Jiaming Li
CSI 5340 Intro to Deep Learning and Reinforcement Learning
Homework Exercise 2
Oct 5, 2021

Question 1

1. Parameters Settings:

The k of matrix A and B are randomly chosen from k = {1, 2, 3, 4, 5, 10}, and learning rate is decided as 0.001, epoch is 1000.

2. Plot Show

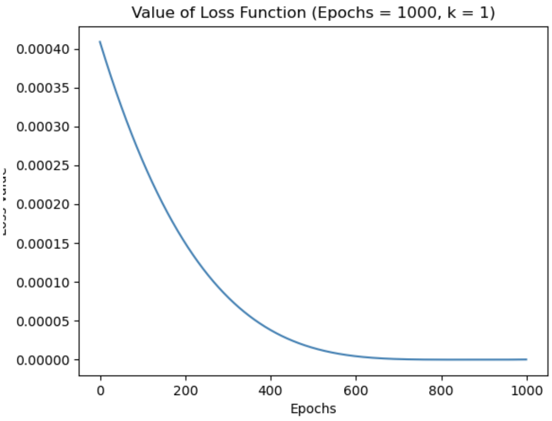


Figure 1: Value of Loss Function (when k = 1)

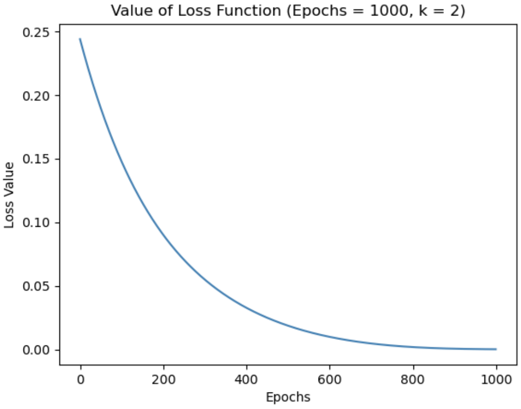


Figure 2: Value of Loss Function (when k = 2)

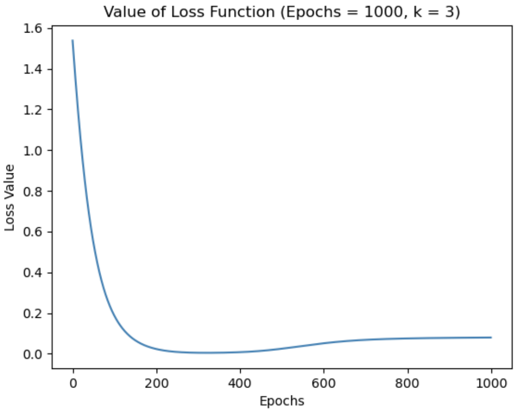


Figure 3: Value of Loss Function (when k = 3)

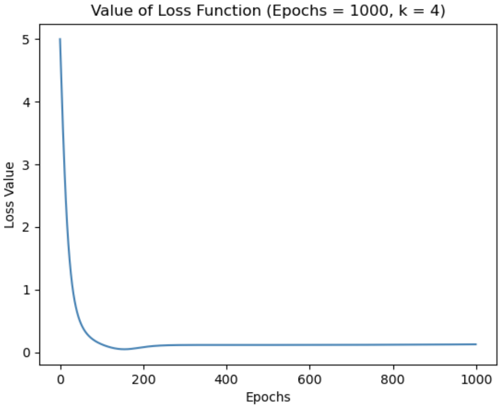
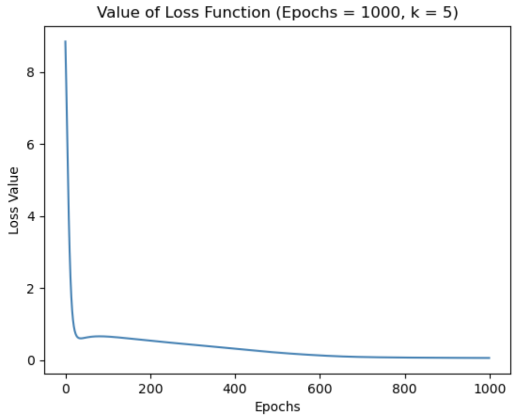


