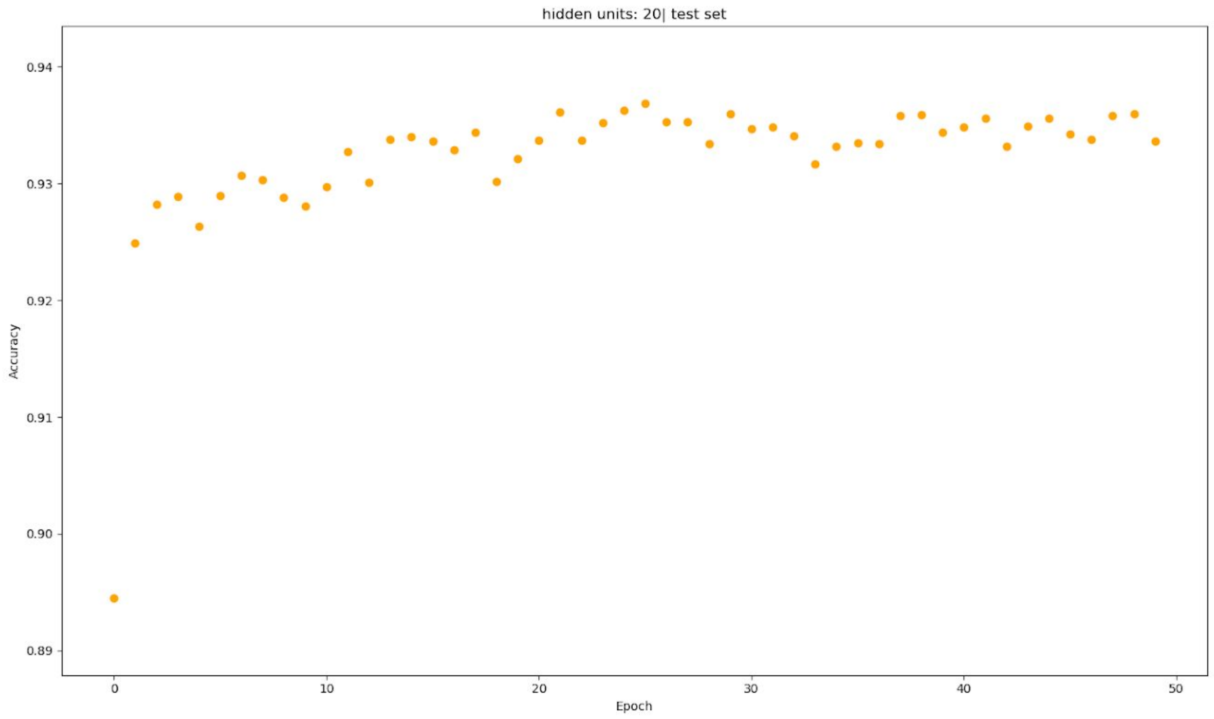
Charles Bolton
Machine Learning
4/26/2020

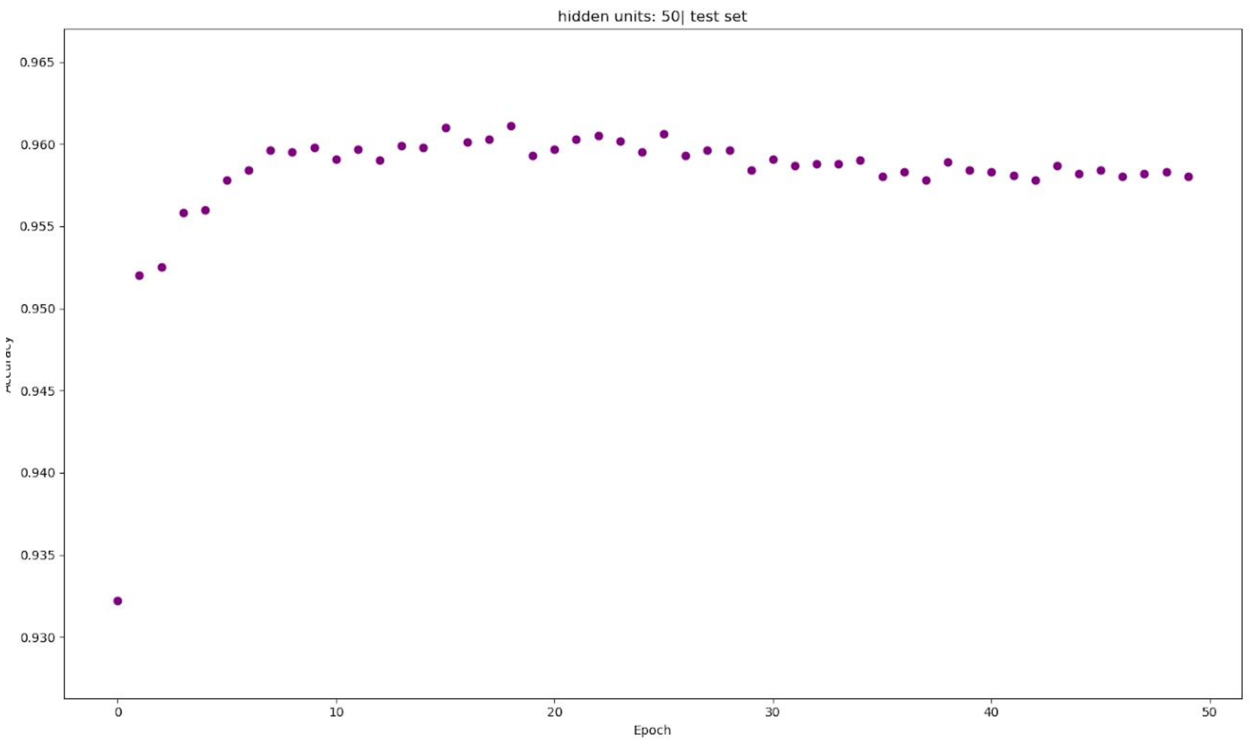
Multilayer Perceptron on MNIST Dataset with One Hidden Layer

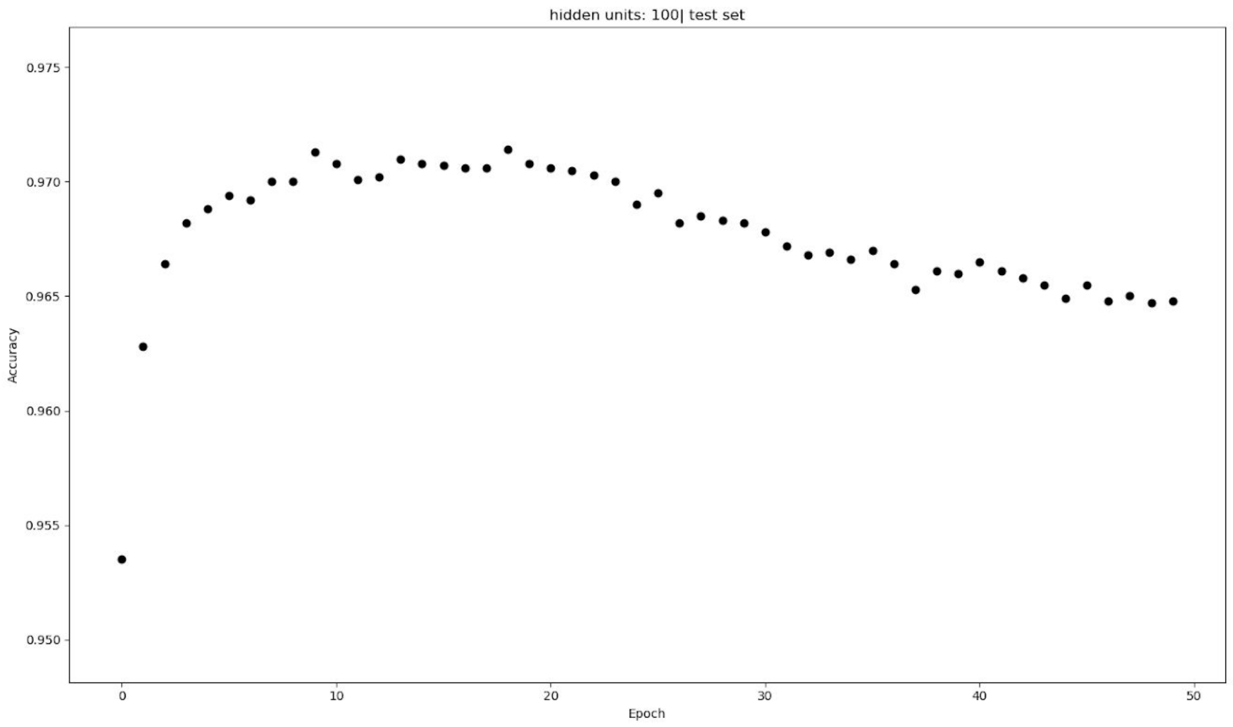
For the following report, the different scatter plots indicate (at the top) what variable was being tested, and which set in {train, test} the data represents. The y-axis represents accuracy and the x-axis the number of epochs.

Experiment 1: Vary Number of Hidden Units

How does the number of hidden units affect the final accuracy on the test data?







From the three plots above, it was observed that the final accuracy rate with just **20** hidden units was between 92-94%.

With **50** hidden units, the final accuracy was a little under 96%.

With **100** hidden units, the final accuracy was just above 96.5%, but the peak accuracy was above 97% around 18 epochs.

How does it affect the number of epochs needed for training to converge?

With **20** hidden units, the accuracy appears to stop increasing around the 25th epoch, and in general, performs worse than the model with **50** hidden units. The accuracy rate tended to oscillate more than the trials with **50** and **100** units, and it shows that this model is less stable in its predictions. It nonetheless converges to within half a percent after 25 epochs.

