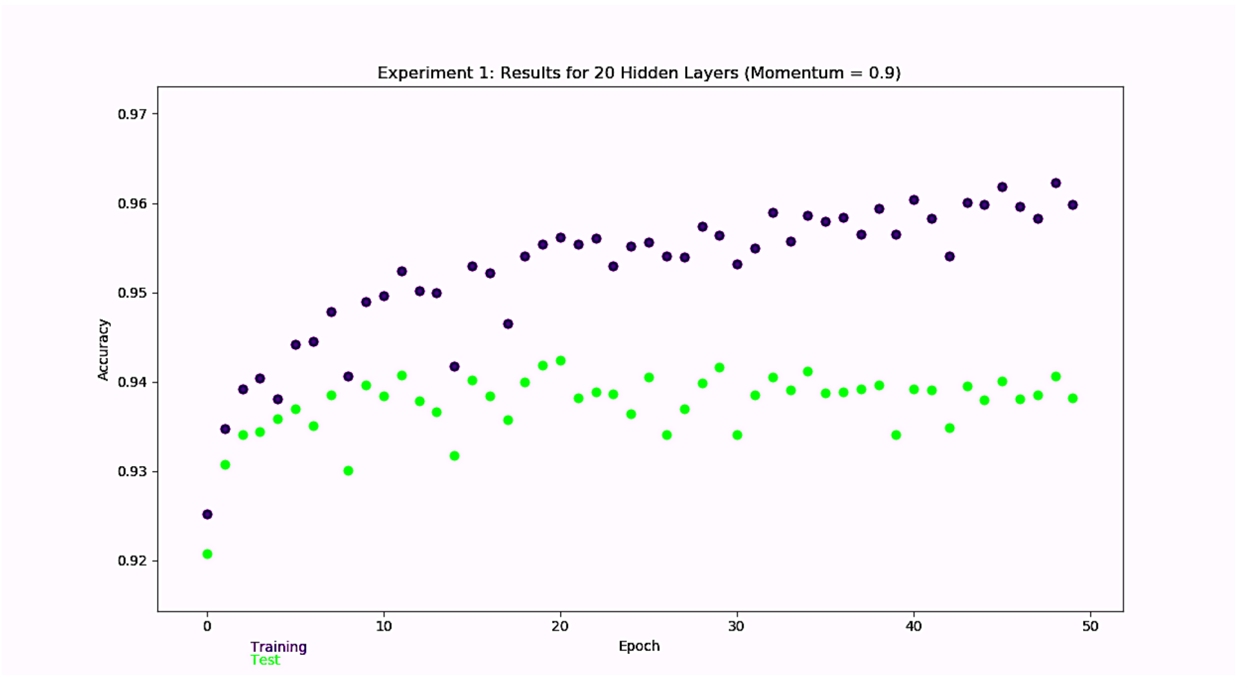
Machine Learning

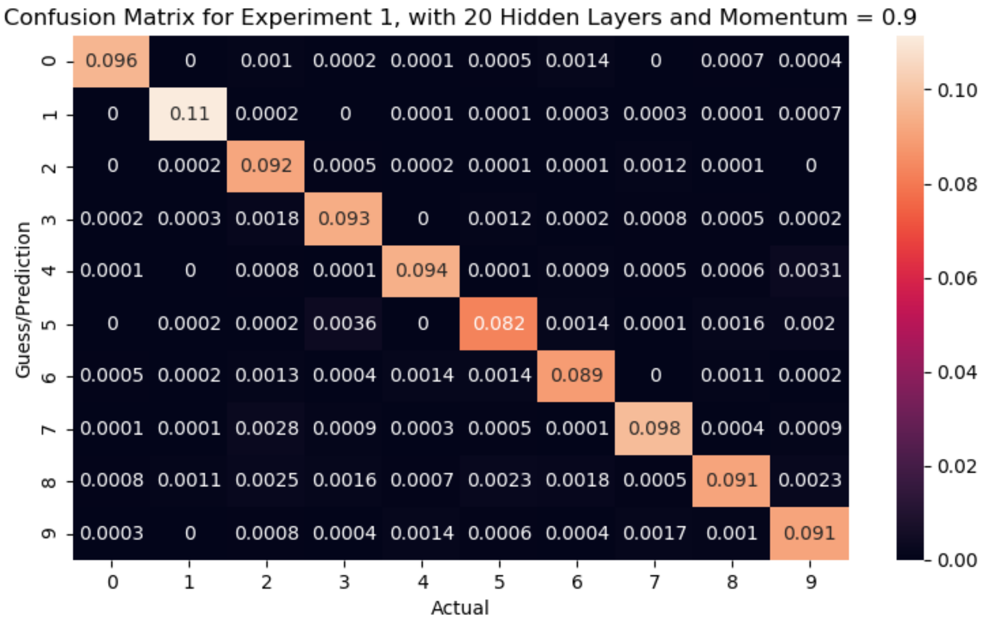
Programming Assignment 1

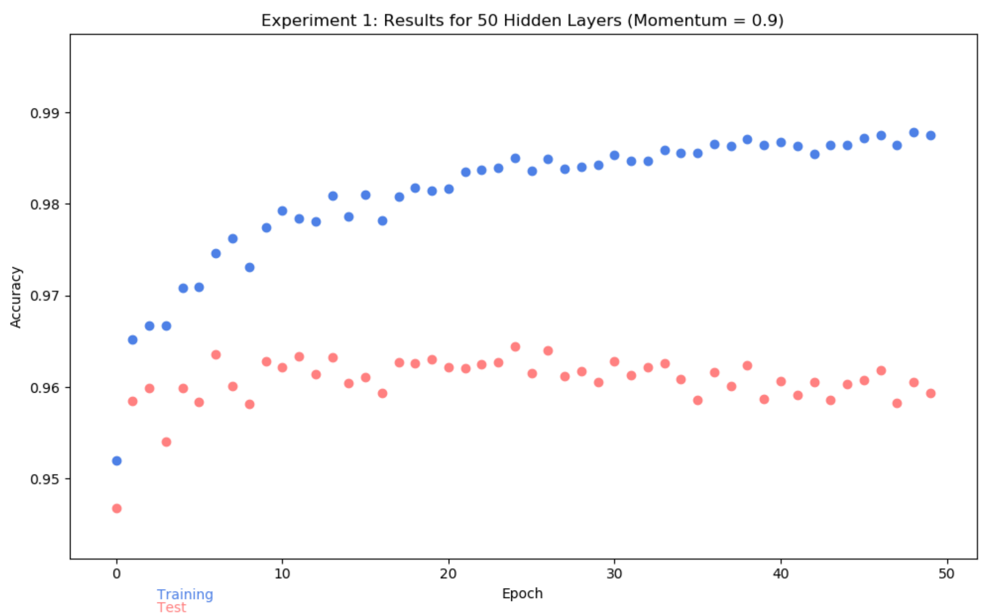