Figure 4: Value of Loss Function (when k = 4)



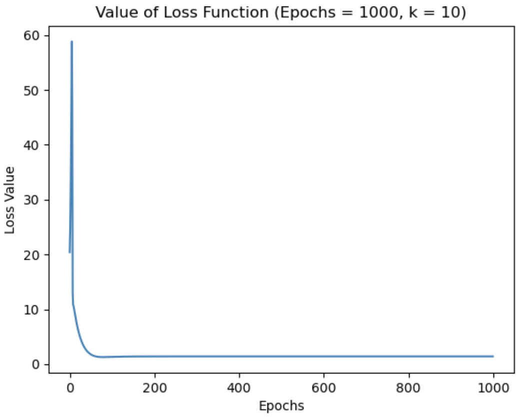


Figure 5: Value of Loss Function (when k = 5)

Figure 6: Value of Loss Function (when k = 10)

3. Conclusion

From the six figures below, it can be concluded that the loss drops rapidly as k increases. When k = 1 and 2, the curve is a bit flatten than the rest four plots. From k = 3, the curve starts to decrease quickly before epoch = 200, then becomes flatten after that and for k = 4, 5, 10, the curve is pretty much like a horizontal straight line.

Question 2

This question is realized by Python, which is `from scipy.special import softmax.` Four plots are showed below, which prove that H1 and H2 are equal.

1. Parameters Settings

Times = 10 (defined as 10 examples)
K and m are both integers randomly chosen from range 10 to 1000. The program prints the value of them, which are listed below:
m = 413, K = 690
m = 800, K = 659
m = 318, K = 858
m = 472, K = 512
m = 782, K = 334
m = 990, K = 847
m = 376, K = 822
m = 447, K = 564
m = 209, K = 45
m = 856, K = 740