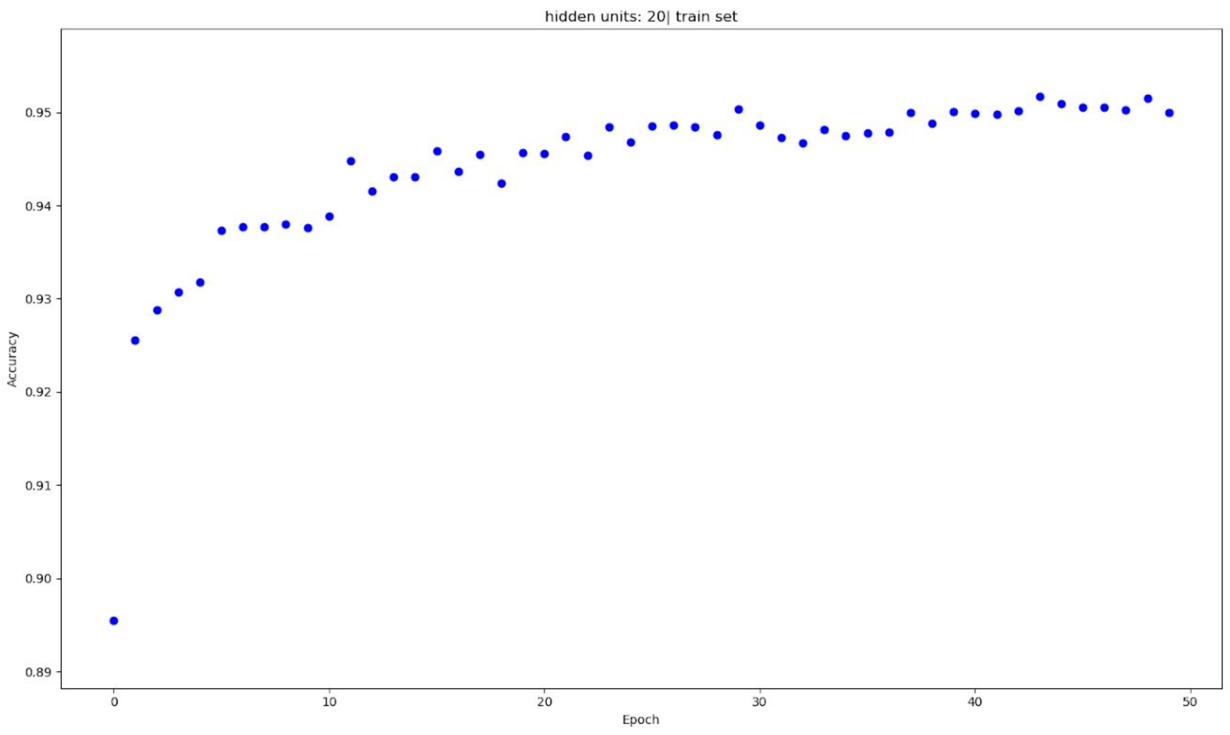
With **50** hidden units, it converges within half a percent after less than 5 epochs, and within a tenth of a percent in 30 epochs. This model appears to be the most stable in the long run.

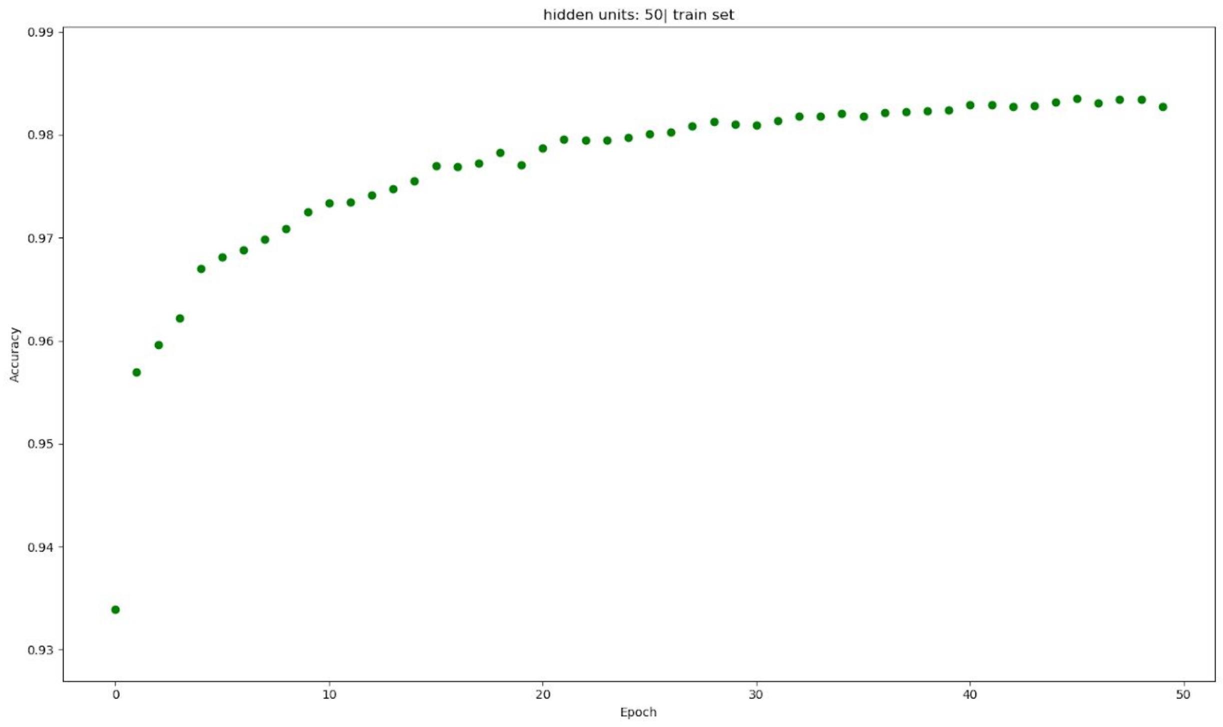
With **100** hidden units, the model converges to within a half a percent after 20 epochs, but shows signs of decreasing accuracy over time.

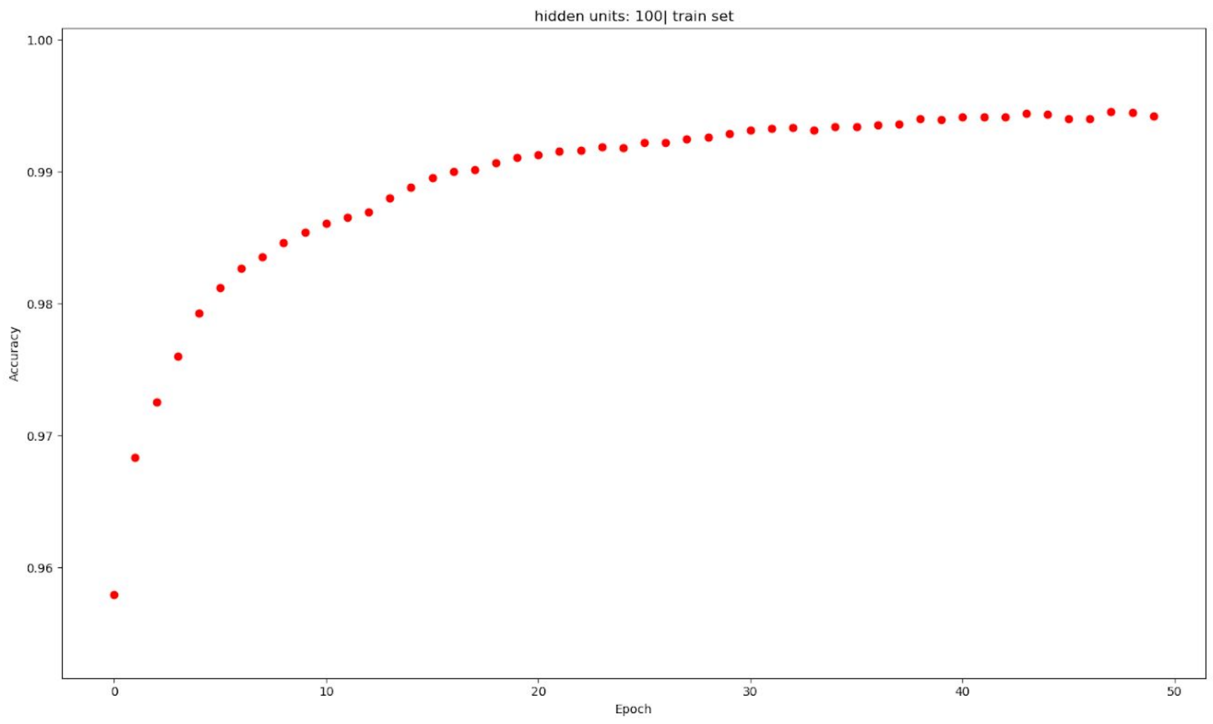
Is there evidence that any of your networks has overfit to the training data? If so, what is that evidence?

With **100** units, the training accuracy approaches 100% but the test accuracy actually begins to perform worse, which is evidence that this model is overfitting. To see this, compare the red plot below (100/train) with the black plot above (100/test).

