

CSE 574

Assignment 2 Report

Submitted by

Group 76

Paritosh Kumar Velalam (pvelalam) (50295537)

Sampreeth Boddi Reddy (sboddire) (50298591)

Abhinay Reddy Alluri (aalluri) (50288936)

Strategy for choosing number of hidden nodes and lambda:

We can say that the model fits data reasonably when we have low bias (high accuracy) and low variance.

The low bias can be obtained by trying out different neural network architectures. In this project, we increased the number of hidden nodes to increase the accuracy. We have tried out 4,8,12,16,20,50 and 75 number of hidden nodes.

Low variance can be obtained by using regularization which means we are reducing over-fitting on training data.

We first fixed $\lambda=0$ and increased the number of hidden nodes until 95% accuracy is achieved for training data set. Then regularization is used to decrease the variance between training and validation data sets.

The below tables and figures show the accuracy obtained on training, validation and test data sets for different number of hidden nodes and different values of λ to show the performance of neural network on these parameters.

No. of Hidden nodes = 4

Lambda Value	Training Set Accuracy	Validation Set Accuracy	Test Set Accuracy	Training Time
0	65.424	63.54	64.74	46.976
5	64.914	63.149	64.51	44.77615
10	64.368	62.71	64.22	43.61418
15	64.6379	62.97	64.3	41.8852
20	64.039	62.26	63.7399	42.86
25	64.918	63.43	64.51	43.187
30	64.744	63.28	64.37	43.90057
35	63.544	61.87	63.11	41.735
40	64.44	62.92	64.18	42.34
45	64.102	62.3	63.79	41.438
50	63.932	62.17	63.759	41.5857
55	65.256	63.71	64.94	41.549
60	64.524	63	64.33	41.515

Table 1.1

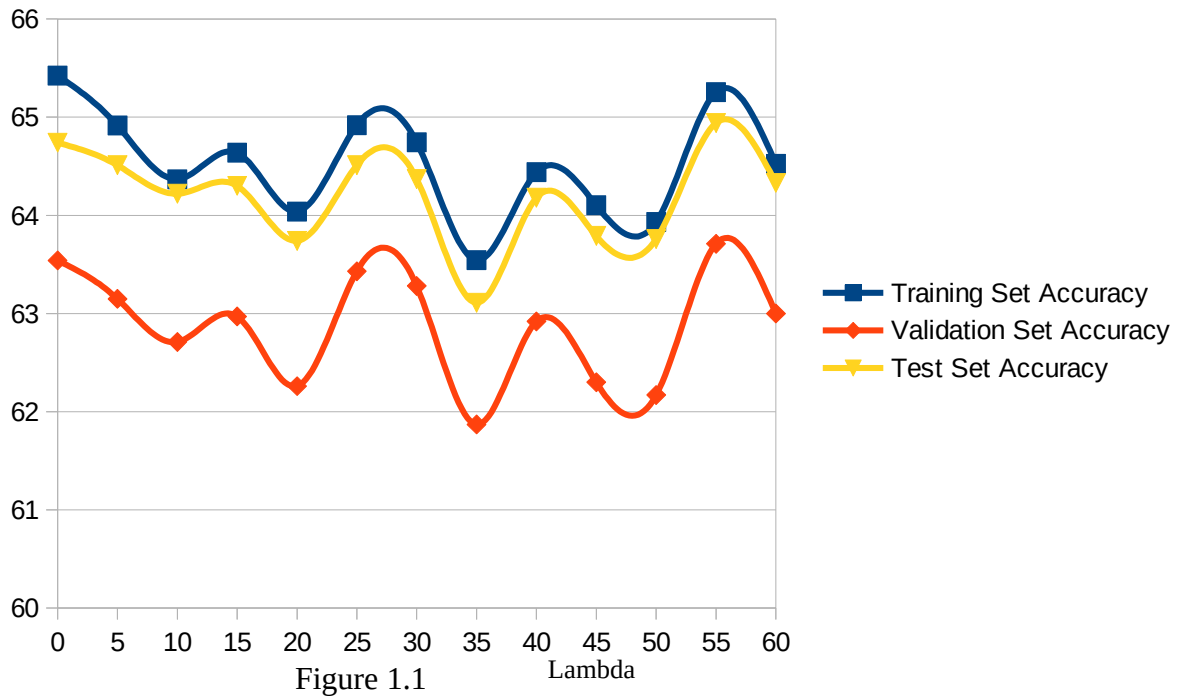


Figure 1.1 shows the accuracy obtained for Training, Validation and Test data sets with number of hidden nodes=4. From the figure we can see that the accuracy is very low for lower number of hidden nodes. We can improve accuracy by increasing the number of hidden nodes.

