CS 464: Introduction to Machine Learning
Homework 2 Report

Onurcan Ataç
22002194

Q1. PCA Analysis

Q1.1.

Results of the PCA:

PVE for each component separately:
[0.09704664 0.07095924 0.06169089 0.05389419 0.04868797 0.04312231
 0.0327193 0.02883895 0.02762029 0.02357001]

PVE for the first 10 principal components cumulatively:
[0.09704664 0.16800588 0.22969677 0.28359097 0.33227894 0.37540125
 0.40812055 0.4369595 0.4645798 0.4881498]

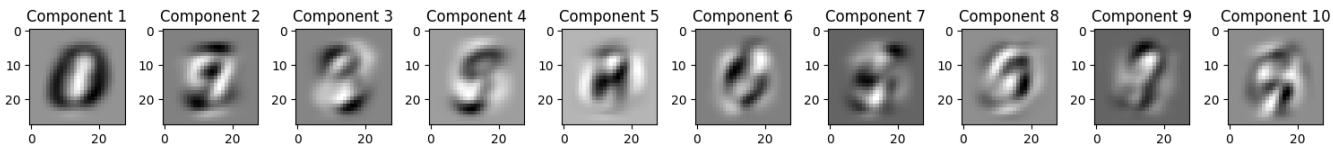
Results indicate that more than 48% variance in the data can be explained only by the first 10 principal components. Nearly 10% of the variance is explained only by the first principal component. The first part shows the PVE for each component separately, which is descending like expected since the components are sorted. The second part shows the cumulative PVE. It increases expectedly.

Q1.2.

The number of components needed to explain at least 70% of the variance is found to be 26 with the function implemented in the code.

Q1.3.

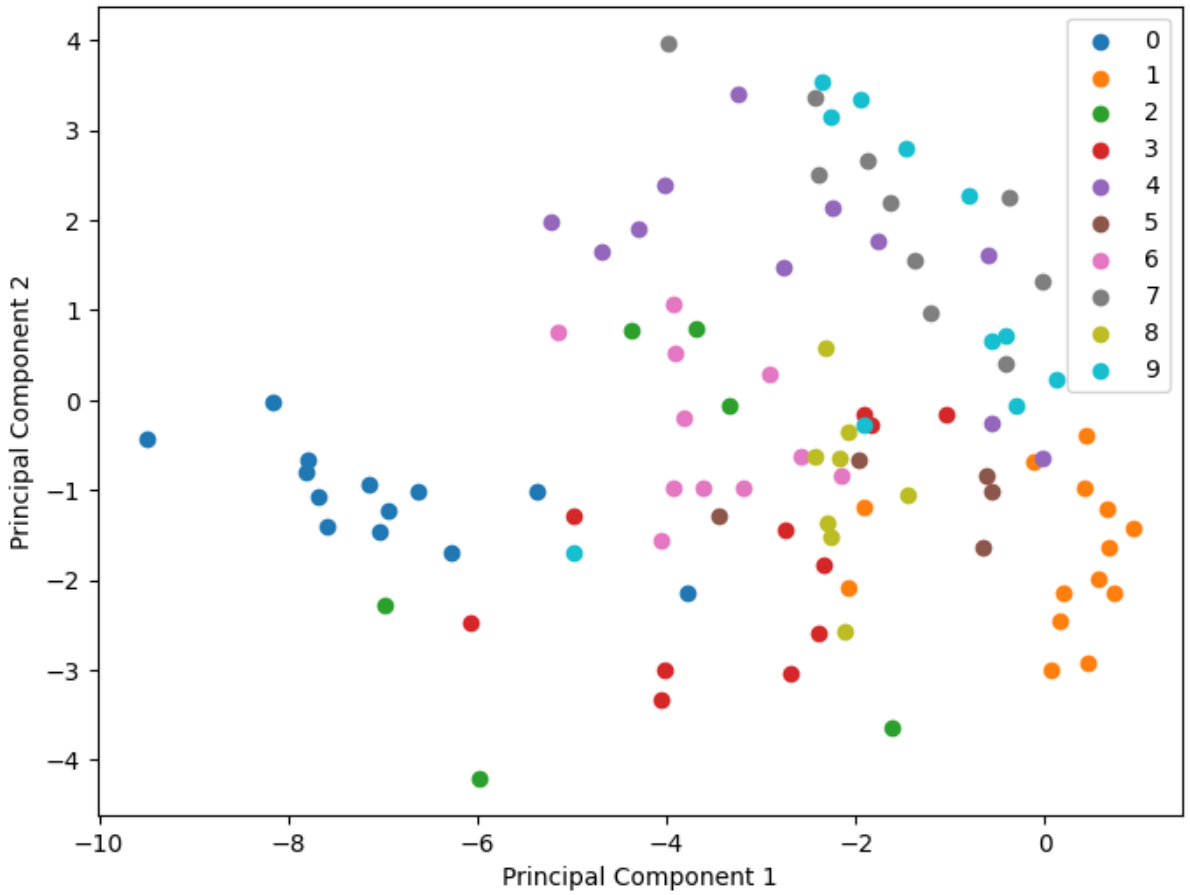
Obtained grayscale principal component images:



These greyscale images show the significant patterns that contribute to the variance in the data. Component 1 accounts for most of the variability. As these patterns are captured by the model, the model provides better results.

Q1.4

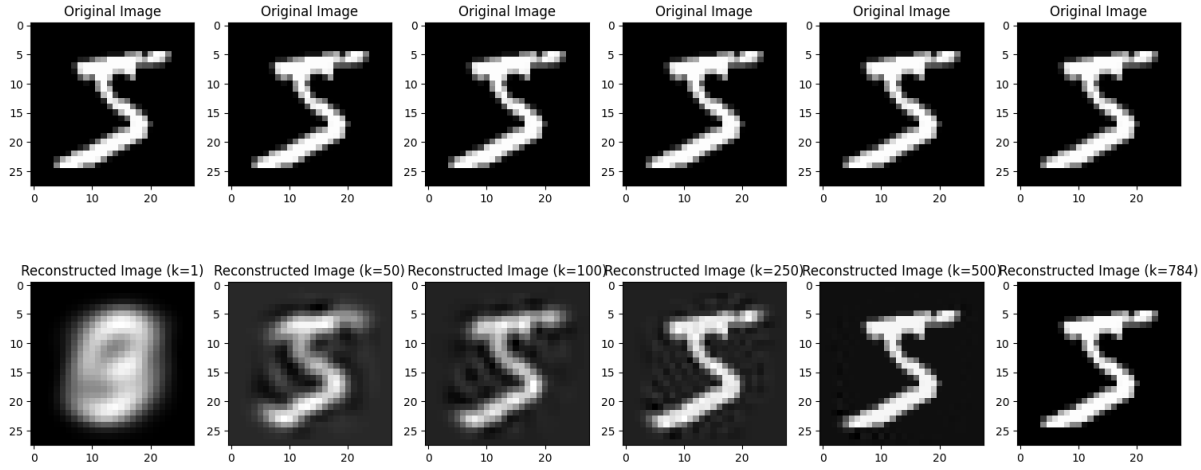
Plot:



Even with the 2 principal components, the plot shows that digits have started to cluster effectively. The plot shows that digits 0 and 1 are the most effectively clustered by the two principal components.

