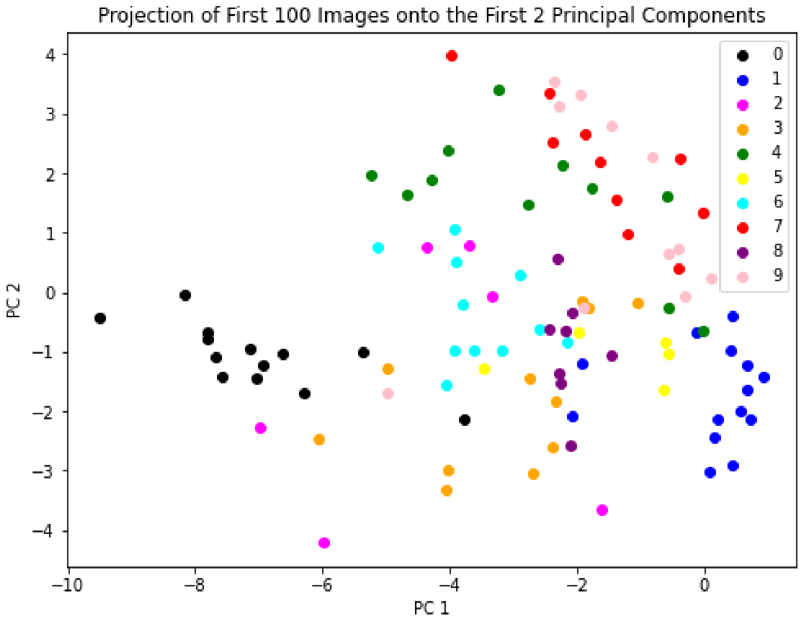
- Required number of principal components to explain the 70% of the data is found as 26.

Q.1.3

First 10 principal components:



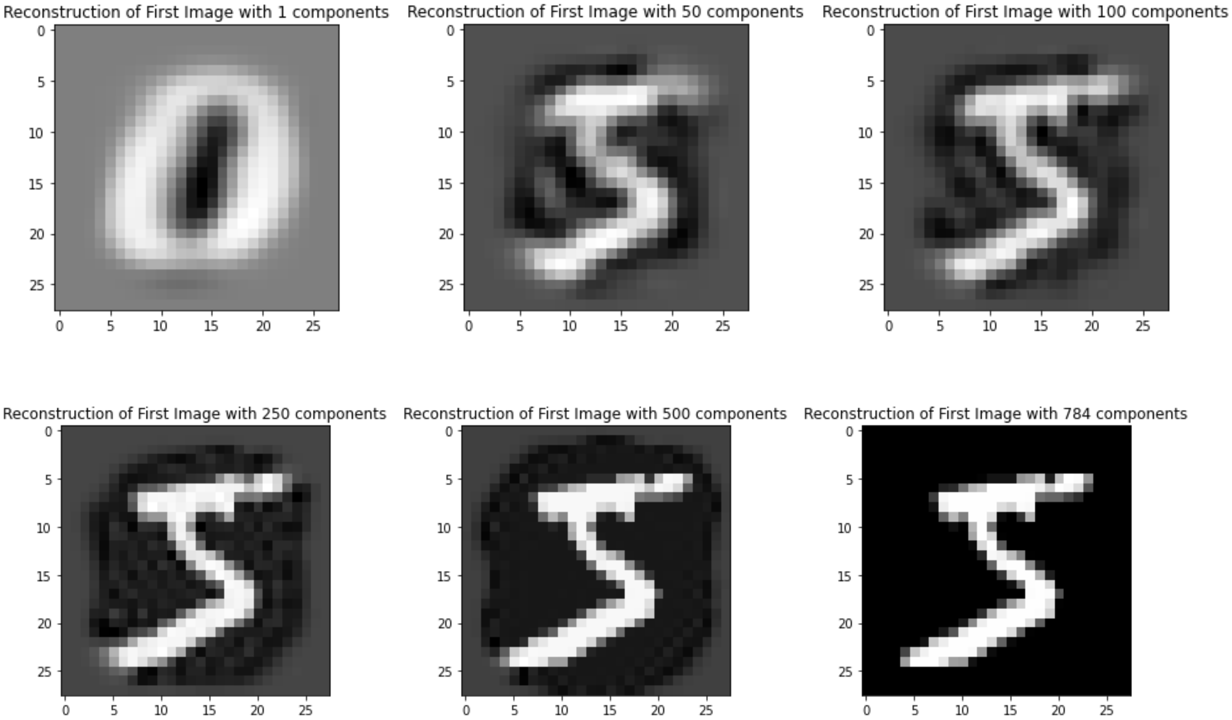
Q.1.4



Investigating this scatter plot more closely, one can see that while the data points of some classes form clusters around the space (such as the data points of the classes 0 and 1), the data points of some other classes overlap (such as the data points of the classes 4, 7, and 9). This suggests that some classes like 0 and 1 are more distinguishable along the first two principal components while other classes like 4,7, and 9 share more characteristic properties along the first two principal components and might require a greater number of components to be distinguished more easily.

Q.1.5

To reduce the dimensionality of the data, we projected the original data onto the space spanned by the selected principal components. In the process of reconstruction, we need to project the data in the space of principal components back to the original space. This is achieved by multiplying the reduced data matrix by the transpose of the matrix of the selected components.



Reconstruction of First Image with 1 components

Reconstruction of First Image with 50 components

Reconstruction of First Image with 100 components

Reconstruction of First Image with 250 components

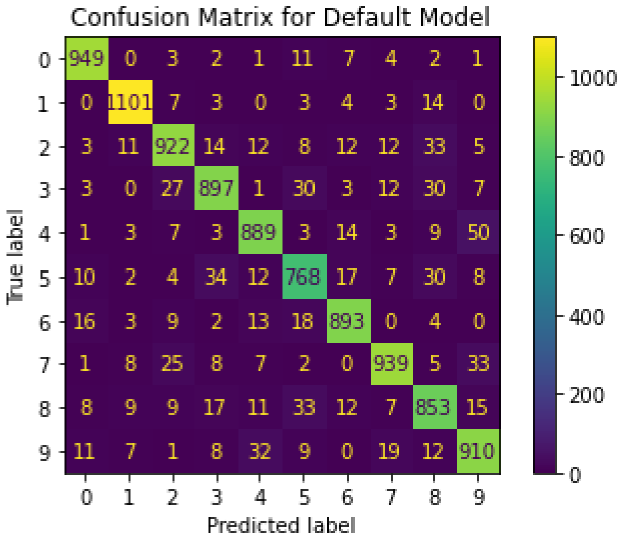
Reconstruction of First Image with 500 components

Reconstruction of First Image with 784 components

Question 2

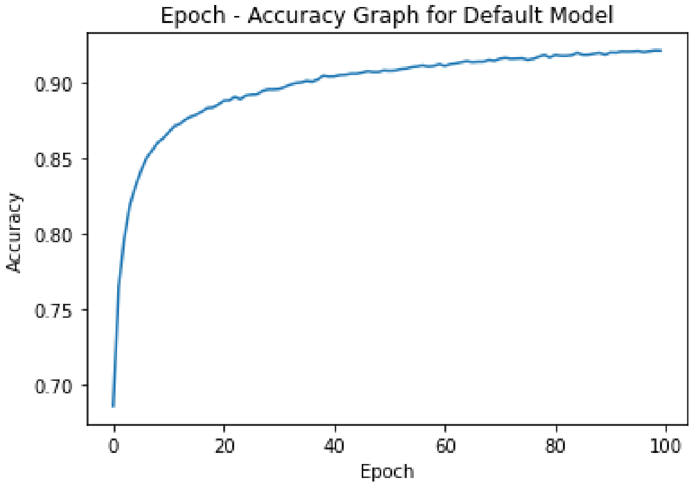
Q.2.1

Training the default Logistic Regression Classifier with the default parameters, we get an accuracy of 0.9121, and the confusion matrix:



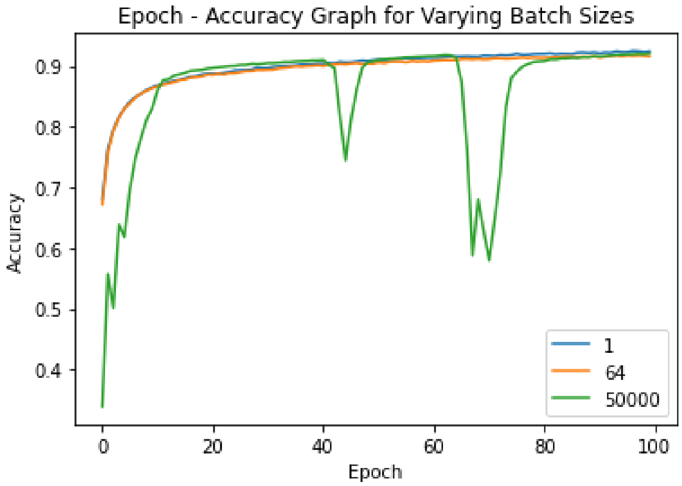
	0	1	2	3	4	5	6	7	8	9
0	949	0	3	2	1	11	7	4	2	1
1	0	1101	7	3	0	3	4	3	14	0
2	3	11	922	14	12	8	12	12	33	5
3	3	0	27	897	1	30	3	12	30	7
4	1	3	7	3	889	3	14	3	9	50
5	10	2	4	34	12	768	17	7	30	8
6	16	3	9	2	13	18	893	0	4	0
7	1	8	25	8	7	2	0	939	5	33
8	8	9	9	17	11	33	12	7	853	15
9	11	7	1	8	32	9	0	19	12	910

Also, we can visualize the Epoch – Accuracy graph as:

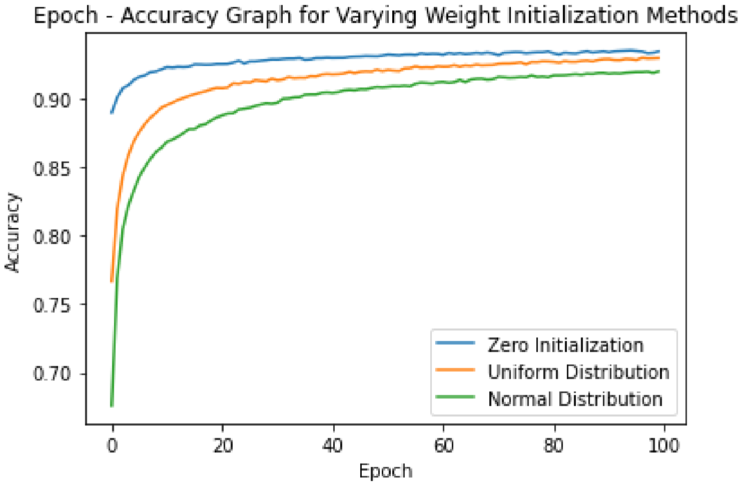


Q.2.2

For this part of the homework, we will experiment with varying hyperparameter values. Firstly, we will vary the batch size parameter while keeping all the other parameters as default: (as in Q.2.1).