For **20** and **50** hidden units, the train accuracy gets to 95% and 98%, respectively, but the test accuracy doesn't begin to drastically decrease, so there is less evidence of overfitting, but nonetheless the test accuracy isn't keeping pace with the training accuracy, so some overfitting is occuring.


hidden units: 20| train set
Accuracy
Epoch
0.89
0.90
0.91
0.92
0.93
0.94
0.95
0
10
20
30
40
50


hidden units: 50| train set
Accuracy
Epoch
0.93
0.94
0.95
0.96
0.97
0.98
0.99
0
10
20
30
40
50



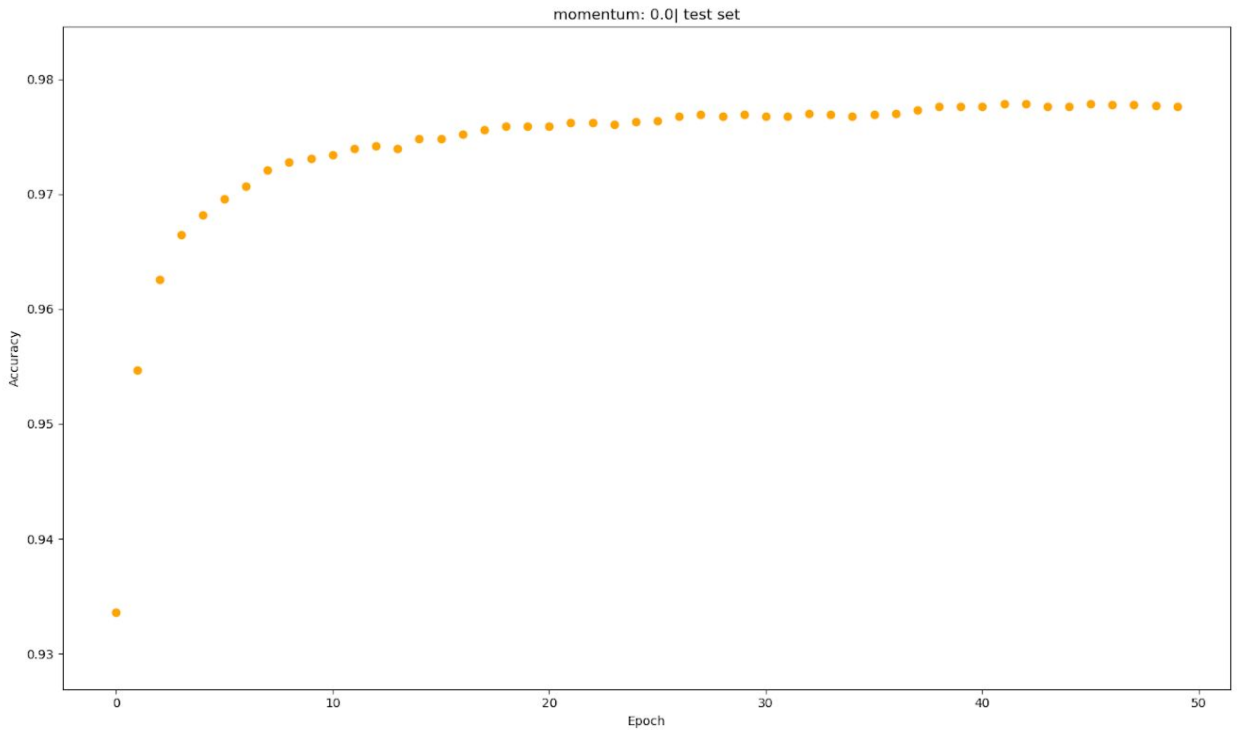
How do your results compare to the results obtained by your perceptron in HW 1?

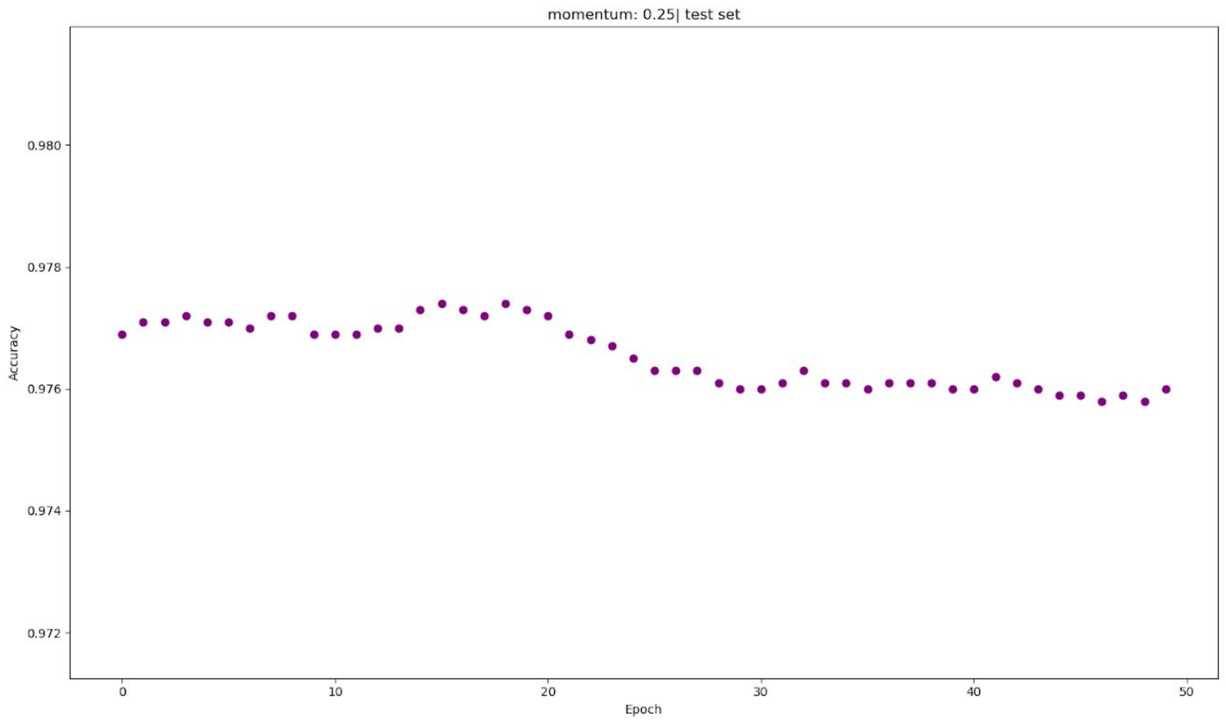
The results here are much better than the accuracies observed in HW1. Although here we are not stopping the trials after convergence, the overall accuracy was never better than 90% for the classic perceptron. The hidden layer gives us that extra 5-8% in improved accuracy.

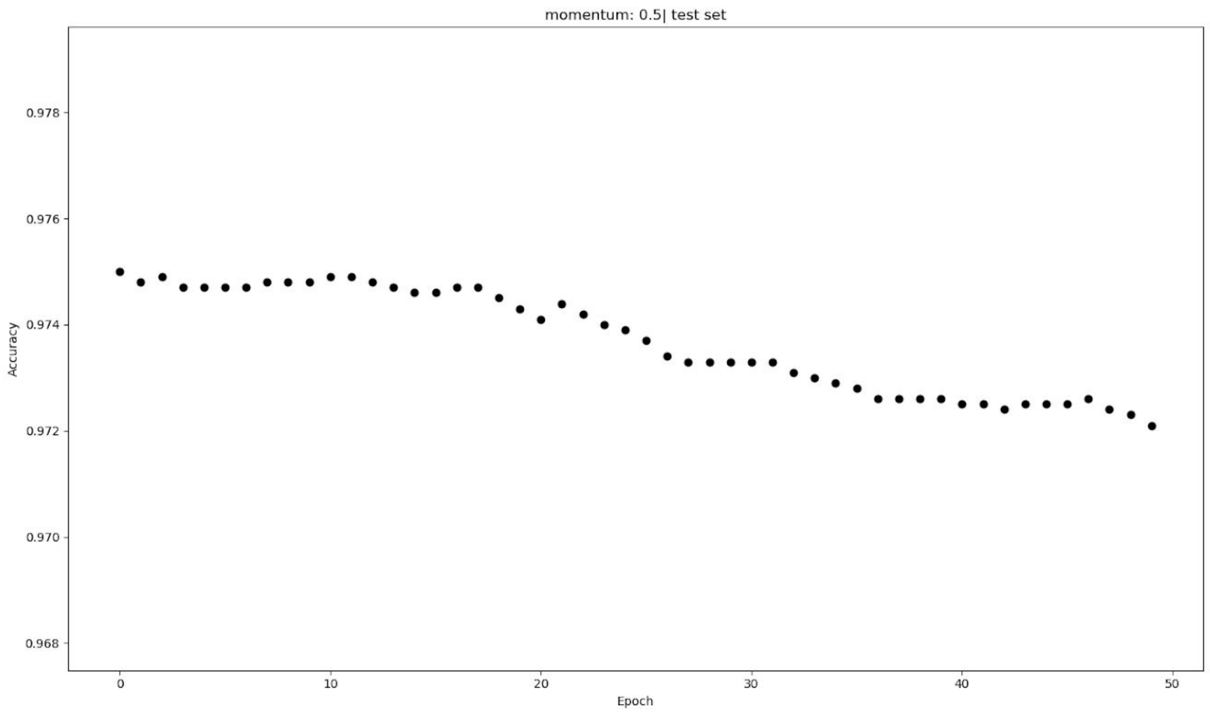
Experiment Two: Vary Momentum

How does the momentum value affect the final accuracy on the test data?

The more momentum, the worse the final accuracy, as you can see below, the test set accuracy with zero momentum performs the best, while the other two, with .24 and .5 respectively, perform worse.

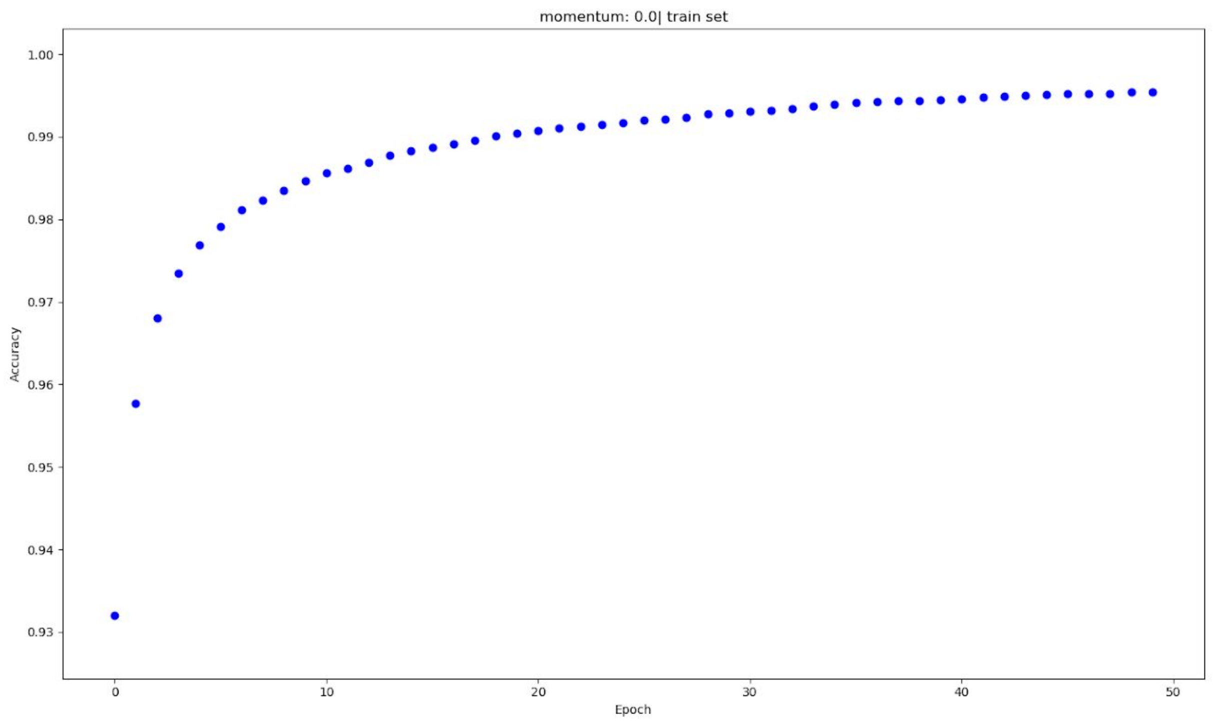

momentum: 0.0| test set
Accuracy
Epoch
0.98
0.97
0.96
0.95
0.94
0.93
0
10
20
30
40
50