Portland State U, Spring 2020 04//2020

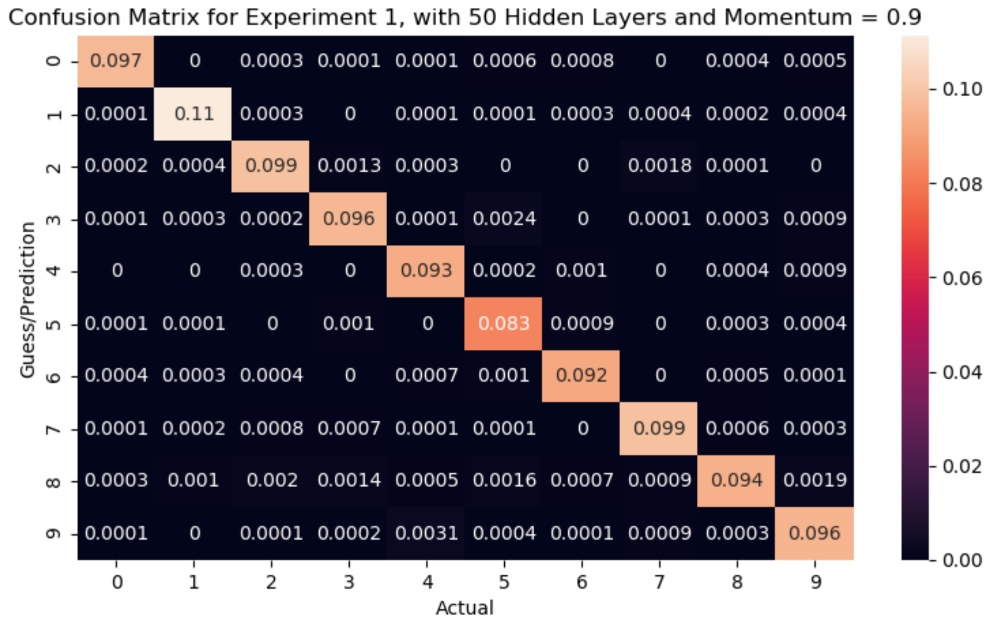
Student: Patrick Rademacher Due: 6:00pm PDT, 04/26/2020