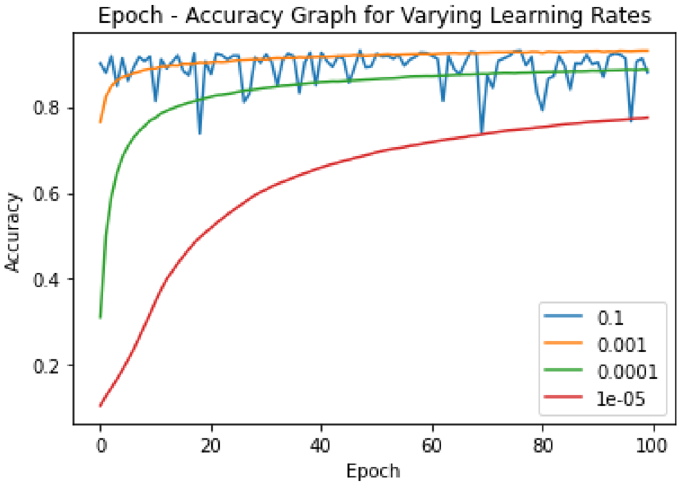


Secondly, we will vary the weight visualization technique while keeping all the other parameters as default:

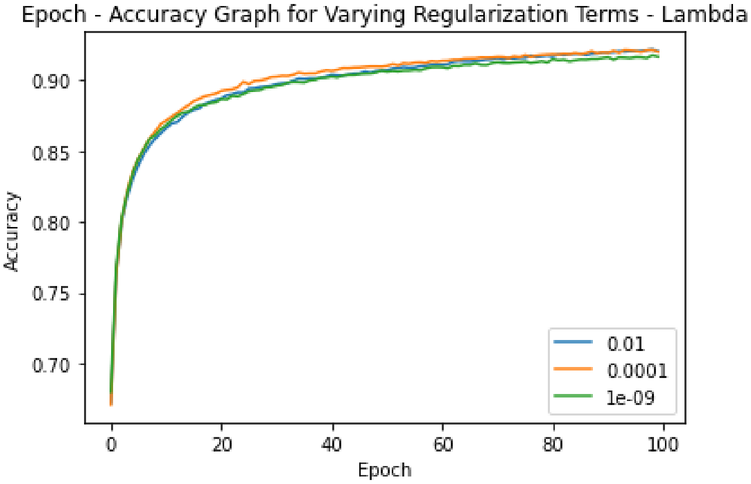


Then, we will vary the learning rate while keeping all the other parameters as default:

Q.2.3



Lastly, we will vary the regularization coefficient while keeping all the other parameters as default:



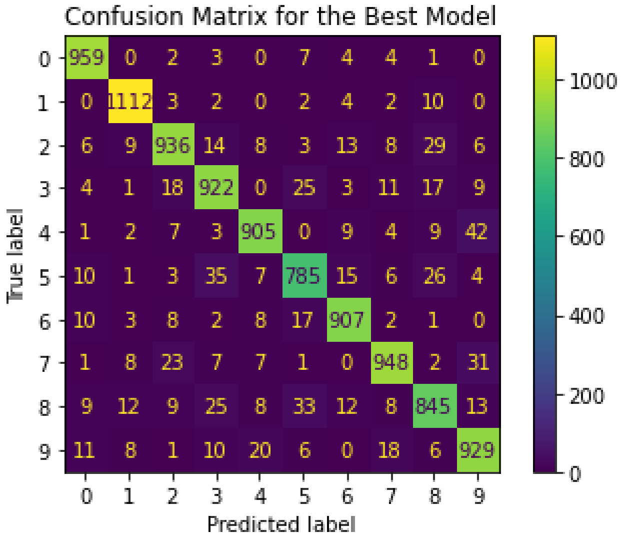
Q.2.3

Based on the above experiments, we create the best model with the following parameters:

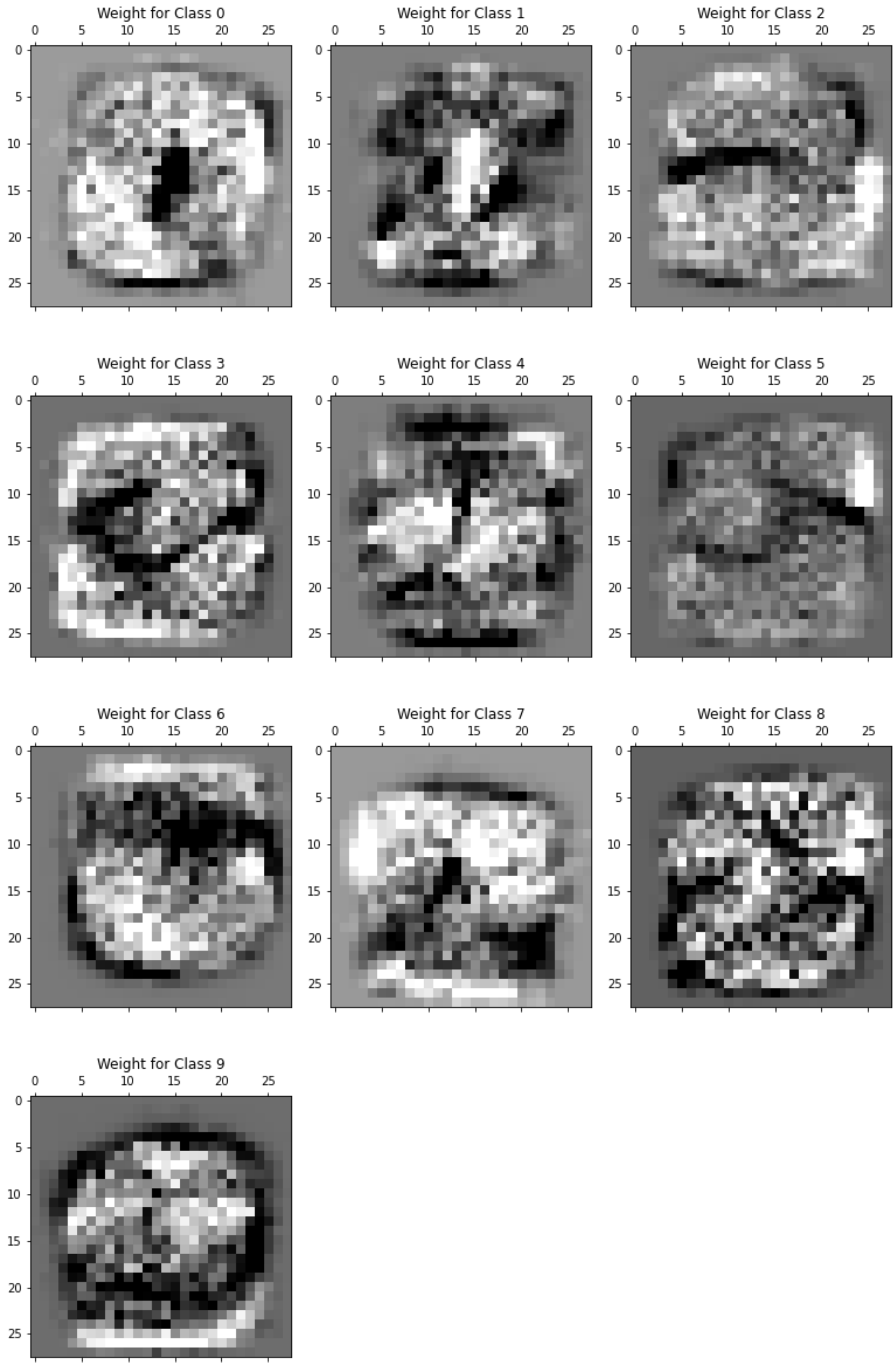
- Batch Size: 64, similar performance to Batch Size = 1, but works way faster.
- Weight Initialization Method: Zero Initialization
- Learning Rate: 0.001
- Regularization Coefficient: 0.0001, almost the same performance as others.

Then, we get the test accuracy 0.9248 for this model.

