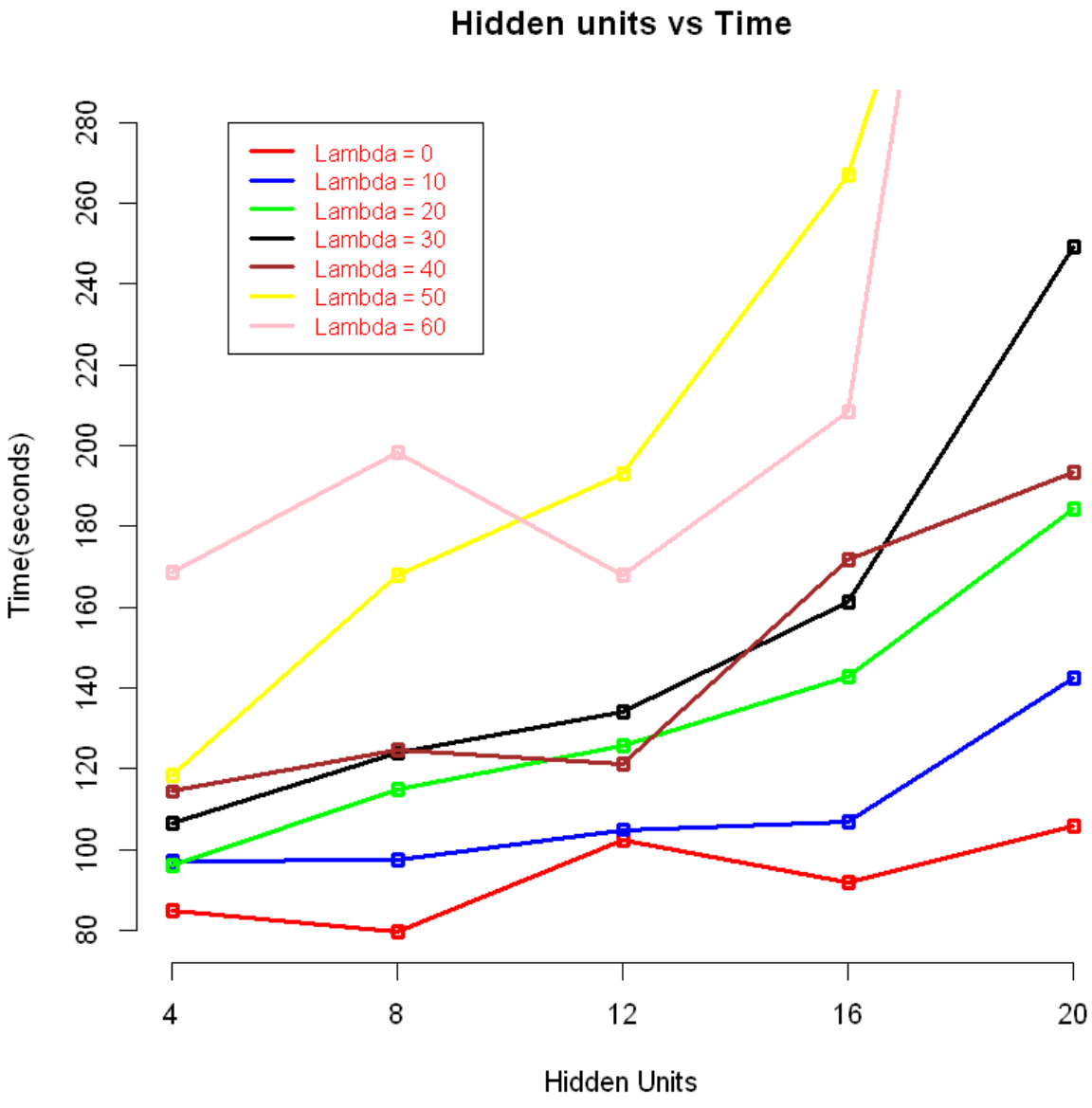
CSE 574 ASSIGNMENT 2 REPORT

SUBMITTED BY SHUBHAM SHAILESH PANDEY

UBID – spandey4, GROUP NO. – 15

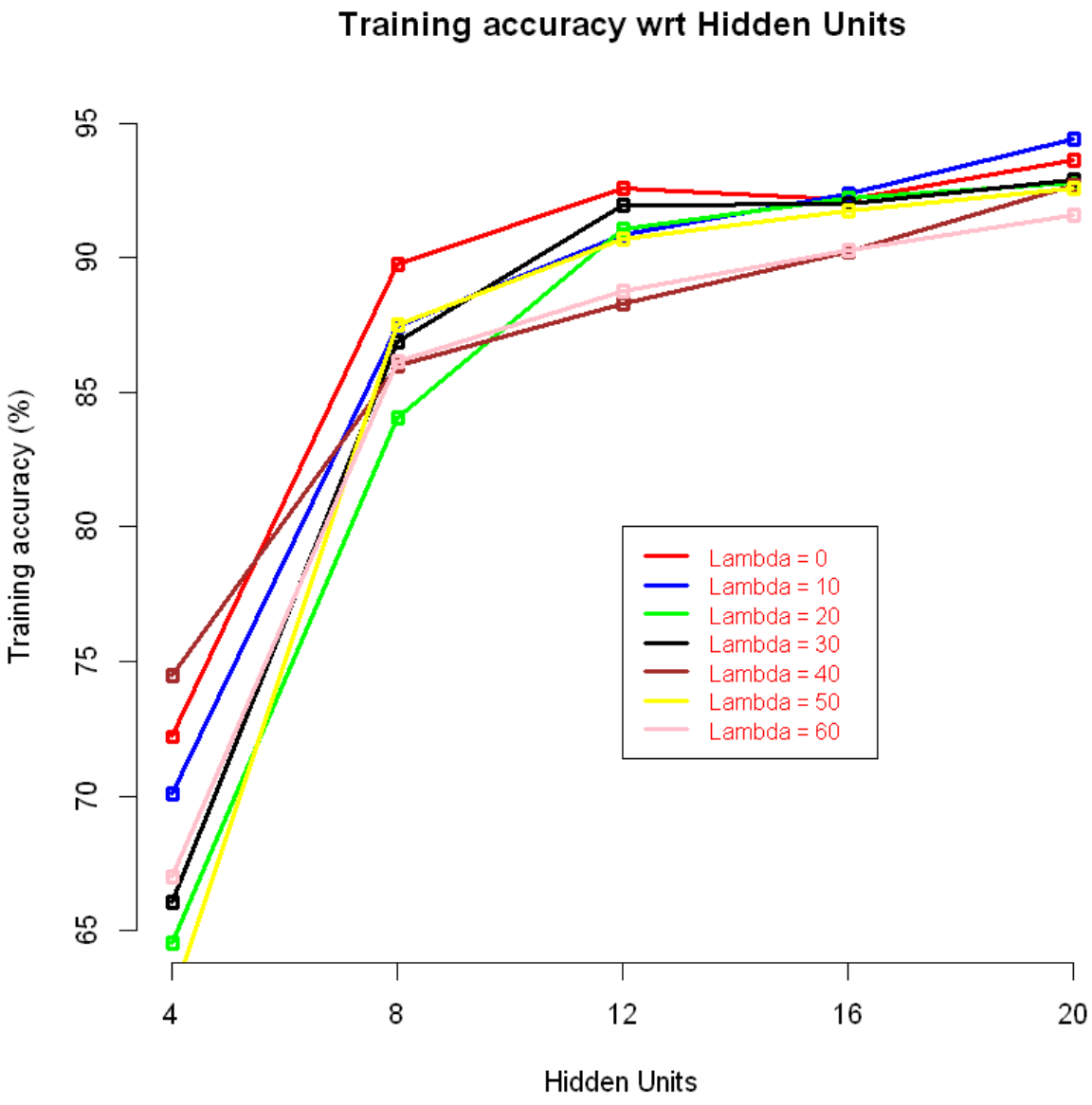
Lambda	Hidden units	Training accuracy(%)	Validation accuracy(%)	Testing accuracy(%)	Time taken(in seconds)
0	4	72.204	70.48	71.73	84.7721
0	8	89.768	89.11	89.45	79.70168
0	12	92.566	91.83	92.08	102.544971
0	16	92.152	91.68	91.56	92.079577
0	20	93.624	93.04	93.12	105.9734
10	4	70.094	68.37	69.72	97.23771
10	8	87.44	86.78	87.48	97.6249
10	12	90.87	89.93	90.53	104.9998
10	16	92.364	91.38	91.97	106.7887
10	20	**94.408**	**93.23**	**93.61**	142.375786
20	4	64.528	62.88	63.68	96.25
20	8	84.052	83.11	84.57	114.86236
20	12	91.076	90.58	90.75	125.9076
20	16	92.224	92.21	92	142.8119
20	20	92.796	91.97	92.45	184.3491
30	4	66.034	65.1	65.19	106.6457
30	8	86.906	86.39	86.74	124.1426
30	12	91.942	91.32	91.77	134.2075
30	16	92.014	91.68	92.12	161.2634
30	20	92.912	92.85	92.75	249.316
40	4	74.476	72.96	74.59	114.4216
40	8	85.968	85.68	85.97	124.661
40	12	88.31	87.6	88.21	121.192
40	16	90.24	89.55	90.3	171.7754
40	20	92.708	92.01	92.56	193.3521
50	4	62.404	61.63	61.49	118.3711
50	8	87.516	87.16	88.09	168.095
50	12	90.698	90.31	90.35	192.9392
50	16	91.738	91.22	91.89	267.0158
50	20	92.568	92.48	92.55	441.21
60	4	66.979	65.65	66.15	168.659
60	8	86.16	84.72	85.58	198.2578
60	12	88.74	88.71	89.26	167.851
60	16	90.272	89.7	90.49	208.4229
60	20	91.57	91.14	91.79	561.5397