Q1.5

Reconstruction Results:

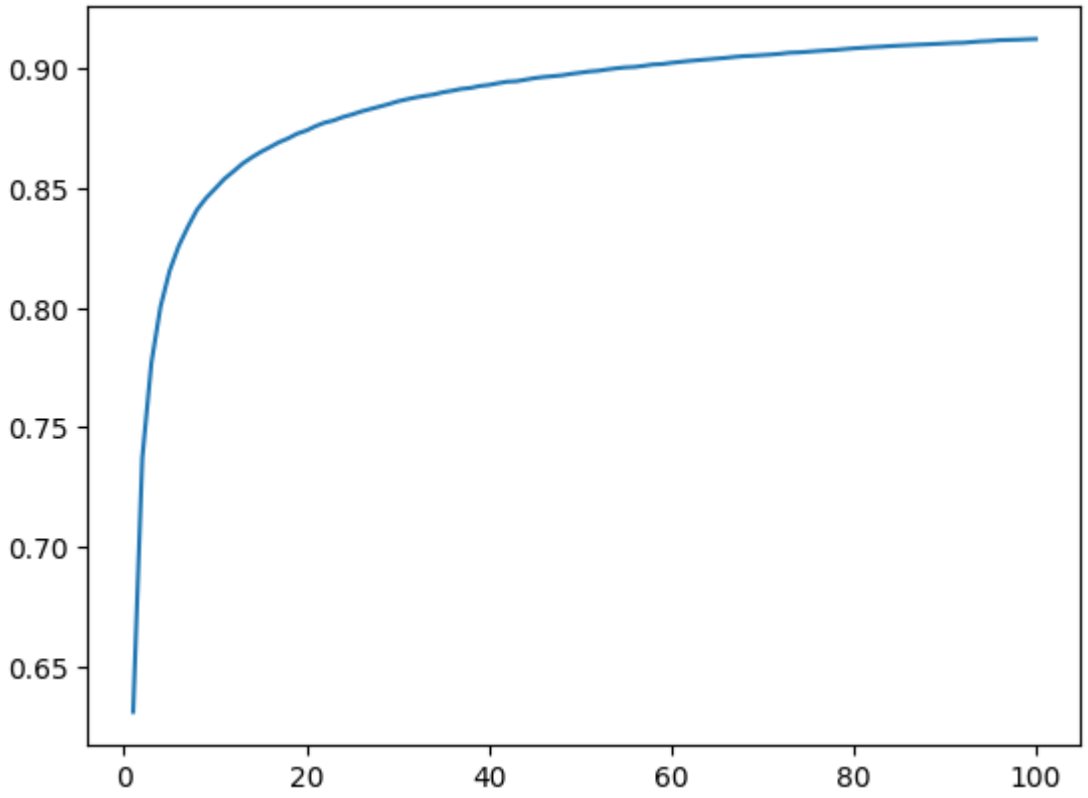


By these results, it is observed how the image gets significantly closer to the original image as the k value increases. When k approaches 50, the reconstructed digit is already recognizable. This can be explained by the result of Q1.2 since 70% of the variability can be explained by the first 26 principal components. When k is equal to 50, it is already past 26. Expectedly, we see the reconstructed image getting more and more similar to the original image as k increases, and when k is equal to 784, they are exactly the same since there are no other principal components.

Q2. Logistic Regression

Q2.1.

Training Accuracy Over Epochs:



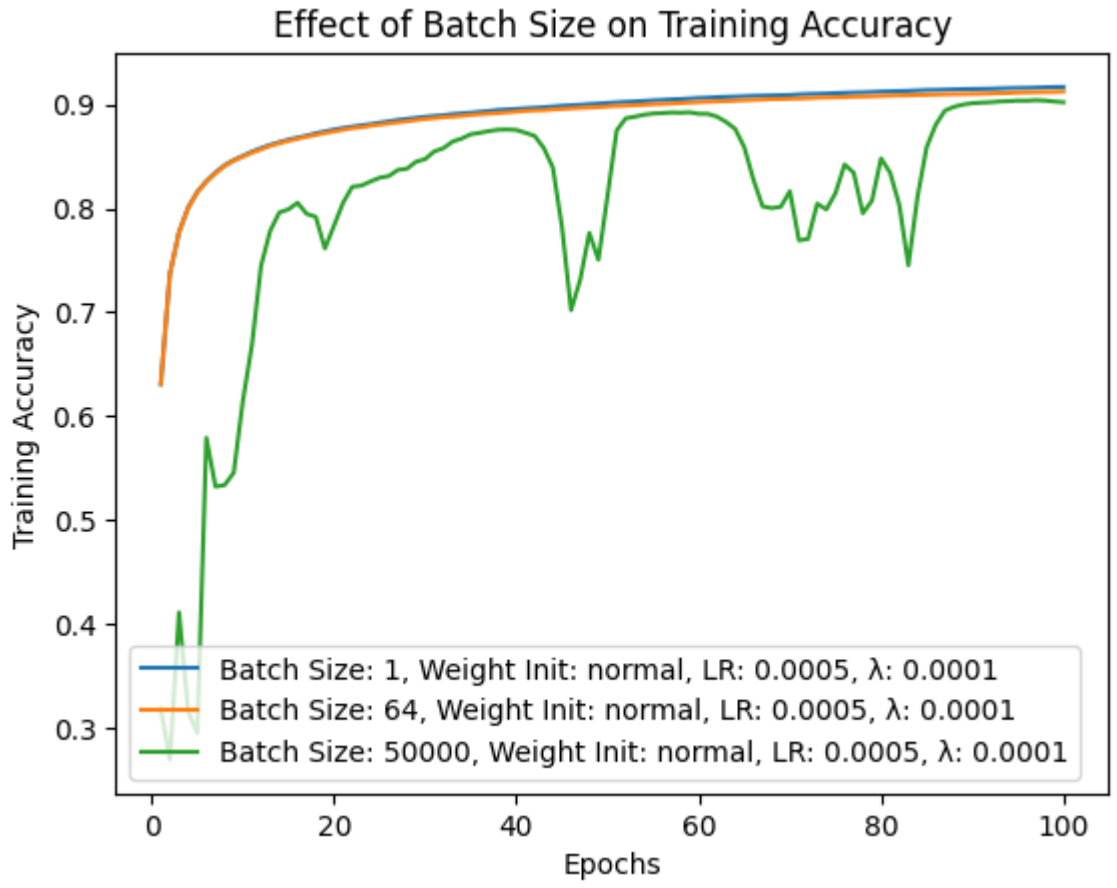
Test Accuracy: 0.9064

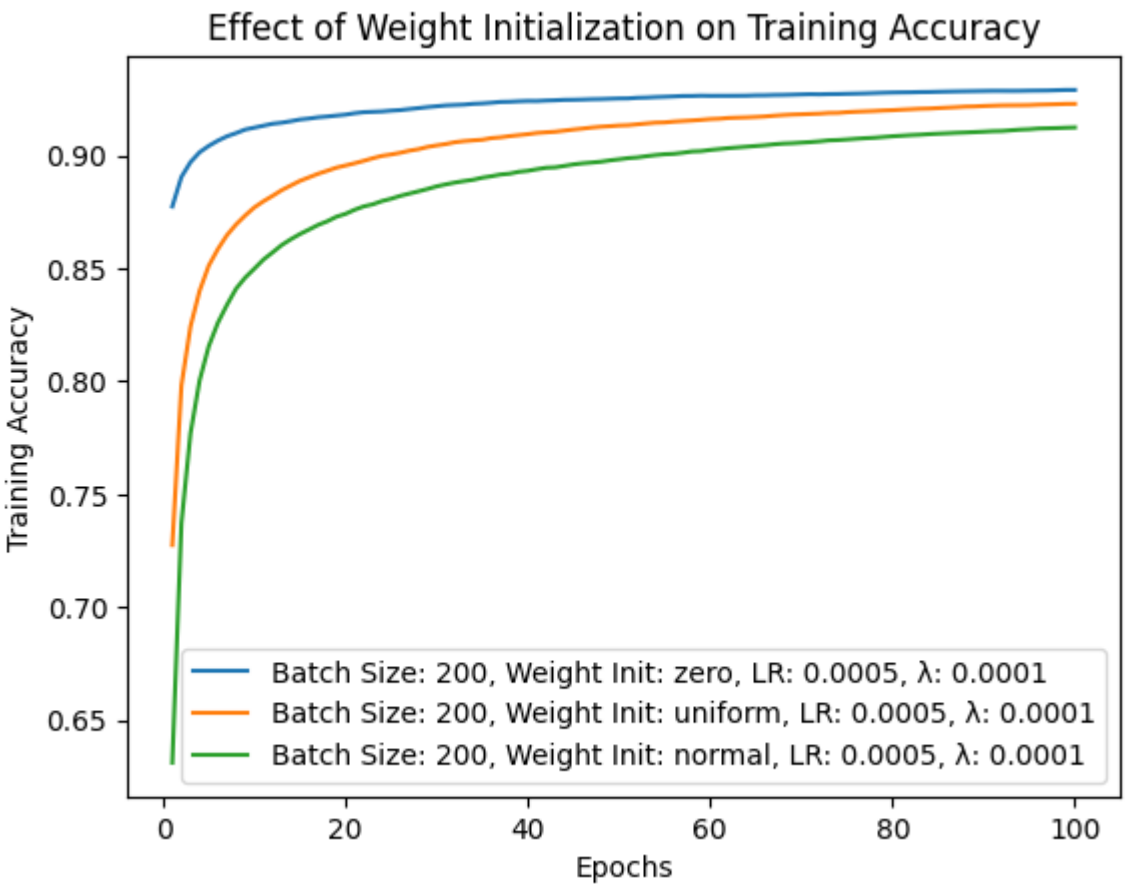
Confusion Matrix:

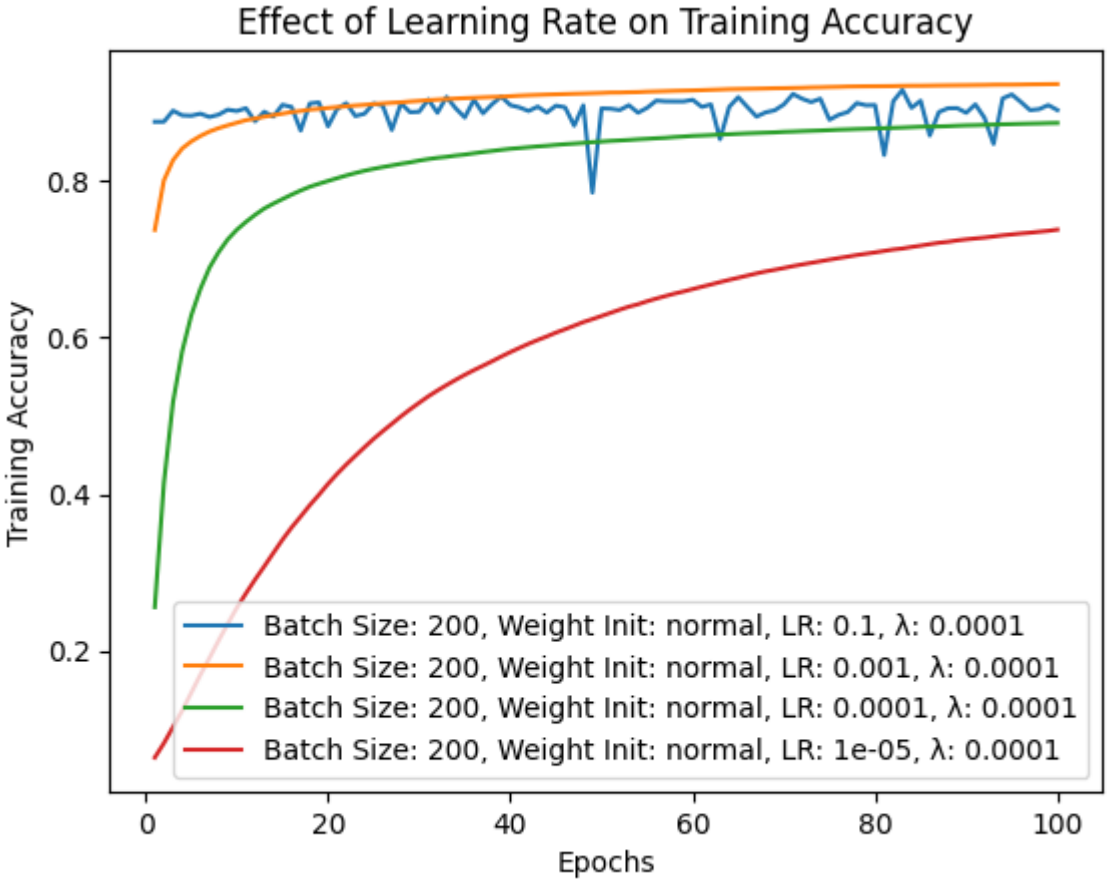
True/ Predicted	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	954	0	5	2	1	5	6	4	2	1
1	0	1104	4	4	0	1	4	2	15	1
2	5	13	903	27	5	6	12	9	46	6
3	6	2	20	894	1	37	4	9	27	10
4	3	2	7	2	898	0	14	4	10	42
5	11	3	6	45	13	747	14	8	35	10