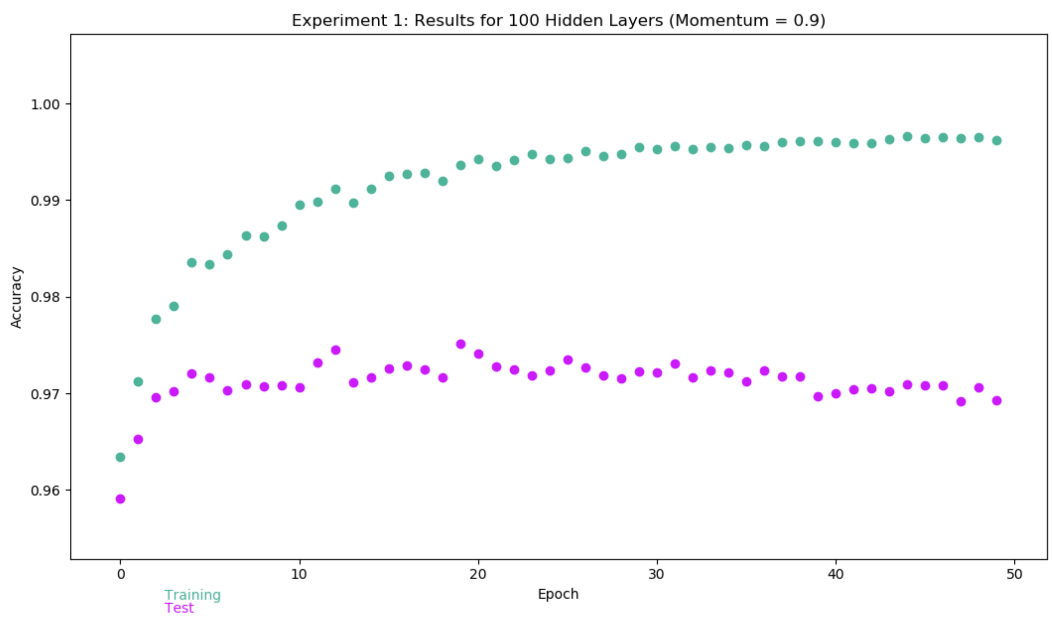
EXPERIMENT #1 REPORT

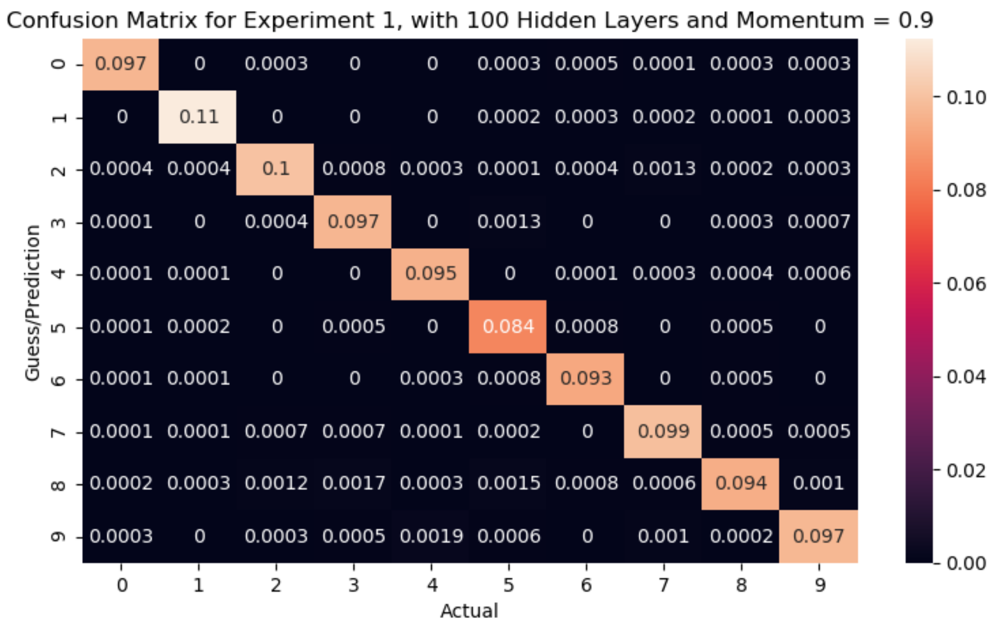



Confusion Matrix for Experiment 1, with 20 Hidden Layers and Momentum = 0.9
Guess/Prediction
Actual
0.096 0 0.001 0.0002 0.0001 0.0005 0.0014 0 0.0007 0.0004
0 0.11 0.0002 0 0.0001 0.0001 0.0003 0.0003 0.0001 0.0007
0 0.0002 0.092 0.0005 0.0002 0.0001 0.0001 0.0012 0.0001 0
0.0002 0.0003 0.0018 0.093 0 0.0012 0.0002 0.0008 0.0005 0.0002
0.0001 0 0.0008 0.0001 0.094 0.0001 0.0009 0.0005 0.0006 0.0031
0 0.0002 0.0002 0.0036 0 0.082 0.0014 0.0001 0.0016 0.002
0.0005 0.0002 0.0013 0.0004 0.0014 0.0014 0.089 0 0.0011 0.0002