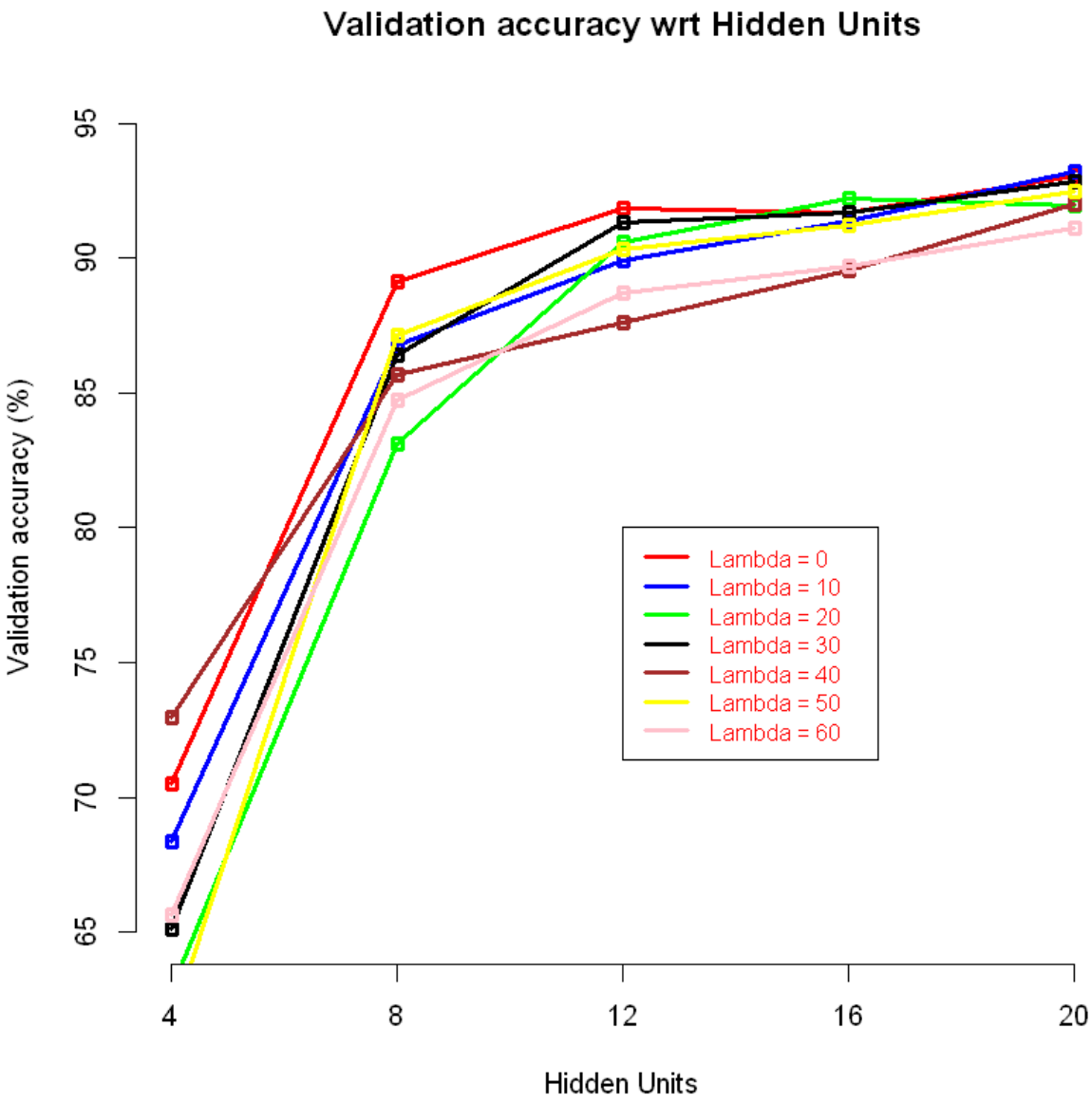
The table above shows the different hyper-parameters tuned for getting accurate training, validation and test results. We tuned the model for lambda ranging from 0-60 and hidden units ranging from 4 to 20. From the table, we see that the highest accuracy for all 3 datasets was obtained when we chose lambda = 10 and 20 units for the hidden layer of the neural network. Hence, they are the optimal paramaters. Also, there isn't much difference in the accuracy performance for various lambdas as long as we keep 20 hidden units. So, keeping low lambda values with 20 hidden units gives a good balance between performance and time for completion.