2. Plot Show

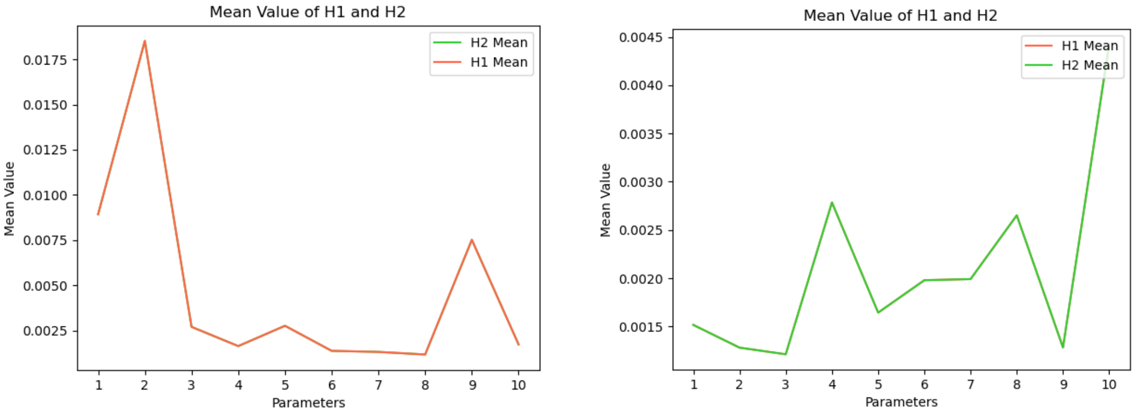


Figure 7: Mean Value of H1 and H2

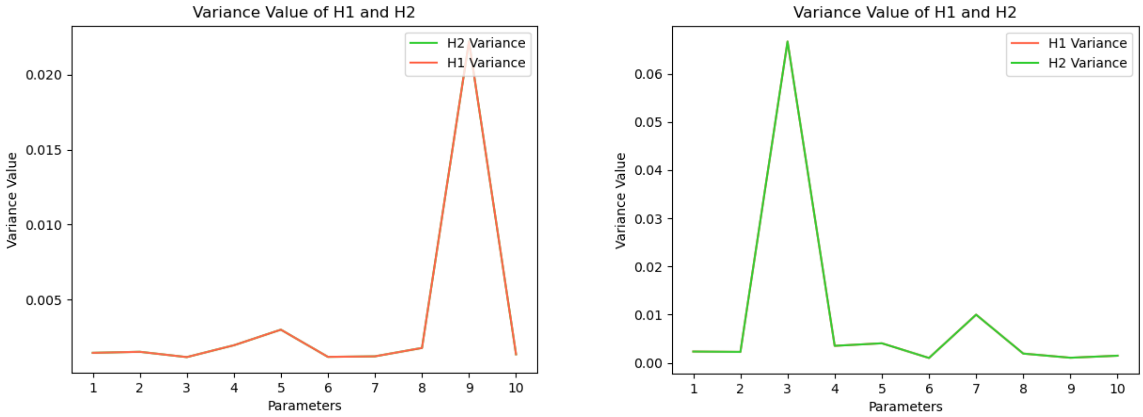


Figure 8: Variance Value of H1 and H2

3. Conclusion

Chart 1: Mean Comparison of H1 and H2

Parameters	H1 Mean	H2 Mean
m = 413, K = 690	0.0014492753623188397	0.0014492753623188805
m = 800, K = 659	0.0015174506828528036	0.0015174506828528073
m = 318, K = 858	0.001165501165501166	0.0011655011655011655
m = 472, K = 512	0.001953125000000006	0.001953125
m = 782, K = 334	0.0029940119760478914	0.0029940119760479044
m = 990, K = 847	0.0011806375442739124	0.001180637544274027
m = 376, K = 822	0.0012165450121654532	0.0012165450121654504
m = 447, K = 564	0.001773049645390061	0.0017730496453902118
m = 209, K = 45	0.022222222222222164	0.022222222222222223
m = 856, K = 740	0.0013513513513513454	0.0013513513513513514

Chart 2: Variance Comparison of H1 and H2

Parameters	H1 Variance	H2 Variance
m = 413, K = 690	0.000948005593542457	0.0014471749632430158
m = 800, K = 659	0.0009404210811293668	0.0015151480262779165
m = 318, K = 858	0.000844049948718861	0.0011641427725343813
m = 472, K = 512	0.001939461190815947	0.001949310302734375
m = 782, K = 334	0.002721888818238294	0.0029850478683351852
m = 990, K = 847	0.0005955354249832677	0.0011382263784258607
m = 376, K = 822	0.0004873652257093487	0.0012150650303988254
m = 447, K = 564	0.0017699031783830611	0.0017686477962320086
m = 209, K = 45	0.01673885922720778	0.021728395061728398
m = 856, K = 740	0.0010957425123628134	0.0013495252008765529

The two graphs show, respectively, when the H1 mean curve is on top and the H2 mean curve is on top. The two graphs perfectly show that H1 completely blocks H2 and H2 completely blocks H1. In this point, the mean values of two models are totally the same, which can prove that H1 and H2 are equal.

Question 3

This part introduces three models to train MNIST dataset of handwritten digits, which has a training set of 60,000 examples, and a test set of 10,000 examples.

1. Soft-max Regression Model

(1) Parameters Settings

random_seed = 1
learning_rate = 0.001
num_epochs = 10
batch_size = 8
num_features = 28 * 28
num_hidden = 100
num_classes = 10

(2) Plot show

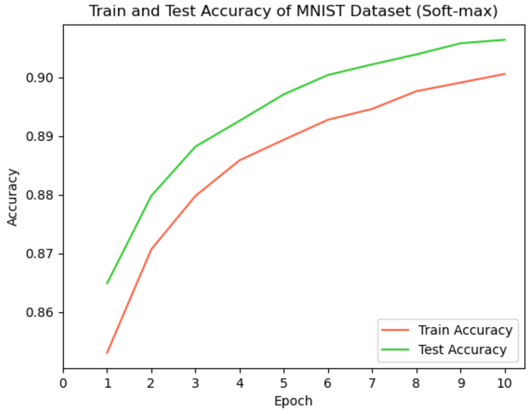


Figure 9: Accuracy (Soft-max Regression)