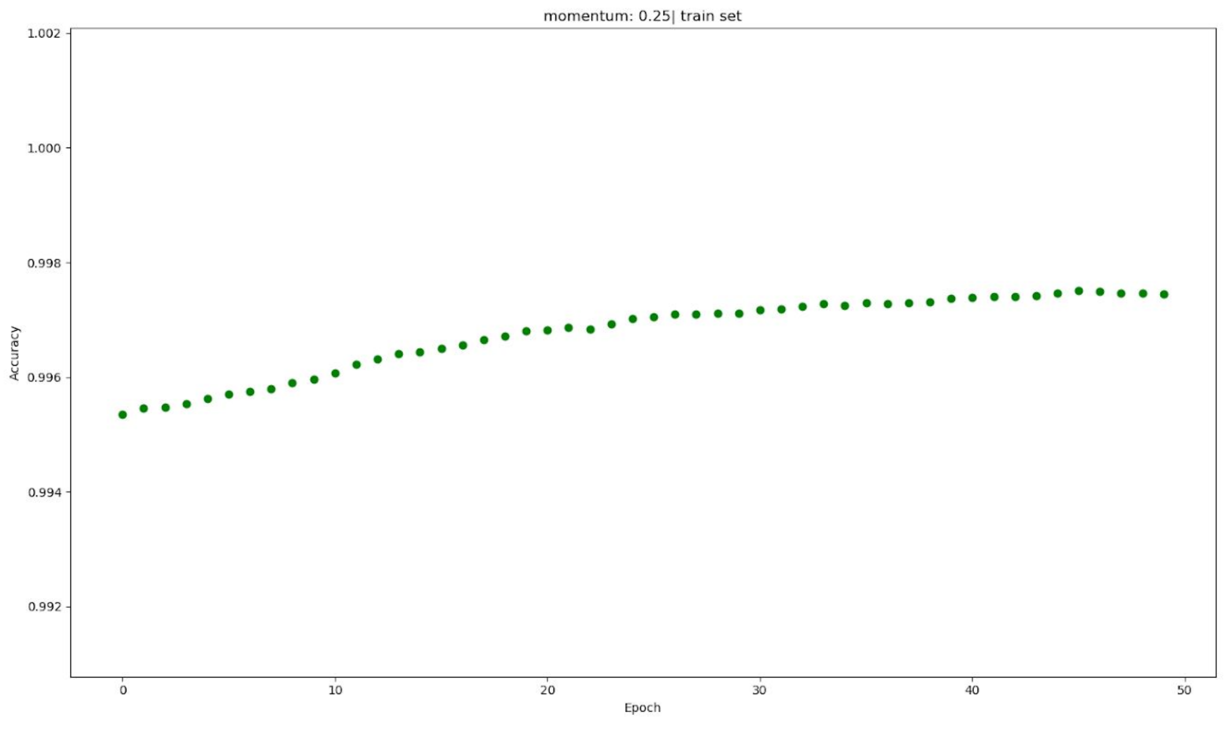

momentum: 0.25| test set
Accuracy
Epoch
0.980
0.978
0.976
0.974
0.972
0
10
20
30
40
50

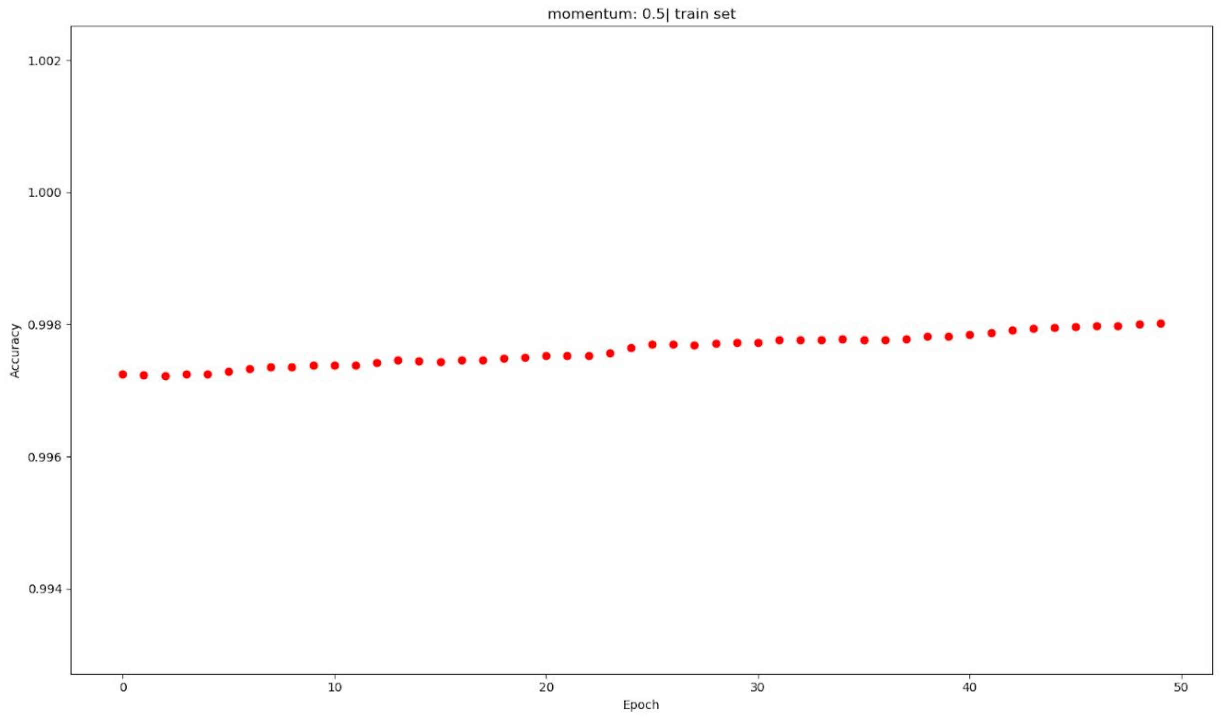


How does it affect the number of epochs needed for training to converge?

Convergence is immediate with a momentum term of .5, and more gradual with zero momentum, only converging at the end of 50 epocs. With momentum of .25, convergence occurs after roughly 40 epochs.


momentum: 0.0| train set
Accuracy
Epoch
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0
10
20
30
40
50


momentum: 0.25| train set
Accuracy
Epoch
1.002
1.000
0.998
0.996
0.994
0.992
0
10
20
30
40
50

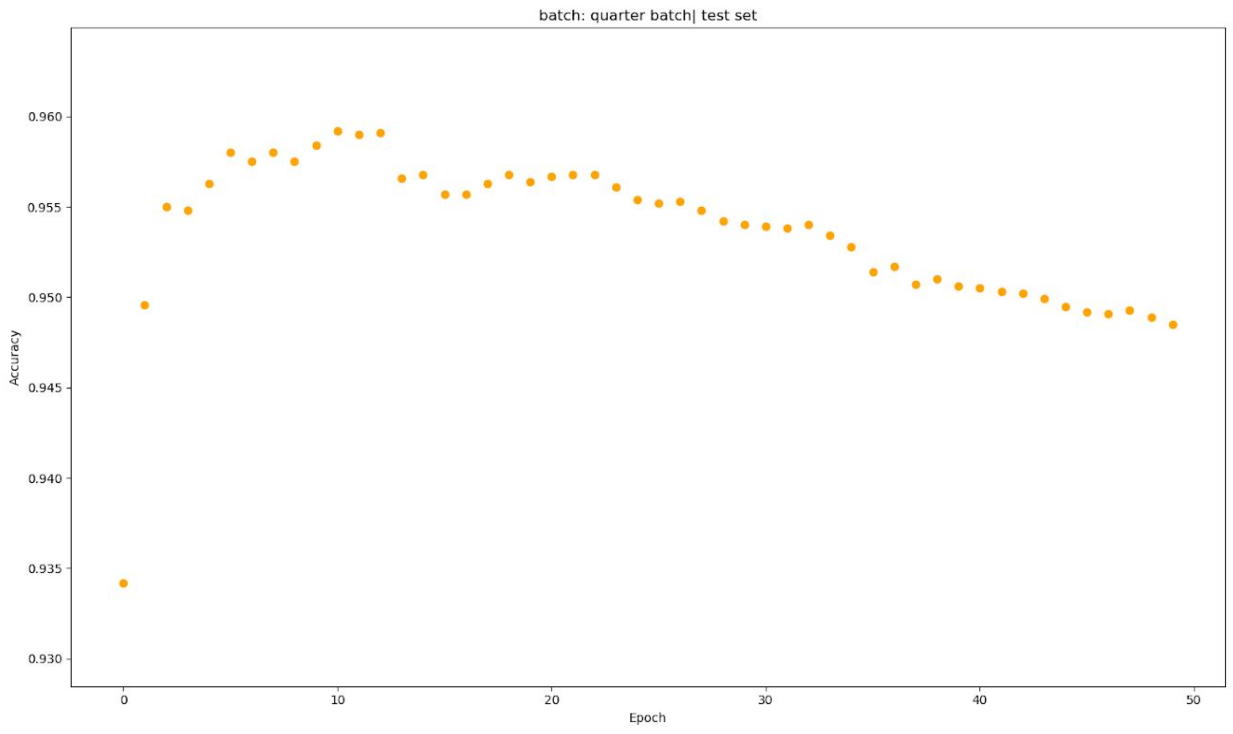


Again, is there evidence that any of your networks has overfit to the training data? If so, what is that evidence?