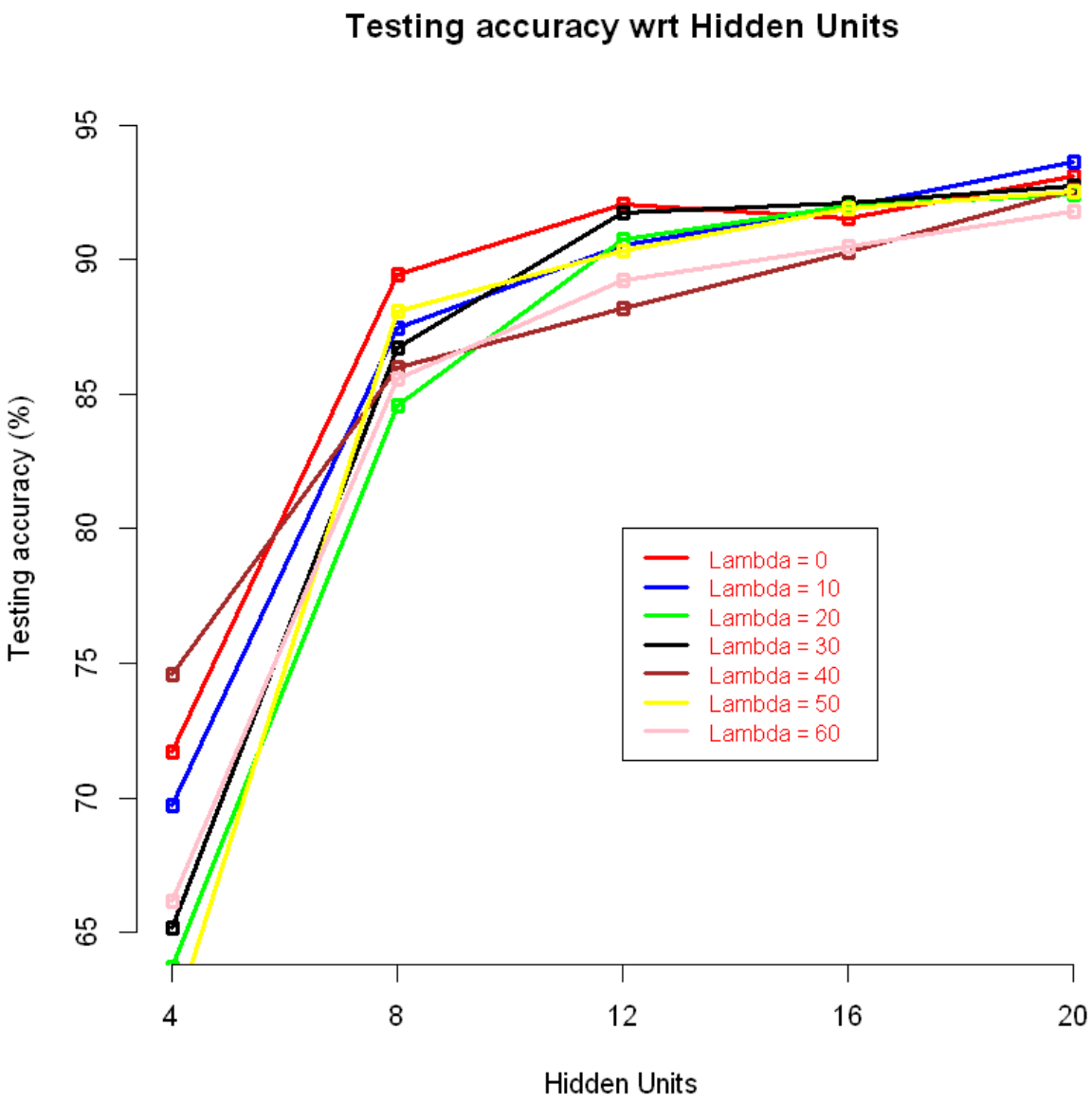


This plot shows the time taken based on the number of units in the hidden layer. It's mostly linear w.r.t the number of units with some abnormalities. This is because higher number of units relates to more computations of weights and gradients.