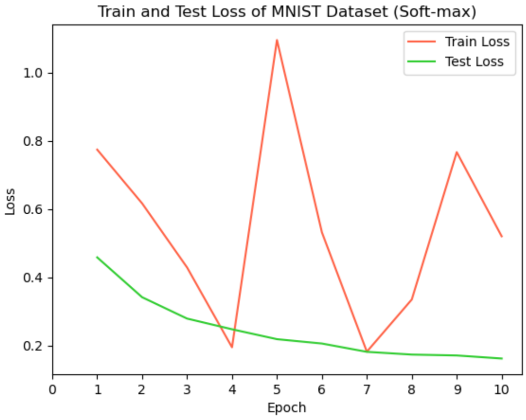


Figure 10: Loss (Soft-max Regression)

(3) Conclusion

Chart 3: Four Values of Soft-max Regression

Epoch	Train Accuracy	Train Loss	Test Accuracy	Test Loss
1	**0.8531**	0.7738	**0.8649**	**0.4583**
2	0.8707	0.6164	0.8798	0.3417
3	0.8798	0.4288	0.8882	0.2792
4	0.8859	0.1952	0.8926	0.2479
5	0.8894	1.0946	0.8971	0.2189
6	0.8928	0.5317	0.9004	0.2062
7	0.8946	**0.1822**	0.9022	0.1819
8	0.8976	0.3351	0.9039	0.1738
9	0.8991	0.7664	0.9058	0.1712
10	**0.9006**	**1.5200**	**0.9064**	**0.1623**

From Figure 9, 10 and Chart 3 showed above, it is clear that the test accuracy curve is always above the train accuracy curve which means that test accuracy is greater than train accuracy. Also, for the value of loss, which shows in Figure 8, the test loss decreases gradually whereas the train loss with larger fluctuations. It seems to decrease from epoch = 1 to 4, but it increases rapidly when epoch = 5 and also 9.

2. MLP Model

(1) Parameters Settings

random_seed = 1
learning_rate = 0.001
num_epochs = 10
batch_size = 8

num_features = 28 * 28
num_hidden = 100
num_classes = 10

(2) Plot show

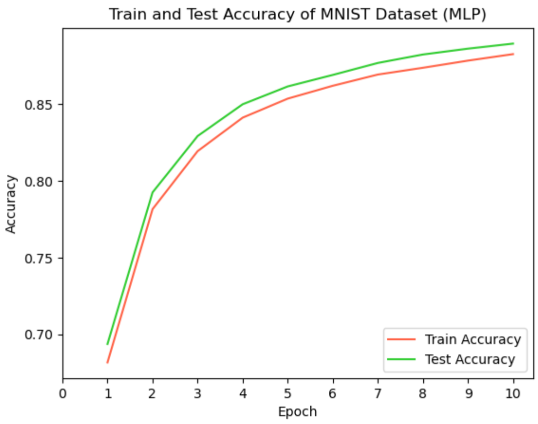


Figure 11: Accuracy (MLP)

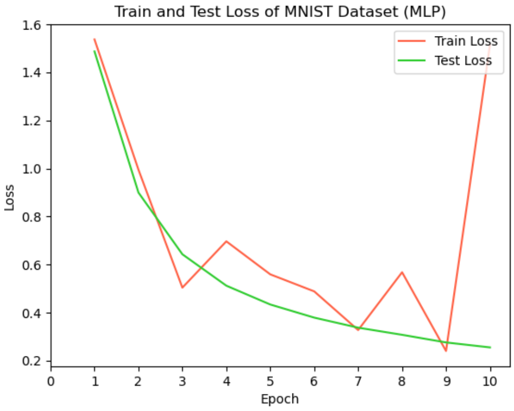


Figure 12: Loss (MLP)