0.0001 0.0001 0.0028 0.0009 0.0003 0.0005 0.0001 0.098 0.0004 0.0009
0.0008 0.0011 0.0025 0.0016 0.0007 0.0023 0.0018 0.0005 0.091 0.0023
0.0003 0 0.0008 0.0004 0.0014 0.0006 0.0004 0.0017 0.001 0.091
0.10
0.08
0.06
0.04
0.02
0.00


Experiment 1: Results for 50 Hidden Layers (Momentum = 0.9)
Accuracy
0.99
0.98
0.97
0.96
0.95
0
10
20
30
40
50
Epoch
Training
Test

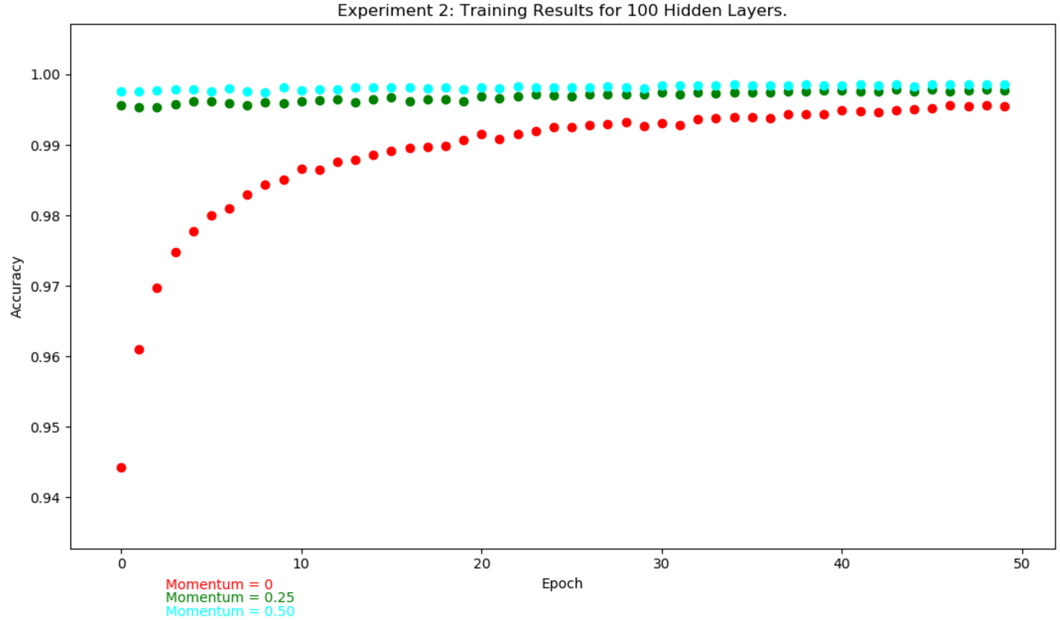

Confusion Matrix for Experiment 1, with 50 Hidden Layers and Momentum = 0.9
Guess/Prediction
Actual


Experiment 1: Results for 100 Hidden Layers (Momentum = 0.9)
Accuracy
Epoch
Training
Test