Now, we plot the Confusion Matrix for the Best Model:



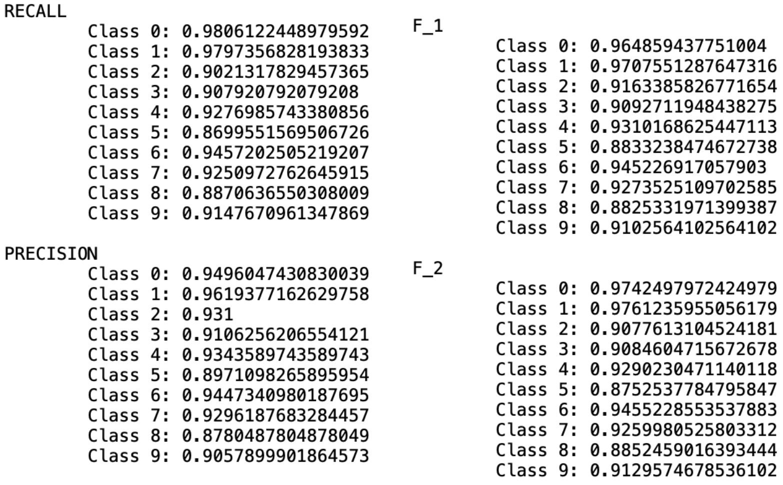
Q.2.4.


Weight for Class 0
Weight for Class 1
Weight for Class 2
Weight for Class 3
Weight for Class 4
Weight for Class 5
Weight for Class 6
Weight for Class 7
Weight for Class 8
Weight for Class 9
0
5
10
15
20
25

Notice that the visualization of the weight parameters of each class resembles the corresponding digit. This happens because the model has updated the weights in a way that keeps the patterns that distinguishes each digit from other digits. The pattern embedded within the weights forms an image that highlights the most distinguishing features of each digit. We can also say that each visualization represents what the model has learned to identify as the key characteristics of the corresponding digit.

Q.2.5

Now we calculate the recall, precision, F_1 score, and F_2 score for each class, using the best model. The metrics shown in the figure below indicate a high level of performance in the classification task. High recall and precision scores suggest that the model has a good balance between sensitivity and positive predictive value. Specifically, the model is performing best in classes 0 and 1. The relatively lower scores for classes 5 and 8 suggest some room for improvement in distinguishing these digits. With that being said, these metrics demonstrate an overall high performance on the classification task.



```
RECALL
        Class 0: 0.9806122448979592
        Class 1: 0.9797356828193833
        Class 2: 0.9021317829457365
        Class 3: 0.907920792079208
        Class 4: 0.9276985743380856
        Class 5: 0.8699551569506726
        Class 6: 0.9457202505219207
        Class 7: 0.9250972762645915
        Class 8: 0.8870636550308009
        Class 9: 0.9147670961347869

PRECISION
        Class 0: 0.9496047430830039
        Class 1: 0.9619377162629758
        Class 2: 0.931
        Class 3: 0.9106256206554121
        Class 4: 0.9343589743589743
        Class 5: 0.8971098265895954
        Class 6: 0.9447340980187695
        Class 7: 0.9296187683284457
        Class 8: 0.8780487804878049
        Class 9: 0.9057899901864573
```

```
F_1
        Class 0: 0.964859437751004
        Class 1: 0.9707551287647316
        Class 2: 0.9163385826771654
        Class 3: 0.9092711948438275
        Class 4: 0.9310168625447113
        Class 5: 0.8833238474672738
        Class 6: 0.945226917057903
        Class 7: 0.9273525109702585
        Class 8: 0.8825331971399387
        Class 9: 0.9102564102564102

F_2
        Class 0: 0.9742497972424979
        Class 1: 0.9761235955056179
        Class 2: 0.9077613104524181
        Class 3: 0.9084604715672678
        Class 4: 0.9290230471140118
        Class 5: 0.8752537784795847
        Class 6: 0.9455228553537883
        Class 7: 0.9259980525803312
        Class 8: 0.8852459016393444
        Class 9: 0.9129574678536102
```