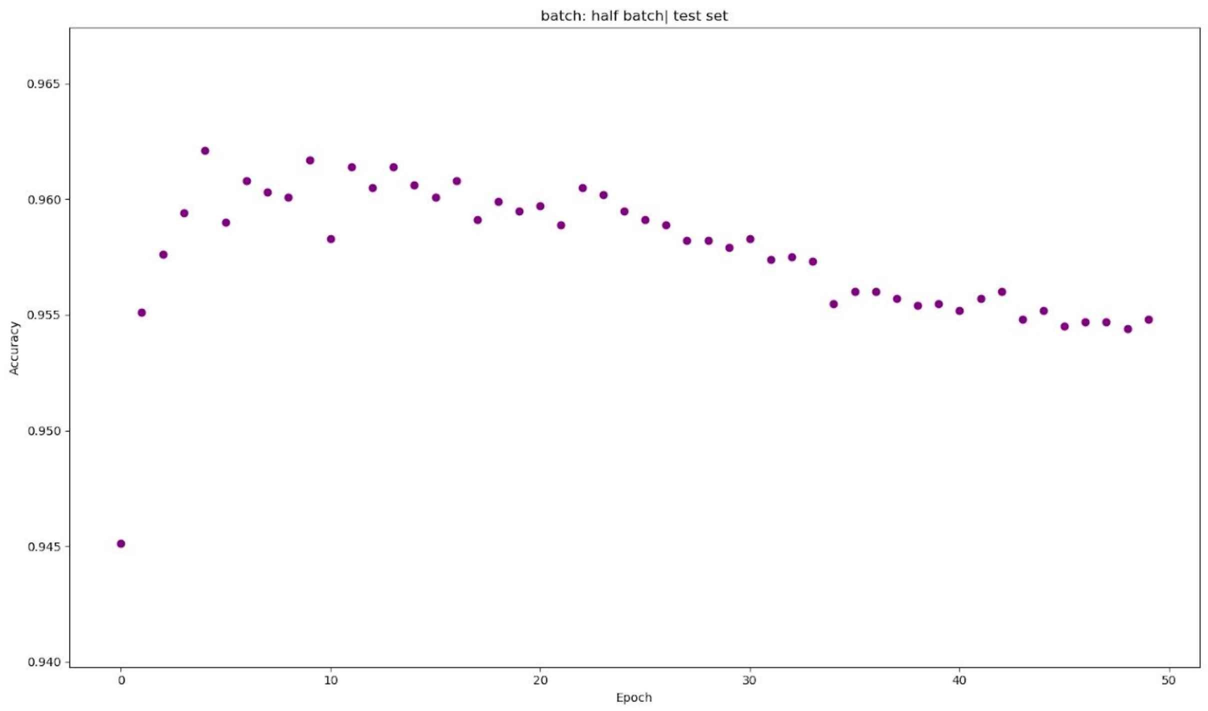
There is evidence of overfitting occurring for momentum = 0.5, as the accuracy on the test set begins to get worse over time, just as we saw in the first experiment. There is little evidence of overfitting without momentum, as the plots of the train and test sets appear to behave the same with momentum set to zero. A less pronounced effect of overfitting can be seen with momentum set to .25

Experiment Three: Vary Batch Size

How does the size of the training data affect the final accuracy on the test data?

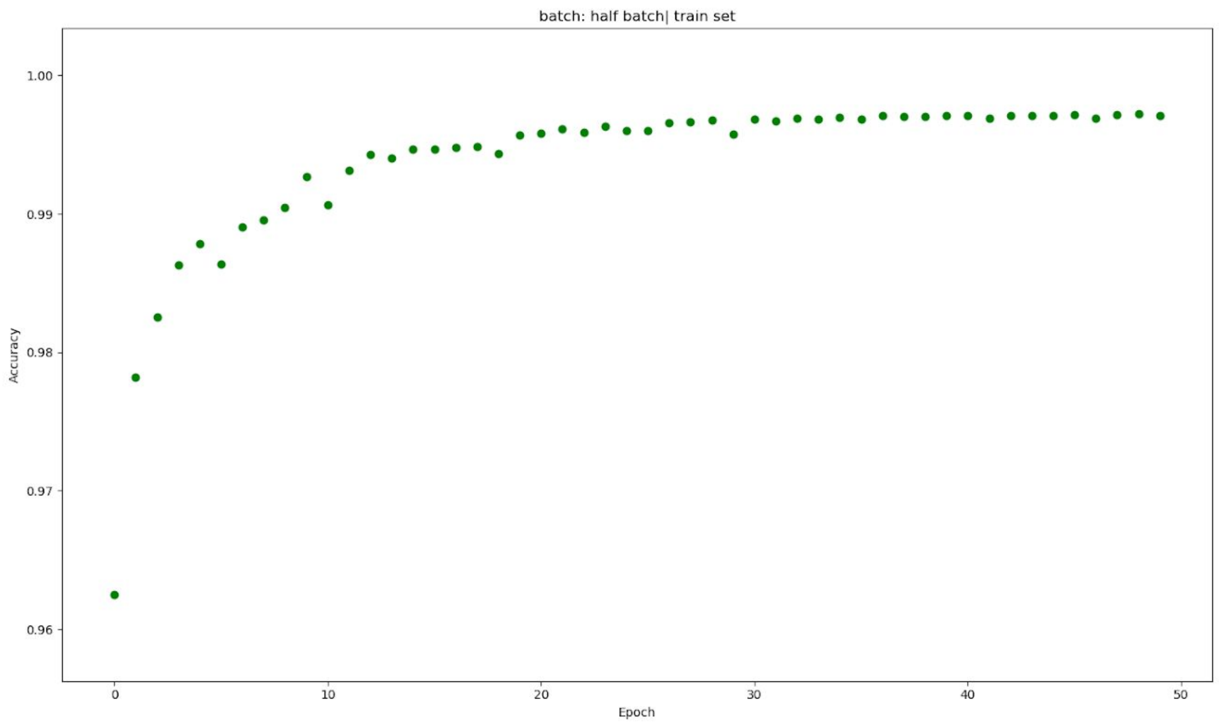


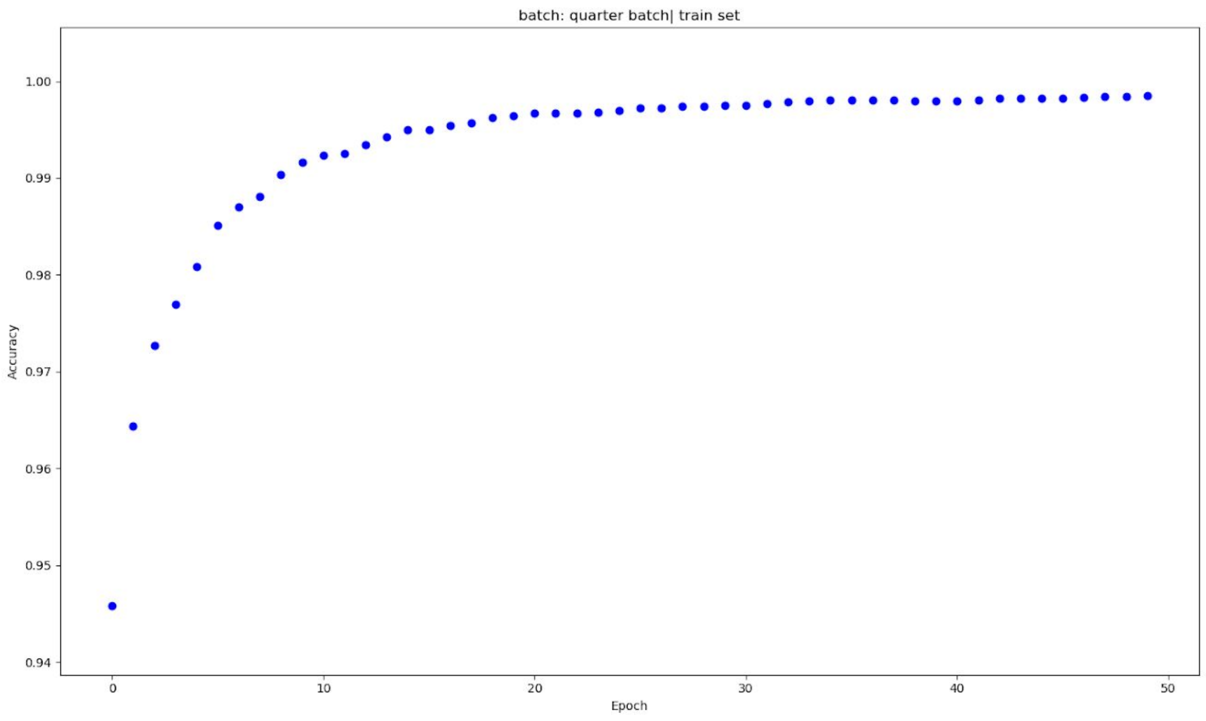
Using only a quarter of the test images, I observed a lower final accuracy than using the full training set, and interestingly, using only half of the test set did not appear to make much of a difference.



How does it affect the number of epochs needed for training to converge?

There is evidence that convergence is earlier for a smaller batch size, which makes sense, because there are fewer images to train on, the network can learn all of the images more quickly.