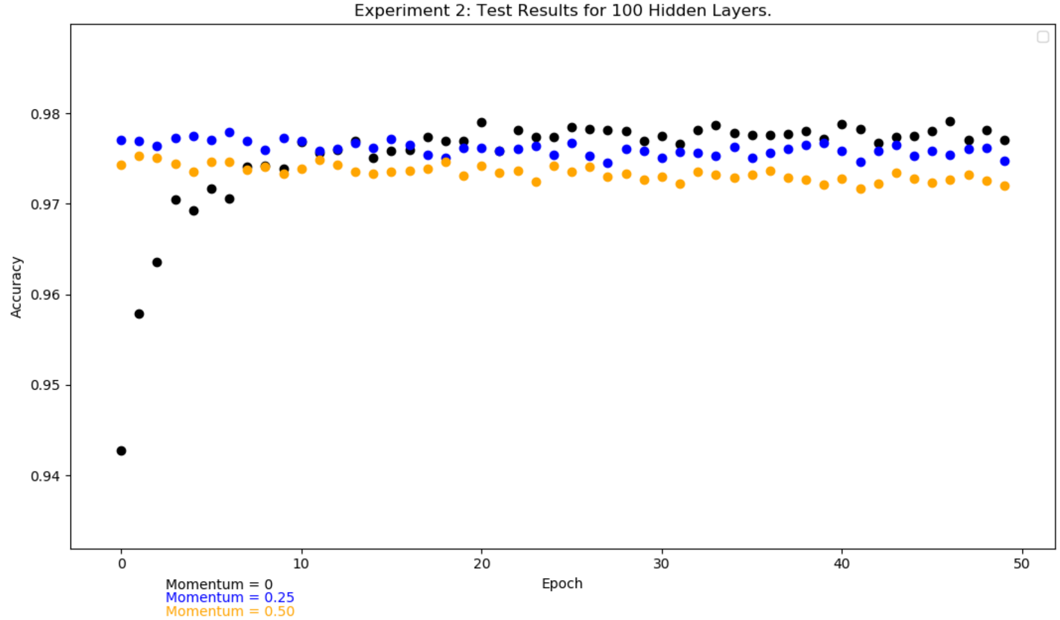


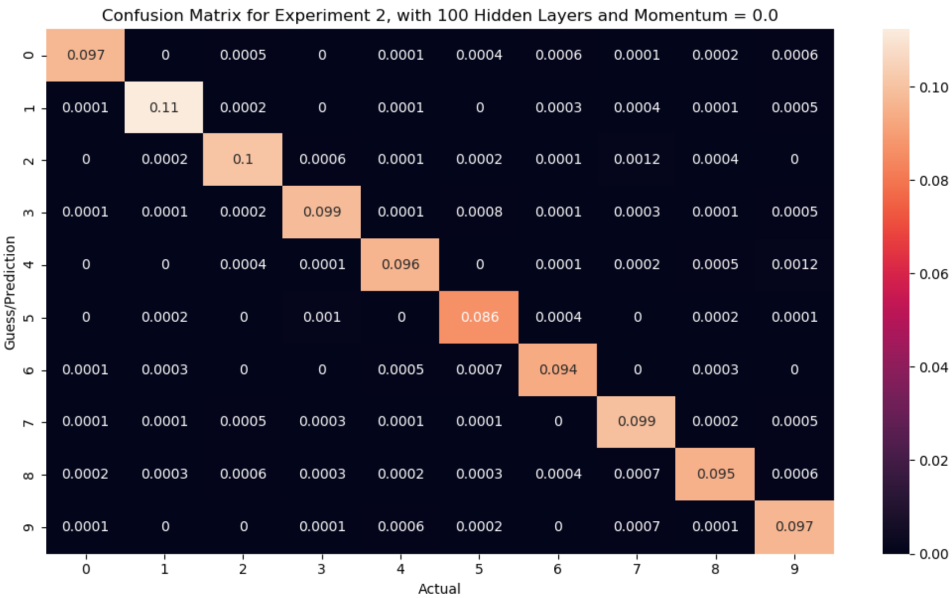
The number of hidden layers - when momentum is equal among all perceptrons - plays a fairly significant role in the overall accuracy for 50 epochs when using this test data. First and foremost, the initial accuracy at $epoch_0$ is higher when the amount of hidden layers is higher. Additionally, the final accuracy also follows this correlation, as one can see from the plots and matrices provided above. The final accuracy for 20, 50, and 100 hidden layers is 93.6%, 95.9%, and 96.6%, respectively. It's also somewhat hard to see without a close analysis, but more hidden layers also results in training to converge at a faster rate, meaning it does not require as many epochs to settle in comparison to perceptrons with less hidden layers. Nonetheless, the final accuracy per perceptron for the test set is less than the accuracy on the training set, indicating that there is some overfitting occurring, where it is somewhat worse as the hidden layers increase. These results are fairly similar to what I had in my first homework assignment.