No. of Hidden nodes = 8

Lambda Value	Training Set Accuracy	Validation Set Accuracy	Test Set Accuracy	Training Time
0	89.88	89.58	89.46	43.419
5	89.932	89.649	89.41	42.5022
10	89.646	89.49	89.48	42.1026
15	90.118	89.71	89.73	42.0993
20	89.908	89.79	89.69	41.6008
25	90.12	89.78	89.8	42.634955
30	89.6819	89.53	89.39	41.6797
35	89.75	89.52	89.53	41.1538
40	89.69	89.67	89.5399	41.6226
45	89.6619	89.37	89.4299	41.7652
50	89.826	89.89	89.59	42.5678
55	89.982	89.87	89.68	42.922
60	89.592	89.32	89.42	41.91

Table 1.2

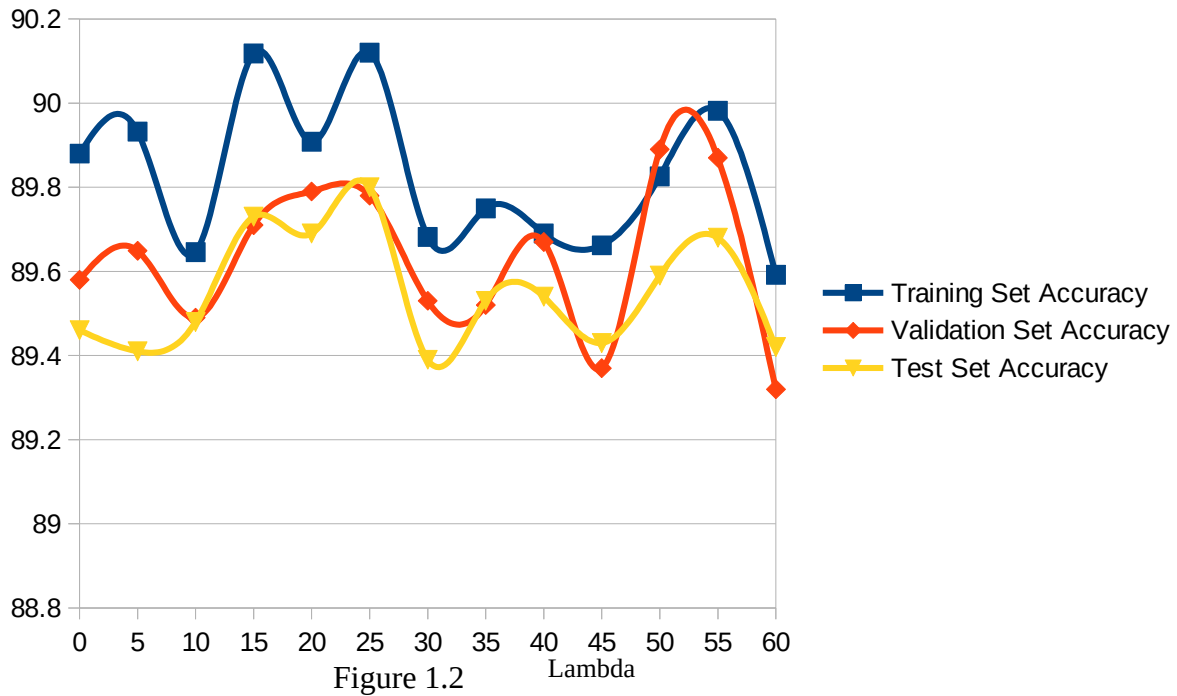


Figure 1.2 shows the accuracy obtained for Training, Validation and Test data sets with number of hidden nodes=8. From the figure we can see that compared to hidden nodes=4, the accuracy has increased. But still the accuracy can be improved further by increasing the number of hidden nodes.