The following graphs show the relation of training accuracy, validation accuracy and testing accuracy w.r.t hidden units for various values of lambdas. All perform well at high number of hidden inputs.




Validation accuracy wrt Hidden Units
Validation accuracy (%)
65
70
75
80
85
90
95
4
8
12
16
20
Hidden Units
Lambda = 0
Lambda = 10
Lambda = 20
Lambda = 30
Lambda = 40
Lambda = 50
Lambda = 60



Comparison between the results of deep neural network and neural network with one hidden layer on the CelebA data set

- facennScript.py (One Hidden layer) gave these results -
    1. Training set Accuracy:83.91469194312796%
    2. Validation set Accuracy:82.4015009380863%
    3. Test set Accuracy:84.10295230885693%
    4. Time taken (in seconds): 146.52489
- deepnnScript.py (Deep Neural Network) gave these results –

  3 layers => Accuracy: 0.79523087, Time taken (in seconds): 188.519144

  5 layers => Accuracy: 0.7702498, Time taken (in seconds): 199.149211

  7 layers => Accuracy: 0.76003027, Time taken (in seconds): 225.961324

We see that a neural network with one hidden layer outperforms all iterations of deep neural network for any number of layers (3,5 or 7) in terms of both accuracy and time. I suspect overfitting the dataset is causing this to happen.

cnnScript.py Accuracy Report

Size of:

- Training-set: 55000

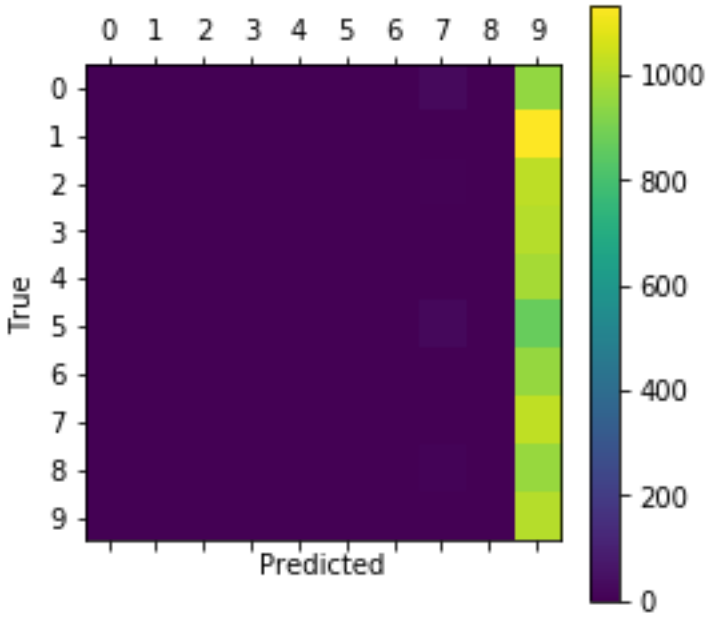
- Test-set: 10000

- Validation-set: 5000

Accuracy on Test-Set: 10.1% (1008 / 10000)

Confusion Matrix:

```
[[  0   0   0   0   0   0   0  27   0 953]
 [  0   0   0   0   0   0   0   0   0 1135]
 [  0   0   0   0   0   0   0   8   0 1024]
 [  0   0   0   0   0   0   0   0   0 1010]
 [  0   0   0   0   0   0   0   2   0  980]
 [  0   0   0   0   0   0   0  23   0  869]
 [  0   0   0   0   0   0   0   3   0  955]
 [  0   0   0   0   0   0   0   0   0 1028]
 [  0   0   0   0   0   0   0  12   0  962]
 [  0   0   0   0   0   0   0   1   0 1008]]
```



Optimization Iteration: 1, Training Accuracy: 18.8%

Time usage: 0:00:00

Accuracy on Test-Set: 10.1% (1007 / 10000)

Confusion Matrix:

[[0   0   0   0   0   0   0 49   0 931]

 [0   0   0   0   0   0   0   0   0 1135]

 [0   0   0   0   0   0   0   9   0 1023]

 [0   0   0   0   0   0   0   1   0 1009]

 [0   0   0   0   0   0   0   2   0 980]