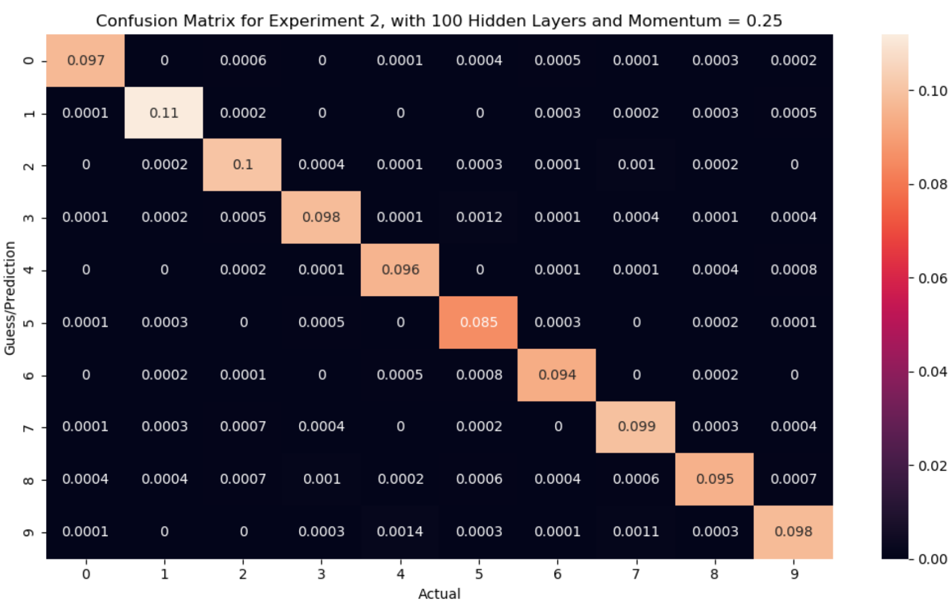
Experiment 2: Training Results for 100 Hidden Layers.

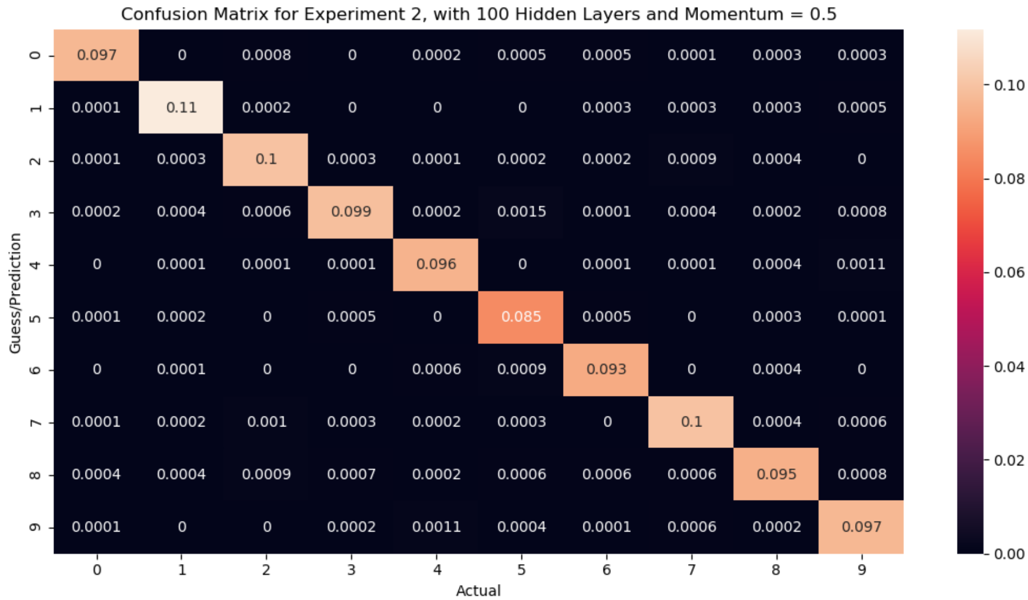


Experiment 2: Test Results for 100 Hidden Layers.