(3) Conclusion

Chart 4: Four Values of MLP

Epoch	Train Accuracy	Train Loss	Test Accuracy	Test Loss
1	**0.6818**	**1.5369**	**0.6938**	**1.4871**
2	0.7816	0.9950	0.7925	0.8996
3	0.8194	0.5039	0.8292	0.6427
4	0.8412	0.6966	0.8499	0.5121
5	0.8536	0.5592	0.8615	0.4337
6	0.8619	0.4883	0.8690	0.3791
7	0.8692	0.3273	0.8768	0.3372
8	0.8737	0.5679	0.8823	0.3075
9	0.8783	**0.2403**	0.8861	0.2758
10	**0.8826**	1.5150	**0.8894**	**0.2551**

From Figure 11, 12 and Chart 4 above, the trend of train accuracy curve and test accuracy curve are similar. The test accuracy curve is still greater than train accuracy, which has the same situation as Soft-max Regression. However, the gap between train accuracy and test accuracy of MLP is much smaller the gap of Soft-max Regression.

For loss part, before the tenth epoch, the first nine epochs tend to be decreased even though train loss sometimes fluctuate. But on the tenth epoch, it is almost as great as it was at the beginning.

3. CNN Model

(1) Parameters Settings

random_seed = 1

batch_size = 8

num_epochs = 10

learning_rate = 0.001

dropout_p = 0.5

log_interval = 1

num_hidden_units = 50

num_classes = 10

decay_rate = 0.9999

(2) Plot show

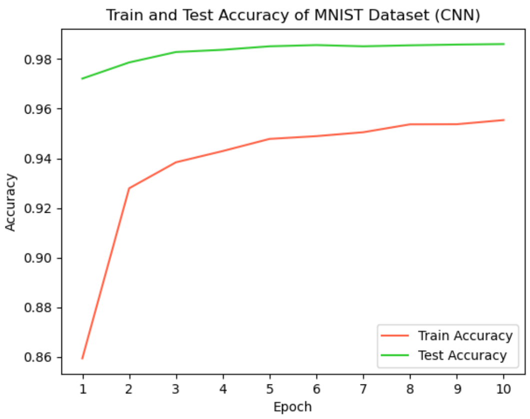


Figure 13: Accuracy (CNN)

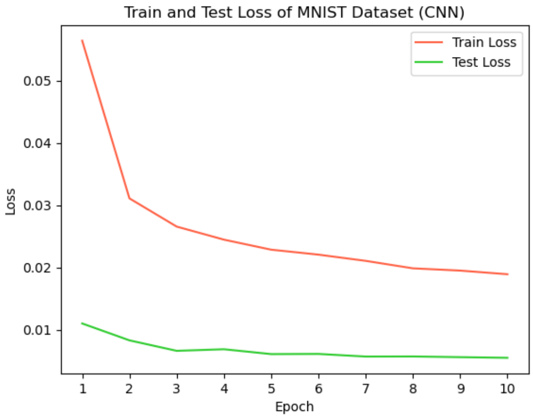


Figure 14: Loss (CNN)

(3) Conclusion

Chart 5: Four Values of CNN

Epoch	Train Accuracy	Train Loss	Test Accuracy	Test Loss
1	**0.8594**	**0.0564**	**0.9721**	**0.0110**
2	0.9279	0.0311	0.9786	0.0083
3	0.9384	0.0266	0.9828	0.0066
4	0.9429	0.0245	0.9837	0.0069
5	0.9478	0.0229	0.9851	0.0061
6	0.9489	0.0221	0.9856	0.0061
7	0.9505	0.0211	0.9851	0.0057
8	0.9537	0.0199	0.9855	0.0057
9	0.9537	0.0195	0.9858	0.0056
10	**0.9554**	**0.0189**	**0.9860**	**0.0055**

Compared to the values of Soft-max Regression and MLP, the trend and value of CNN are much more ideal as Figure 13, 14 and Chart 5 show since both train accuracy and test accuracy increase from the first epoch, and both train loss and test loss decrease from the first epoch. And from what we conclude for Soft-max Regression and MLP model, test accuracy is always greater than train accuracy, and test loss is always smaller than train loss.