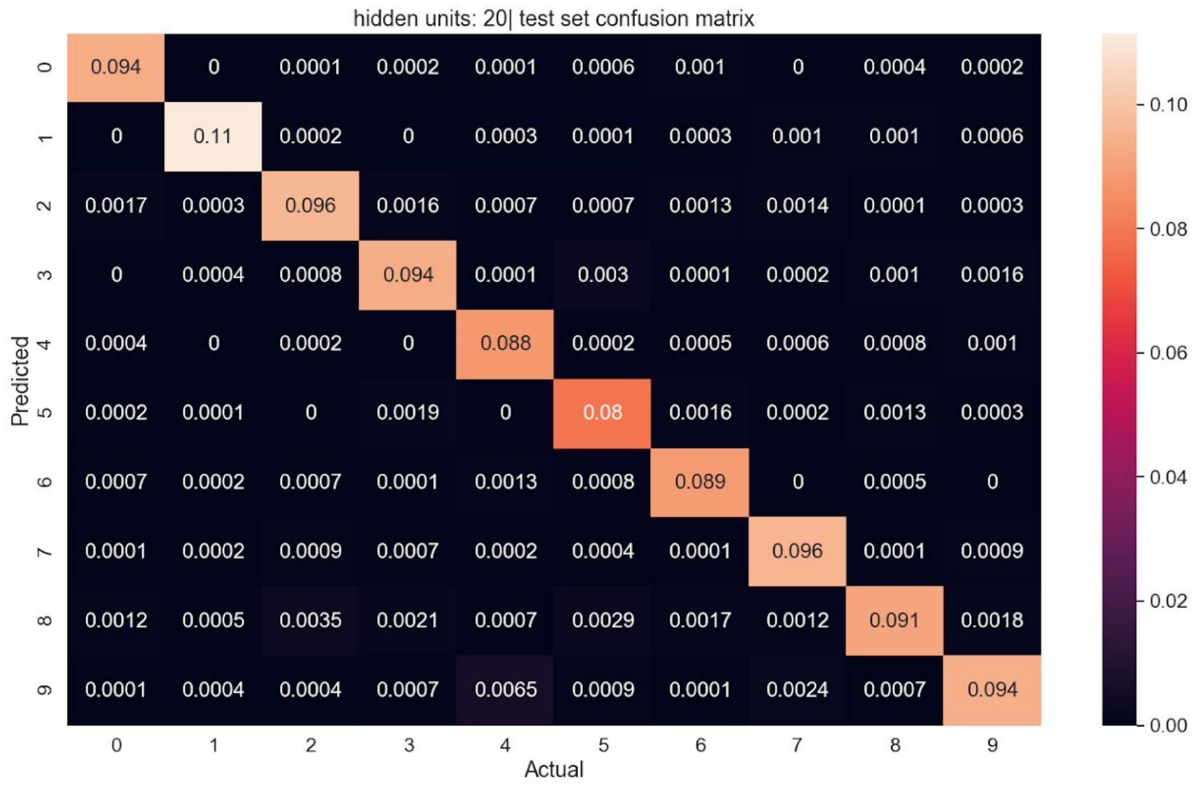




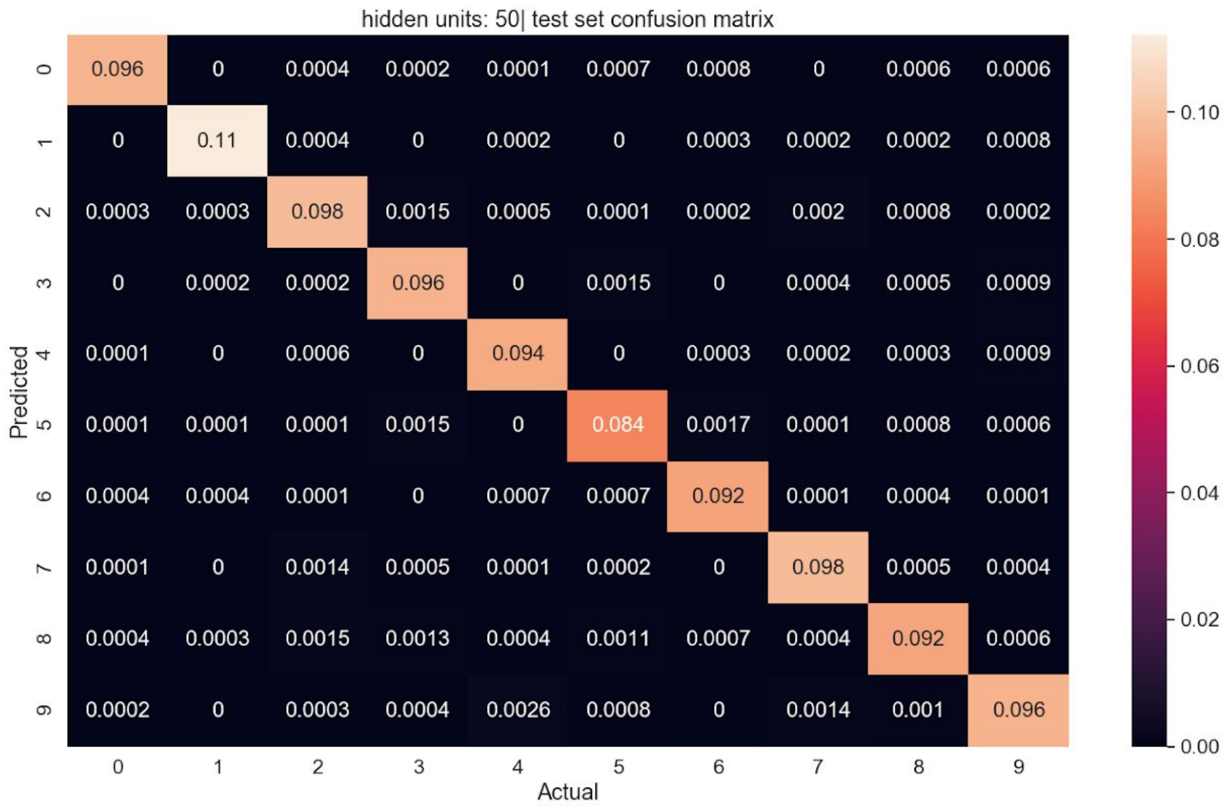
Again, is there evidence that any of your networks has overfit to the training data? If so, what is that evidence?

There is evidence of overfitting in both cases, which also makes sense, since the network is not given enough examples, it overgeneralizes on the training set.

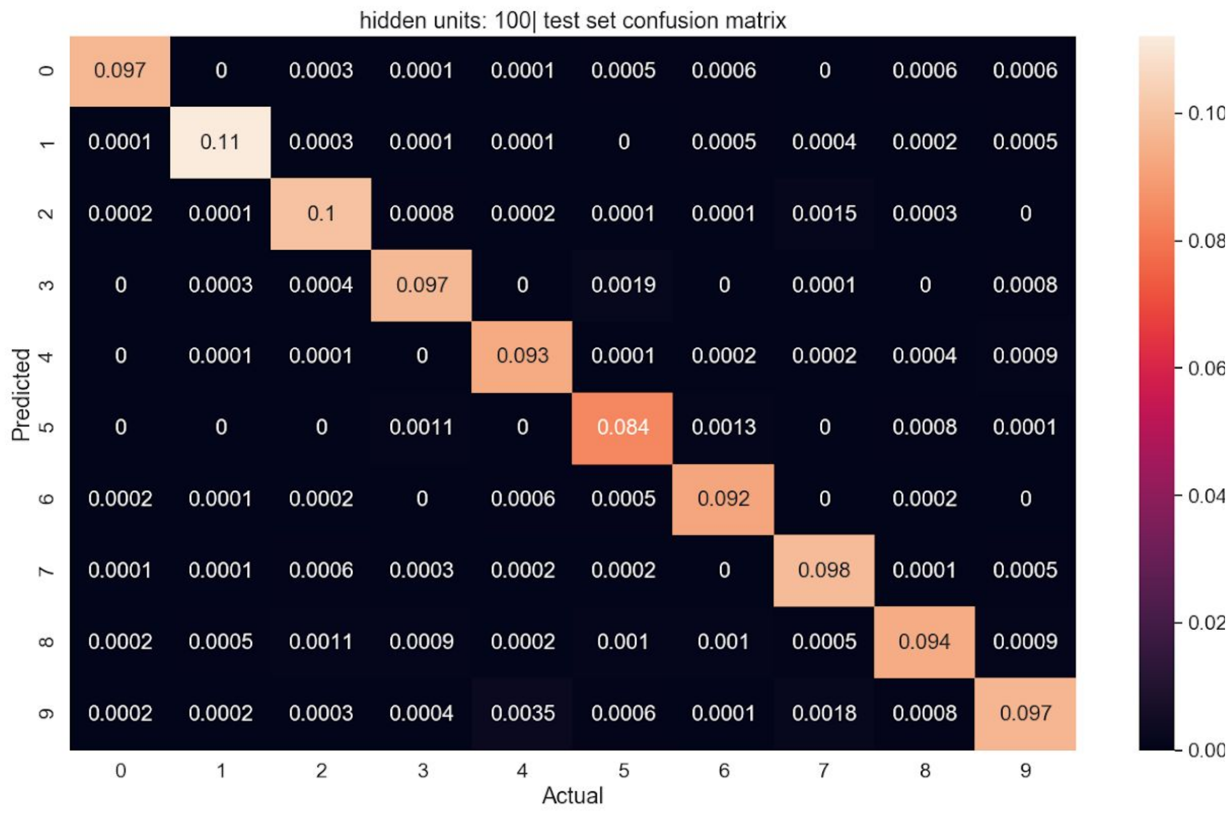
Finally, on the following pages are the final confusion matrices for the tests of all the experiments. Interestingly, the number 5 appears to be the hardest number for a single-hidden layer MLP to recognize, no matter which hyperparameter is variable. Meanwhile, the number 1 appears to be the easiest number for the MLP to recognize. It appears that the number 5 is often confused with the numbers 3 and 6.