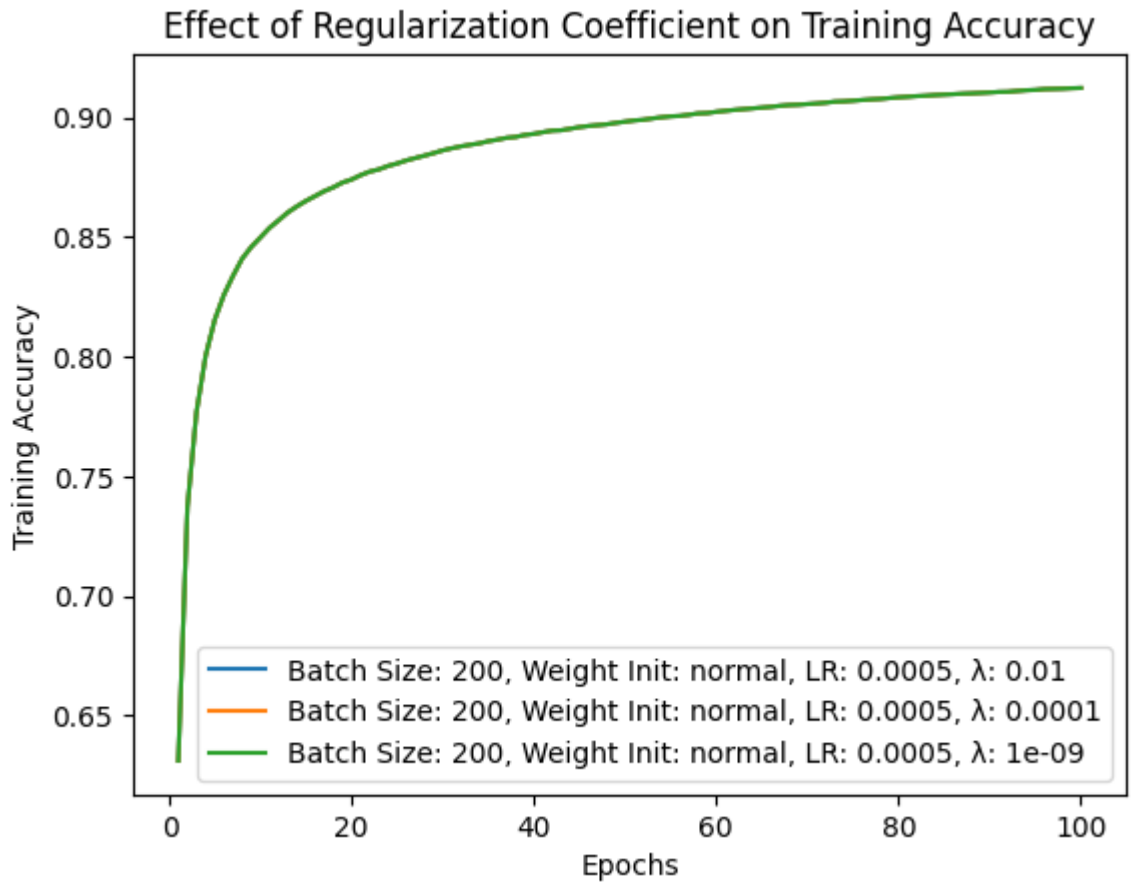
6	15	3	9	2	14	19	890	2	4	0
7	2	10	21	7	8	5	0	925	4	46
8	6	8	10	29	8	32	13	5	852	11
9	8	6	1	10	38	7	1	31	10	897