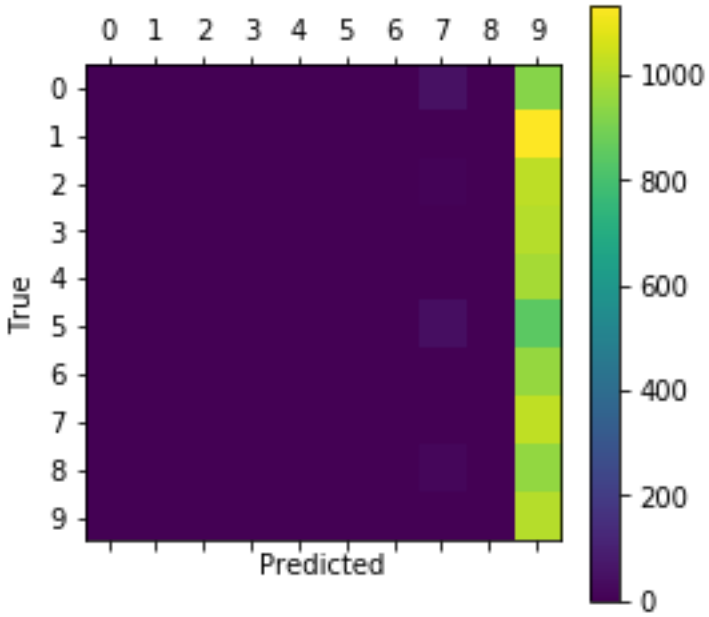
 [0   0   0   0   0   0   0 43   0 849]

 [0   0   0   0   0   0   0   4   0 954]

 [0   0   0   0   0   0   0   0   0 1028]

 [0   0   0   0   0   0   0 22   0 952]

 [0   0   0   0   0   0   0   2   0 1007]]



Time usage: 0:00:05

Accuracy on Test-Set: 66.1% (6610 / 10000)

Confusion Matrix:

[[958 6 0 0 0 0 8 1 7 0]

 [0 1109 10 5 0 0 7 0 4 0]

 [81 82 582 60 1 0 157 22 47 0]

 [56 59 32 751 0 0 7 35 69 1]

 [37 82 8 0 531 0 145 48 20 111]

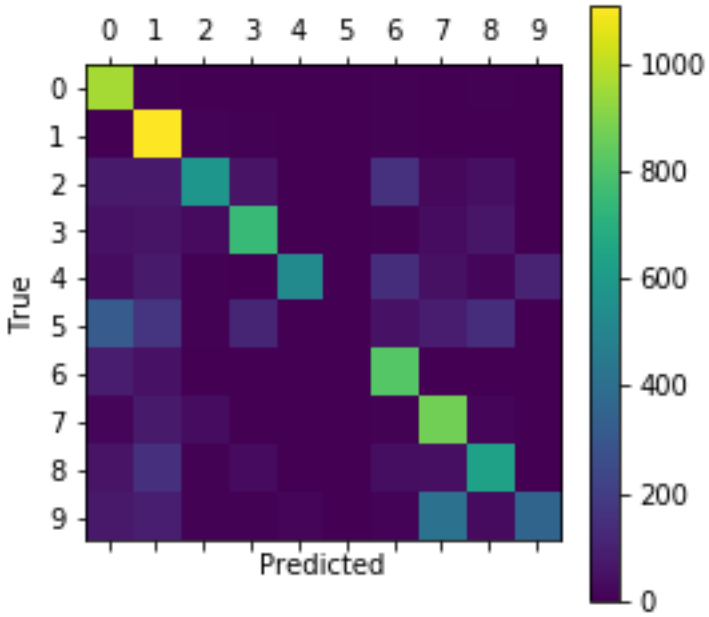
 [317 166 8 114 1 0 53 83 147 3]

 [84 53 0 0 1 0 818 1 1 0]

 [20 78 37 0 0 0 5 871 14 3]

 [60 154 8 31 0 0 41 45 633 2]

 [75 92 5 5 16 0 9 418 32 357]]



Optimization Iteration: 101, Training Accuracy: 65.6%

Optimization Iteration: 201, Training Accuracy: 84.4%

Optimization Iteration: 301, Training Accuracy: 87.5%

Optimization Iteration: 401, Training Accuracy: 81.2%

Optimization Iteration: 501, Training Accuracy: 87.5%

Optimization Iteration: 601, Training Accuracy: 95.3%

Optimization Iteration: 701, Training Accuracy: 89.1%

Optimization Iteration: 801, Training Accuracy: 92.2%

Optimization Iteration: 901, Training Accuracy: 90.6%

Time usage: 0:00:45

Accuracy on Test-Set: 93.0% (9299 / 10000)