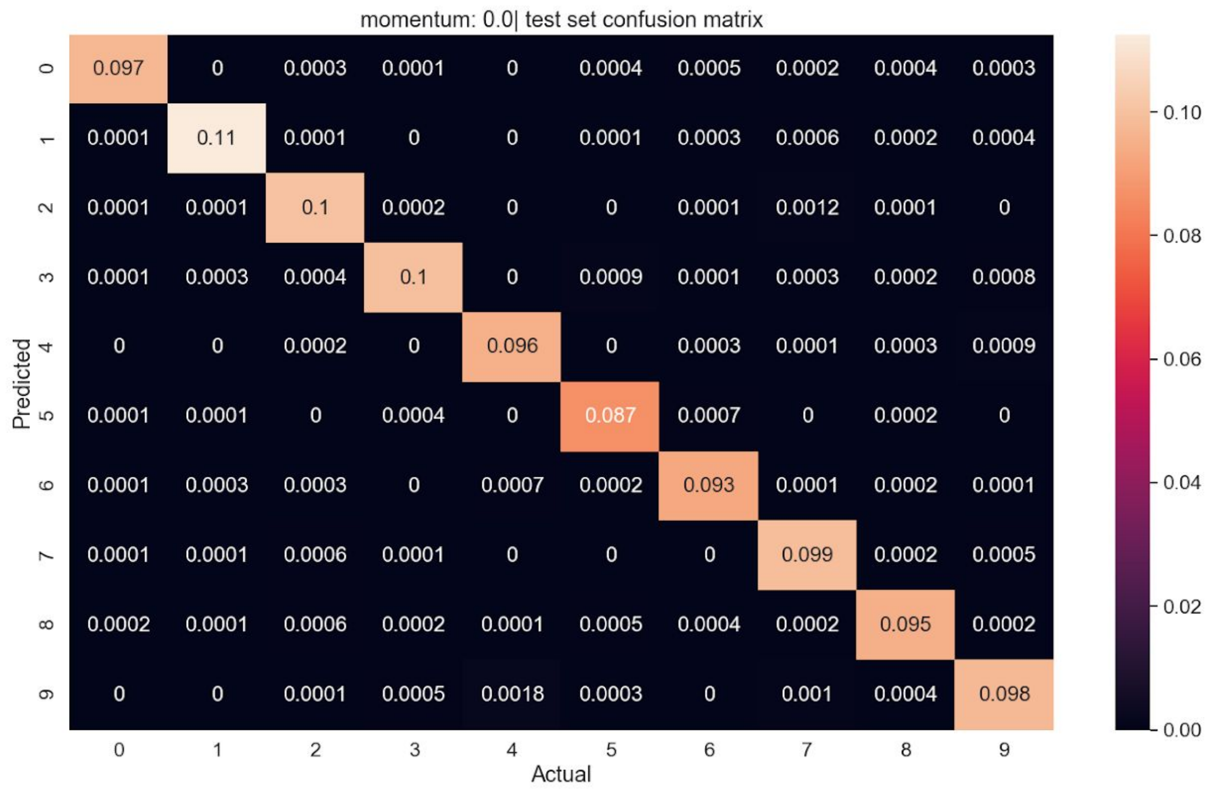
Predicted \ Actual	0	1	2	3	4	5	6	7	8	9
0	0.094	0	0.0001	0.0002	0.0001	0.0006	0.001	0	0.0004	0.0002
1	0	0.11	0.0002	0	0.0003	0.0001	0.0003	0.001	0.001	0.0006
2	0.0017	0.0003	0.096	0.0016	0.0007	0.0007	0.0013	0.0014	0.0001	0.0003
3	0	0.0004	0.0008	0.094	0.0001	0.003	0.0001	0.0002	0.001	0.0016
4	0.0004	0	0.0002	0	0.088	0.0002	0.0005	0.0006	0.0008	0.001
5	0.0002	0.0001	0	0.0019	0	0.08	0.0016	0.0002	0.0013	0.0003
6	0.0007	0.0002	0.0007	0.0001	0.0013	0.0008	0.089	0	0.0005	0
7	0.0001	0.0002	0.0009	0.0007	0.0002	0.0004	0.0001	0.096	0.0001	0.0009
8	0.0012	0.0005	0.0035	0.0021	0.0007	0.0029	0.0017	0.0012	0.091	0.0018
9	0.0001	0.0004	0.0004	0.0007	0.0065	0.0009	0.0001	0.0024	0.0007	0.094



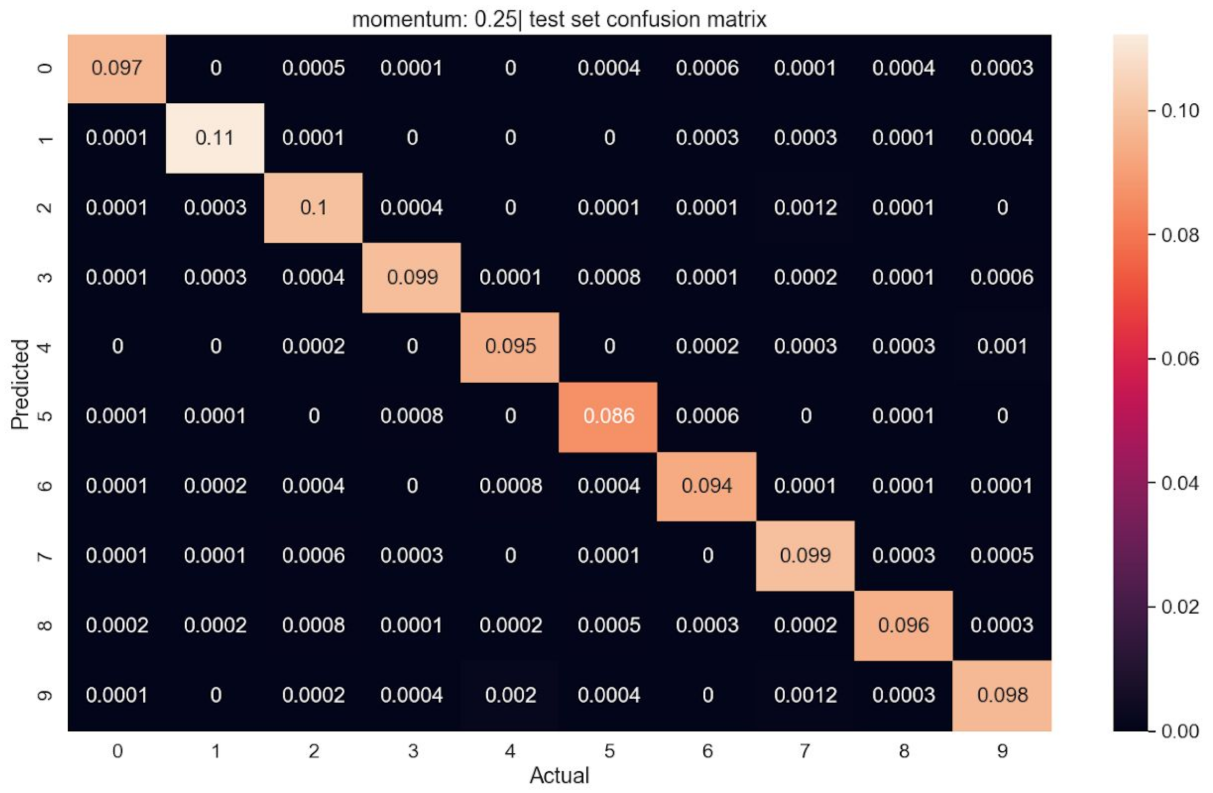
Predicted \ Actual	0	1	2	3	4	5	6	7	8	9
0	0.096	0	0.0004	0.0002	0.0001	0.0007	0.0008	0	0.0006	0.0006
1	0	0.11	0.0004	0	0.0002	0	0.0003	0.0002	0.0002	0.0008
2	0.0003	0.0003	0.098	0.0015	0.0005	0.0001	0.0002	0.002	0.0008	0.0002
3	0	0.0002	0.0002	0.096	0	0.0015	0	0.0004	0.0005	0.0009
4	0.0001	0	0.0006	0	0.094	0	0.0003	0.0002	0.0003	0.0009
5	0.0001	0.0001	0.0001	0.0015	0	0.084	0.0017	0.0001	0.0008	0.0006
6	0.0004	0.0004	0.0001	0	0.0007	0.0007	0.092	0.0001	0.0004	0.0001
7	0.0001	0	0.0014	0.0005	0.0001	0.0002	0	0.098	0.0005	0.0004
8	0.0004	0.0003	0.0015	0.0013	0.0004	0.0011	0.0007	0.0004	0.092	0.0006
9	0.0002	0	0.0003	0.0004	0.0026	0.0008	0	0.0014	0.001	0.096