Q2.2.




Effect of Weight Initialization on Training Accuracy
Training Accuracy
Epochs
0.90
0.85
0.80
0.75
0.70
0.65
0
20
40
60
80
100
Batch Size: 200, Weight Init: zero, LR: 0.0005, λ: 0.0001
Batch Size: 200, Weight Init: uniform, LR: 0.0005, λ: 0.0001
Batch Size: 200, Weight Init: normal, LR: 0.0005, λ: 0.0001


Effect of Learning Rate on Training Accuracy
Training Accuracy
Epochs
0.8
0.6
0.4
0.2
0
20
40
60
80
100
Batch Size: 200, Weight Init: normal, LR: 0.1, λ: 0.0001
Batch Size: 200, Weight Init: normal, LR: 0.001, λ: 0.0001
Batch Size: 200, Weight Init: normal, LR: 0.0001, λ: 0.0001
Batch Size: 200, Weight Init: normal, LR: 1e-05, λ: 0.0001



After selecting the best among each configuration, the best weights are:

Batch Size: 1 and 64 are really close, but 1 is better according to the graph
Weight Initialization Technique: zero
Learning Rate: 0.001
Regularization Coefficient (λ): 0.01

Q2.3.

I trained both for batch size 1 and 64 since the results for them were very close.
After training the "best" model for batch size 1, the results are as follows:

Test Accuracy for the Best Model: 0.9009

Confusion Matrix:

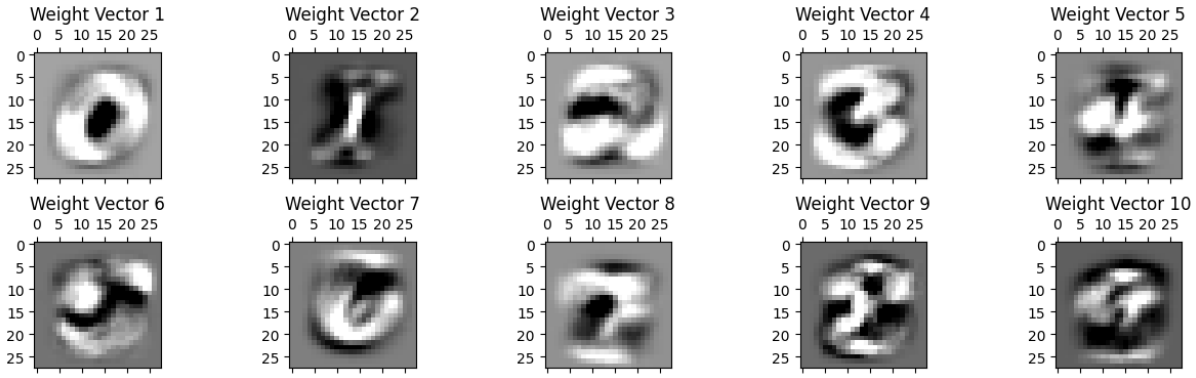
True/ Predicted	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	964	0	1	2	1	0	6	1	5	0