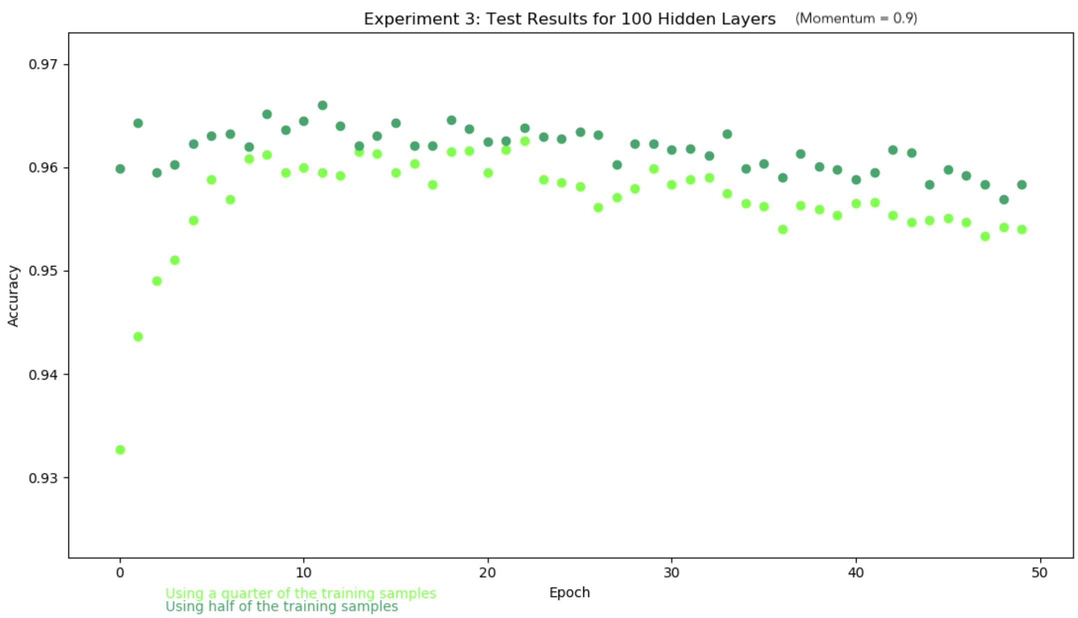

Confusion Matrix for Experiment 2, with 100 Hidden Layers and Momentum = 0.0
Guess/Prediction
Actual
0 1 2 3 4 5 6 7 8 9
0 0.097 0 0.0005 0 0.0001 0.0004 0.0006 0.0001 0.0002 0.0006
1 0.0001 0.11 0.0002 0 0.0001 0 0.0003 0.0004 0.0001 0.0005
2 0 0.0002 0.1 0.0006 0.0001 0.0002 0.0001 0.0012 0.0004 0
3 0.0001 0.0001 0.0002 0.099 0.0001 0.0008 0.0001 0.0003 0.0001 0.0005
4 0 0 0.0004 0.0001 0.096 0 0.0001 0.0002 0.0005 0.0012
5 0 0.0002 0 0.001 0 0.086 0.0004 0 0.0002 0.0001
6 0.0001 0.0003 0 0 0.0005 0.0007 0.094 0 0.0003 0
7 0.0001 0.0001 0.0005 0.0003 0.0001 0.0001 0 0.099 0.0002 0.0005
8 0.0002 0.0003 0.0006 0.0003 0.0002 0.0003 0.0004 0.0007 0.095 0.0006
9 0.0001 0 0 0.0001 0.0006 0.0002 0 0.0007 0.0001 0.097
0.10
0.08
0.06
0.04
0.02
0.00