Confusion Matrix:

[[959 0 2 1 0 7 6 2 3 0]

 [0 1114 3 3 1 2 3 0 9 0]

 [9 2 947 14 12 4 8 19 15 2]

 [1 2 18 929 0 21 0 18 15 6]

 [1 3 5 0 890 1 15 3 3 61]

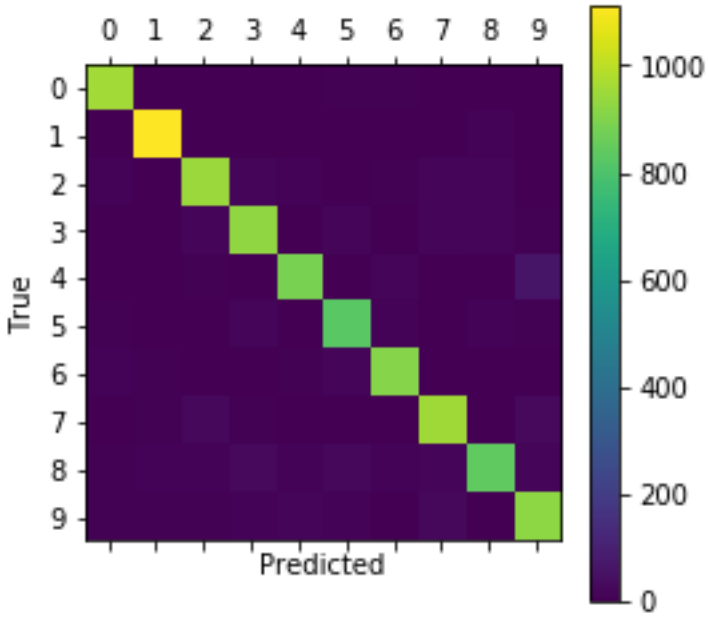
 [6 3 4 19 4 827 13 2 9 5]

 [12 5 3 0 7 19 910 1 1 0]

 [0 7 25 6 4 0 0 957 1 28]

 [6 9 11 30 10 26 10 17 841 14]

 [7 5 6 10 15 12 0 25 4 925]]