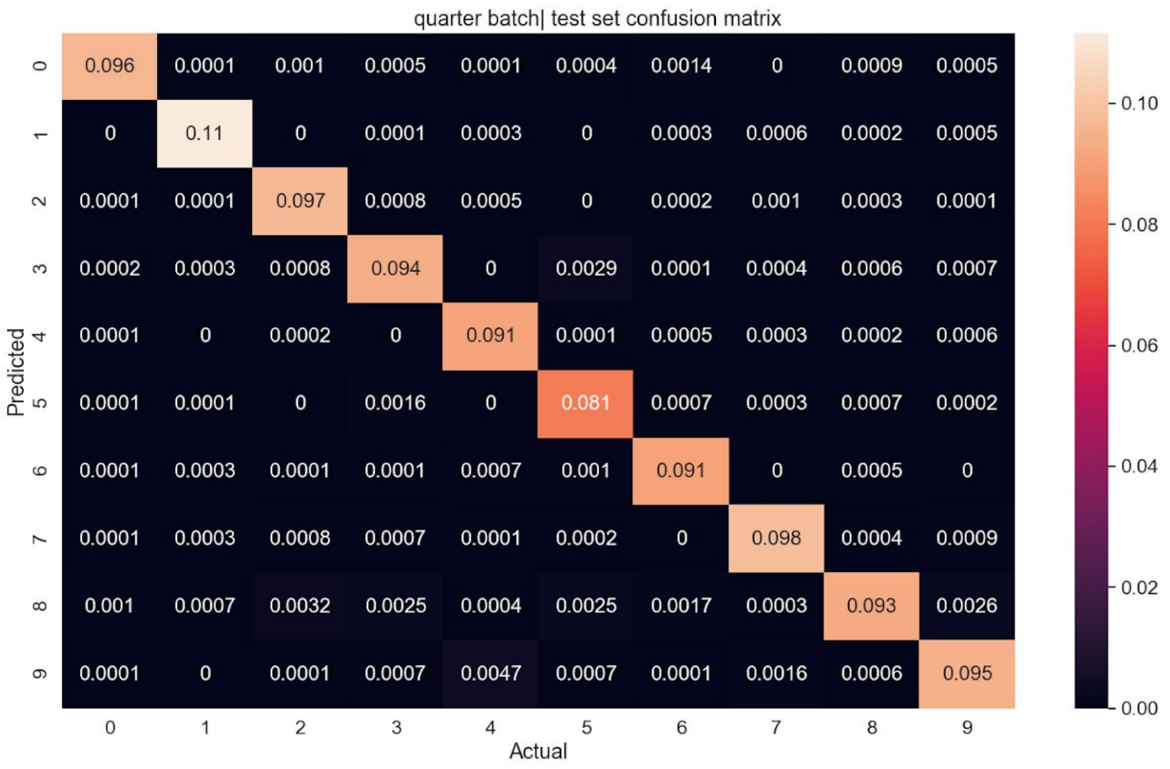
Predicted \ Actual	0	1	2	3	4	5	6	7	8	9
0	0.097	0	0.0003	0.0001	0.0001	0.0005	0.0006	0	0.0006	0.0006
1	0.0001	0.11	0.0003	0.0001	0.0001	0	0.0005	0.0004	0.0002	0.0005
2	0.0002	0.0001	0.1	0.0008	0.0002	0.0001	0.0001	0.0015	0.0003	0
3	0	0.0003	0.0004	0.097	0	0.0019	0	0.0001	0	0.0008
4	0	0.0001	0.0001	0	0.093	0.0001	0.0002	0.0002	0.0004	0.0009
5	0	0	0	0.0011	0	0.084	0.0013	0	0.0008	0.0001
6	0.0002	0.0001	0.0002	0	0.0006	0.0005	0.092	0	0.0002	0
7	0.0001	0.0001	0.0006	0.0003	0.0002	0.0002	0	0.098	0.0001	0.0005
8	0.0002	0.0005	0.0011	0.0009	0.0002	0.001	0.001	0.0005	0.094	0.0009
9	0.0002	0.0002	0.0003	0.0004	0.0035	0.0006	0.0001	0.0018	0.0008	0.097



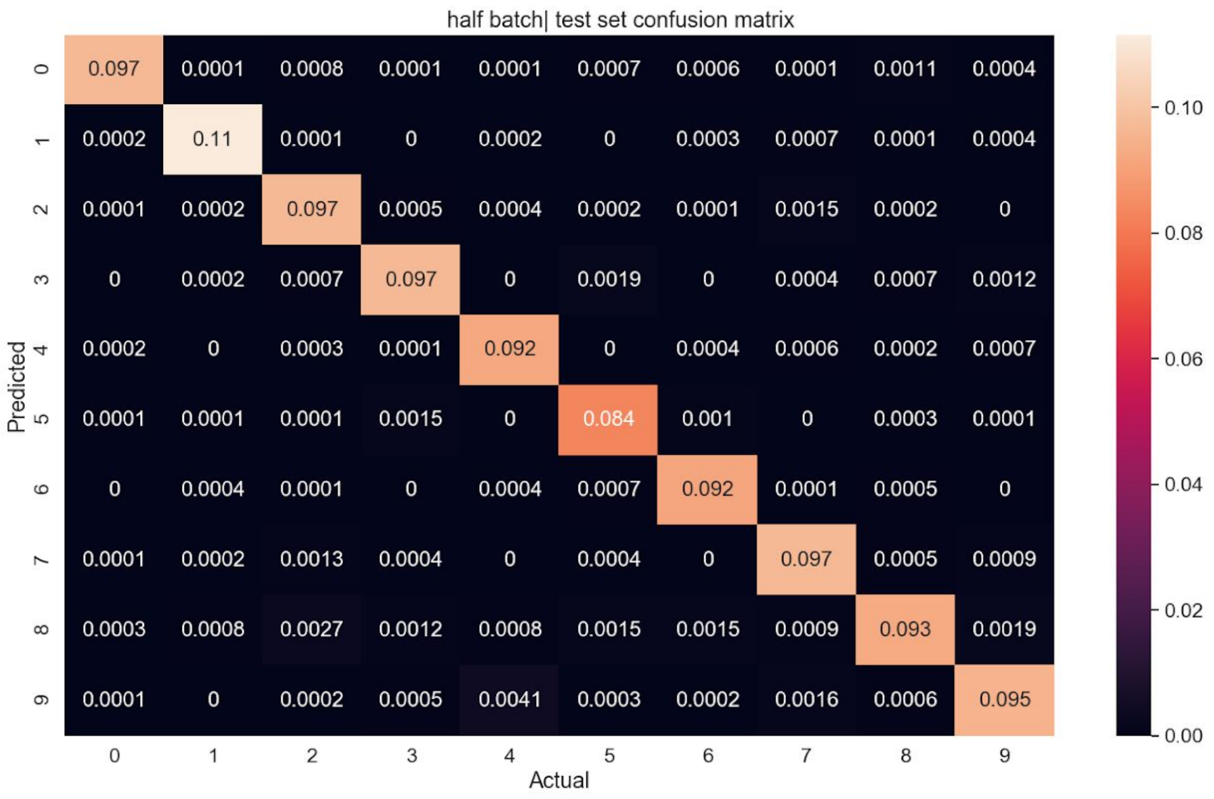
Predicted \ Actual	0	1	2	3	4	5	6	7	8	9
0	0.097	0	0.0003	0.0001	0	0.0004	0.0005	0.0002	0.0004	0.0003
1	0.0001	0.11	0.0001	0	0	0.0001	0.0003	0.0006	0.0002	0.0004
2	0.0001	0.0001	0.1	0.0002	0	0	0.0001	0.0012	0.0001	0
3	0.0001	0.0003	0.0004	0.1	0	0.0009	0.0001	0.0003	0.0002	0.0008
4	0	0	0.0002	0	0.096	0	0.0003	0.0001	0.0003	0.0009
5	0.0001	0.0001	0	0.0004	0	0.087	0.0007	0	0.0002	0
6	0.0001	0.0003	0.0003	0	0.0007	0.0002	0.093	0.0001	0.0002	0.0001
7	0.0001	0.0001	0.0006	0.0001	0	0	0	0.099	0.0002	0.0005
8	0.0002	0.0001	0.0006	0.0002	0.0001	0.0005	0.0004	0.0002	0.095	0.0002
9	0	0	0.0001	0.0005	0.0018	0.0003	0	0.001	0.0004	0.098



Predicted \ Actual	0	1	2	3	4	5	6	7	8	9
0	0.097	0	0.0005	0.0001	0	0.0004	0.0006	0.0001	0.0004	0.0003
1	0.0001	0.11	0.0001	0	0	0	0.0003	0.0003	0.0001	0.0004
2	0.0001	0.0003	0.1	0.0004	0	0.0001	0.0001	0.0012	0.0001	0
3	0.0001	0.0003	0.0004	0.099	0.0001	0.0008	0.0001	0.0002	0.0001	0.0006
4	0	0	0.0002	0	0.095	0	0.0002	0.0003	0.0003	0.001
5	0.0001	0.0001	0	0.0008	0	0.086	0.0006	0	0.0001	0
6	0.0001	0.0002	0.0004	0	0.0008	0.0004	0.094	0.0001	0.0001	0.0001
7	0.0001	0.0001	0.0006	0.0003	0	0.0001	0	0.099	0.0003	0.0005
8	0.0002	0.0002	0.0008	0.0001	0.0002	0.0005	0.0003	0.0002	0.096	0.0003
9	0.0001	0	0.0002	0.0004	0.002	0.0004	0	0.0012	0.0003	0.098



Predicted \ Actual	0	1	2	3	4	5	6	7	8	9
0	0.096	0.0001	0.001	0.0005	0.0001	0.0004	0.0014	0	0.0009	0.0005
1	0	0.11	0	0.0001	0.0003	0	0.0003	0.0006	0.0002	0.0005
2	0.0001	0.0001	0.097	0.0008	0.0005	0	0.0002	0.001	0.0003	0.0001
3	0.0002	0.0003	0.0008	0.094	0	0.0029	0.0001	0.0004	0.0006	0.0007
4	0.0001	0	0.0002	0	0.091	0.0001	0.0005	0.0003	0.0002	0.0006
5	0.0001	0.0001	0	0.0016	0	0.081	0.0007	0.0003	0.0007	0.0002
6	0.0001	0.0003	0.0001	0.0001	0.0007	0.001	0.091	0	0.0005	0
7	0.0001	0.0003	0.0008	0.0007	0.0001	0.0002	0	0.098	0.0004	0.0009
8	0.001	0.0007	0.0032	0.0025	0.0004	0.0025	0.0017	0.0003	0.093	0.0026
9	0.0001	0	0.0001	0.0007	0.0047	0.0007	0.0001	0.0016	0.0006	0.095



Predicted \ Actual	0	1	2	3	4	5	6	7	8	9
0	0.097	0.0001	0.0008	0.0001	0.0001	0.0007	0.0006	0.0001	0.0011	0.0004
1	0.0002	0.11	0.0001	0	0.0002	0	0.0003	0.0007	0.0001	0.0004
2	0.0001	0.0002	0.097	0.0005	0.0004	0.0002	0.0001	0.0015	0.0002	0
3	0	0.0002	0.0007	0.097	0	0.0019	0	0.0004	0.0007	0.0012
4	0.0002	0	0.0003	0.0001	0.092	0	0.0004	0.0006	0.0002	0.0007
5	0.0001	0.0001	0.0001	0.0015	0	0.084	0.001	0	0.0003	0.0001
6	0	0.0004	0.0001	0	0.0004	0.0007	0.092	0.0001	0.0005	0
7	0.0001	0.0002	0.0013	0.0004	0	0.0004	0	0.097	0.0005	0.0009
8	0.0003	0.0008	0.0027	0.0012	0.0008	0.0015	0.0015	0.0009	0.093	0.0019
9	0.0001	0	0.0002	0.0005	0.0041	0.0003	0.0002	0.0016	0.0006	0.095