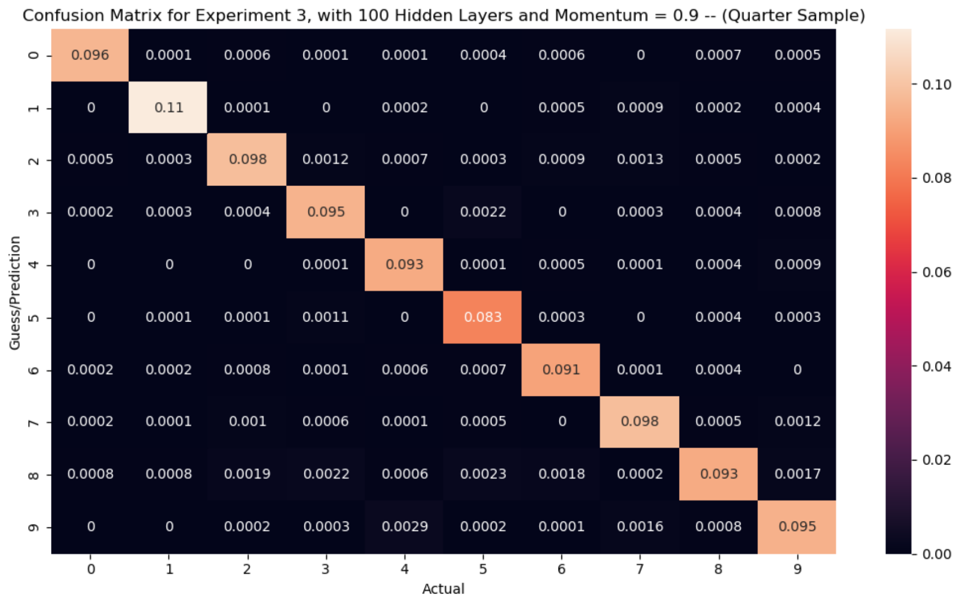

Confusion Matrix for Experiment 2, with 100 Hidden Layers and Momentum = 0.25
Guess/Prediction
Actual
0 1 2 3 4 5 6 7 8 9
0 0.097 0 0.0006 0 0.0001 0.0004 0.0005 0.0001 0.0003 0.0002
1 0.0001 0.11 0.0002 0 0 0 0.0003 0.0002 0.0003 0.0005
2 0 0.0002 0.1 0.0004 0.0001 0.0003 0.0001 0.001 0.0002 0
3 0.0001 0.0002 0.0005 0.098 0.0001 0.0012 0.0001 0.0004 0.0001 0.0004
4 0 0 0.0002 0.0001 0.096 0 0.0001 0.0001 0.0004 0.0008
5 0.0001 0.0003 0 0.0005 0 0.085 0.0003 0 0.0002 0.0001
6 0 0.0002 0.0001 0 0.0005 0.0008 0.094 0 0.0002 0
7 0.0001 0.0003 0.0007 0.0004 0 0.0002 0 0.099 0.0003 0.0004
8 0.0004 0.0004 0.0007 0.001 0.0002 0.0006 0.0004 0.0006 0.095 0.0007
9 0.0001 0 0 0.0003 0.0014 0.0003 0.0001 0.0011 0.0003 0.098
0.10
0.08
0.06
0.04
0.02
0.00

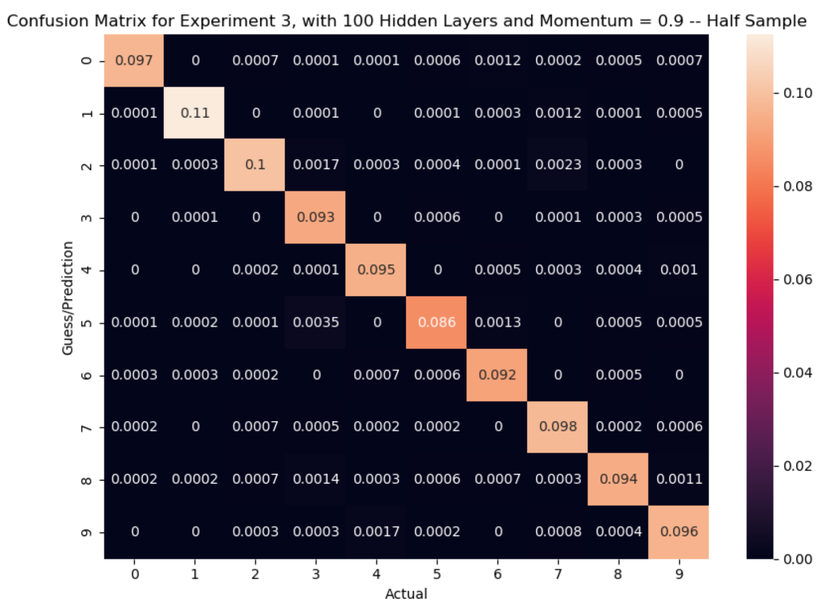


The final accuracy for each model in experiment 2 were almost identical. The one with a momentum value of 0.0 finished with an accuracy of 97.3% while the other two had a end result of 97.2%. However, what's interesting is how the models reached these performances over the course of 50 epochs. When looking at the plots, our perceptron with momentum 0.0 started significantly lower than the other two models, approximately at 94% in comparison to the others floating somewhere between 97% and 98%, where they pretty much remain the whole experiment. Yet it is the one who has drastic change from epoch to epoch that ends up with the best percentage, and also has the smallest margin of difference between the accuracy of the training set and testing set. This demonstrates that the other two models with momentum values more towards the middle of the spectrum result in more overfitting behavior, as it's somewhat obvious when seeing that the training set learning has a slight, steady increase while the test set shows us that there is some decreasing and lack of consistency in comparison. The perceptron with momentum at 0.0 seems to resemble the perceptron from experiment 1 with the same number of hidden layers at a 0.9 momentum value, where both models show increases in learning and less stagnating outcomes than momentum values more towards the middle.


Experiment 3: Training Results for 100 Hidden Layers
(Momentum = 0.9)
Accuracy
1.01
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0
10
20
30
40
50
Epoch
Using a quarter of the training samples
Using half of the training samples


Experiment 3: Test Results for 100 Hidden Layers
(Momentum = 0.9)
Accuracy
0.97
0.96
0.95
0.94
0.93
0
10
20
30
40
50
Epoch
Using a quarter of the training samples
Using half of the training samples





Looking at our last experiment, the correlation we see indicates that as the size of the training data decreases or gets smaller, the accuracy does the same. When using a quarter of the original training data size, the final accuracy is 95.2% and 96.1% when using half of the original training data size. There is also a correlation with the initial accuracies on both the training and test data being smaller when training set size is decreased as well. Lastly, the overfitting is worse for the quarter sample, as it bounces around and eventually starts to decrease in performance while the training set shows a very constant rate. The same goes for the half sample, though its dips in error are not as drastic.