Optimization Iteration: 1001, Training Accuracy: 93.8%

Optimization Iteration: 1101, Training Accuracy: 93.8%

Optimization Iteration: 1201, Training Accuracy: 98.4%

Optimization Iteration: 1301, Training Accuracy: 92.2%

Optimization Iteration: 1401, Training Accuracy: 87.5%

Optimization Iteration: 1501, Training Accuracy: 92.2%

Optimization Iteration: 1601, Training Accuracy: 96.9%

Optimization Iteration: 1701, Training Accuracy: 92.2%

Optimization Iteration: 1801, Training Accuracy: 92.2%

Optimization Iteration: 1901, Training Accuracy: 95.3%

Optimization Iteration: 2001, Training Accuracy: 96.9%

Optimization Iteration: 2101, Training Accuracy: 93.8%

Optimization Iteration: 2201, Training Accuracy: 96.9%

Optimization Iteration: 2301, Training Accuracy: 95.3%

Optimization Iteration: 2401, Training Accuracy: 93.8%

Optimization Iteration: 2501, Training Accuracy: 95.3%

Optimization Iteration: 2601, Training Accuracy: 95.3%

Optimization Iteration: 2701, Training Accuracy: 93.8%

Optimization Iteration: 2801, Training Accuracy: 98.4%

Optimization Iteration: 2901, Training Accuracy: 96.9%

Optimization Iteration: 3001, Training Accuracy: 95.3%

Optimization Iteration: 3101, Training Accuracy: 96.9%

Optimization Iteration: 3201, Training Accuracy: 96.9%

Optimization Iteration: 3301, Training Accuracy: 95.3%

Optimization Iteration: 3401, Training Accuracy: 100.0%

Optimization Iteration: 3501, Training Accuracy: 92.2%

Optimization Iteration: 3601, Training Accuracy: 98.4%

Optimization Iteration: 3701, Training Accuracy: 98.4%

Optimization Iteration: 3801, Training Accuracy: 98.4%

Optimization Iteration: 3901, Training Accuracy: 98.4%

Optimization Iteration: 4001, Training Accuracy: 98.4%

Optimization Iteration: 4101, Training Accuracy: 98.4%

Optimization Iteration: 4201, Training Accuracy: 100.0%

Optimization Iteration: 4301, Training Accuracy: 98.4%

Optimization Iteration: 4401, Training Accuracy: 96.9%

Optimization Iteration: 4501, Training Accuracy: 98.4%

Optimization Iteration: 4601, Training Accuracy: 96.9%

Optimization Iteration: 4701, Training Accuracy: 100.0%

Optimization Iteration: 4801, Training Accuracy: 96.9%

Optimization Iteration: 4901, Training Accuracy: 93.8%

Optimization Iteration: 5001, Training Accuracy: 96.9%

Optimization Iteration: 5101, Training Accuracy: 98.4%

Optimization Iteration: 5201, Training Accuracy: 96.9%

Optimization Iteration: 5301, Training Accuracy: 98.4%

Optimization Iteration: 5401, Training Accuracy: 100.0%

Optimization Iteration: 5501, Training Accuracy: 100.0%

Optimization Iteration: 5601, Training Accuracy: 96.9%

Optimization Iteration: 5701, Training Accuracy: 93.8%

Optimization Iteration: 5801, Training Accuracy: 96.9%

Optimization Iteration: 5901, Training Accuracy: 100.0%

Optimization Iteration: 6001, Training Accuracy: 98.4%

Optimization Iteration: 6101, Training Accuracy: 100.0%

Optimization Iteration: 6201, Training Accuracy: 95.3%

Optimization Iteration: 6301, Training Accuracy: 100.0%

Optimization Iteration: 6401, Training Accuracy: 98.4%

Optimization Iteration: 6501, Training Accuracy: 98.4%

Optimization Iteration: 6601, Training Accuracy: 100.0%

Optimization Iteration: 6701, Training Accuracy: 100.0%

Optimization Iteration: 6801, Training Accuracy: 100.0%

Optimization Iteration: 6901, Training Accuracy: 100.0%

Optimization Iteration: 7001, Training Accuracy: 98.4%

Optimization Iteration: 7101, Training Accuracy: 100.0%

Optimization Iteration: 7201, Training Accuracy: 98.4%

Optimization Iteration: 7301, Training Accuracy: 98.4%

Optimization Iteration: 7401, Training Accuracy: 98.4%

Optimization Iteration: 7501, Training Accuracy: 98.4%

Optimization Iteration: 7601, Training Accuracy: 100.0%

Optimization Iteration: 7701, Training Accuracy: 100.0%

Optimization Iteration: 7801, Training Accuracy: 100.0%

Optimization Iteration: 7901, Training Accuracy: 98.4%

Optimization Iteration: 8001, Training Accuracy: 100.0%

Optimization Iteration: 8101, Training Accuracy: 98.4%

Optimization Iteration: 8201, Training Accuracy: 98.4%

Optimization Iteration: 8301, Training Accuracy: 100.0%

Optimization Iteration: 8401, Training Accuracy: 96.9%

Optimization Iteration: 8501, Training Accuracy: 100.0%

Optimization Iteration: 8601, Training Accuracy: 98.4%

Optimization Iteration: 8701, Training Accuracy: 98.4%

Optimization Iteration: 8801, Training Accuracy: 98.4%

Optimization Iteration: 8901, Training Accuracy: 100.0%

Optimization Iteration: 9001, Training Accuracy: 100.0%

Optimization Iteration: 9101, Training Accuracy: 96.9%

Optimization Iteration: 9201, Training Accuracy: 98.4%

Optimization Iteration: 9301, Training Accuracy: 100.0%

Optimization Iteration: 9401, Training Accuracy: 100.0%

Optimization Iteration: 9501, Training Accuracy: 98.4%

Optimization Iteration: 9601, Training Accuracy: 100.0%

Optimization Iteration: 9701, Training Accuracy: 100.0%

Optimization Iteration: 9801, Training Accuracy: 98.4%

Optimization Iteration: 9901, Training Accuracy: 100.0%

Time usage: 0:07:46

Accuracy on Test-Set: 98.7% (9871 / 10000)

Confusion Matrix:

```
[[ 975    0    0    0    0    0    2    1    2    0]
 [   0 1128    2    0    0    0    1    2    2    0]
 [   2    2 1020    0    1    0    0    4    3    0]
 [   0    0    3  998    0    4    0    3    2    0]
 [   0    0    1    0  977    0    1    2    0    1]
 [   2    0    0    3    0  885    1    1    0    0]
 [   4    2    0    0    3    5  944    0    0    0]
 [   1    2    4    1    0    0    0 1017    1    2]
 [   7    1    5    1    2    4    1    3  947    3]
 [   2    5    0    2    9    2    0    8    1  980]]
```