1	0	1110	2	4	1	2	4	0	12	0
2	13	16	864	26	15	0	18	19	49	12
3	5	2	16	904	1	32	5	15	16	14
4	2	7	3	1	889	0	10	1	8	61
5	13	7	0	41	14	736	17	11	41	12
6	17	3	4	2	9	18	900	1	4	0
7	2	22	19	5	11	0	0	910	5	54
8	11	16	8	29	9	34	13	14	823	17
9	12	8	4	10	31	13	1	16	5	909

When we find the “best” model configurations separately, we think of these configurations as independent since we do not change other configurations while testing different values for a specific configuration. However, in reality, these configurations are dependent on each other to some extent. Because of that, the separate best configuration values do not make up the overall “best” model. Even though batch size 1 was superior to batch size 64 by a little margin, the model with batch size 64 performs better and therefore is the best model.

Test Accuracy for the Best Model with Batch Size 64: 0.9227

Q2.4.



It can be said that the images of the vectors should represent specific characteristics of the digits that are classified.

- The first image looks associated with 0.
- The second image looks associated with 1.
- The third image looks associated with 2.
- The fourth image looks associated with 3.
- The fifth image looks associated with 4.
- The sixth image looks associated with 5.
- The seventh image looks associated with 6.
- The eighth image looks associated with 7.
- The ninth image looks associated with 8.
- The tenth image looks associated with 9.

Q2.5.

Metrics for Class 0:
Precision: 0.9278
Recall: 0.9837
F1 Score: 0.9549
F2 Score: 0.9720

Metrics for Class 1:
Precision: 0.9320
Recall: 0.9780
F1 Score: 0.9544
F2 Score: 0.9684

Metrics for Class 2:
Precision: 0.9381
Recall: 0.8372
F1 Score: 0.8848
F2 Score: 0.8556

Metrics for Class 3:
Precision: 0.8828
Recall: 0.8950
F1 Score: 0.8889
F2 Score: 0.8926

Metrics for Class 4:
Precision: 0.9062
Recall: 0.9053
F1 Score: 0.9058
F2 Score: 0.9055

Metrics for Class 5:
Precision: 0.8814
Recall: 0.8251
F1 Score: 0.8523
F2 Score: 0.8358

Metrics for Class 6:
Precision: 0.9240
Recall: 0.9395
F1 Score: 0.9317
F2 Score: 0.9363

Metrics for Class 7:
Precision: 0.9211
Recall: 0.8852
F1 Score: 0.9028
F2 Score: 0.8922

Metrics for Class 8:
Precision: 0.8502
Recall: 0.8450
F1 Score: 0.8476
F2 Score: 0.8460

Metrics for Class 9:
Precision: 0.8424
Recall: 0.9009
F1 Score: 0.8707
F2 Score: 0.8886

Inferences:

The performance of the classification model is quite good for digits 0 and 1. It shows that it can make accurate predictions with a high level of precision and recall, which implies that it performs well by covering actual instances. Also, class 2 does pretty well here though the recall drops slightly. Digit 3 seems to have found the right balance between precision and recall. On digit 4, this model seems to just get it, keeping accuracy and coverage on top, as evidenced by an excellent F1 score. Class 5 balances out precision and recall well, while digit six has accurate predictions with quality coverage. However, class seven does not fare much worse only that its recall is slightly lower. Likewise, digit eight shows moderate levels of both precision and recall scores. On the other hand, digit nine looks nice since both precision and recall are rather high. All in all, most classes seem to be predicted accurately by this model with slight moderation for those with moderate levels of either precision or recall.