No. of hidden nodes 12

Lambda Value	Training Set Accuracy	Validation Set Accuracy	Test Set Accuracy	Training Time
0	91.908	91.67	91.62	44.993
5	91.948	91.61	91.67	44.521
10	92.066	91.75	91.75	44.3904
15	91.978	91.7	91.93	44.525
20	91.886	91.77	91.759	44.617
25	91.97	91.869	91.79	45.0783
30	91.86	91.69	91.77	44.8619
35	91.916	91.77	91.78	45.44809
40	91.736	91.7	91.67	45.3716
45	91.628	91.7	91.6	45.337
50	91.69	91.62	91.7	45.3867
55	91.526	91.57	91.55	45.443
60	91.416	91.44	91.36	45.403

Table 1.3

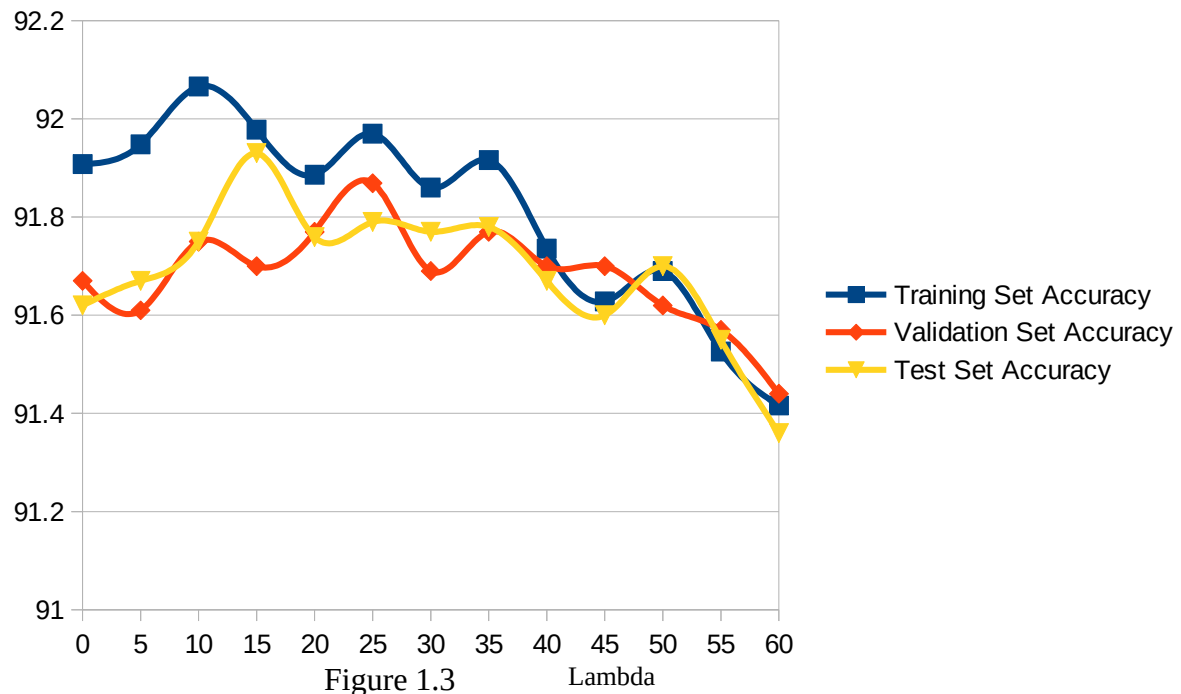


Figure 1.3 shows the accuracy obtained for Training, Validation and Test data sets with number of hidden nodes=12. From the figure we can see that the accuracy improved compared to hidden nodes=8. But still we need to achieve better accuracy. So we can increase the number of hidden nodes. Also as lamda increases, we can see that the variance between the training, validation and test data sets is getting decreased.

No of hidden nodes 16

Lambda Value	Training Set Accuracy	Validation Set Accuracy	Test Set Accuracy	Training Time
0	92.976	92.589	92.679	50.417
5	93.02	92.62	92.73	50.219
10	93.136	92.72	93.03	49.2044
15	93.188	92.89	93.02	49.698
20	93.05	92.89	92.74	48.313
25	92.96	92.77	92.75	48.2438
30	92.878	92.66	92.66	48.4838
35	92.73	92.56	92.589	48.318
40	92.594	92.44	92.51	48.1711
45	92.572	92.44	92.47	48.31
50	92.464	92.39	92.38	48.299
55	92.324	92.21	92.28	48.3574
60	92.194	92.01	92.14	48.39

Table 1.4

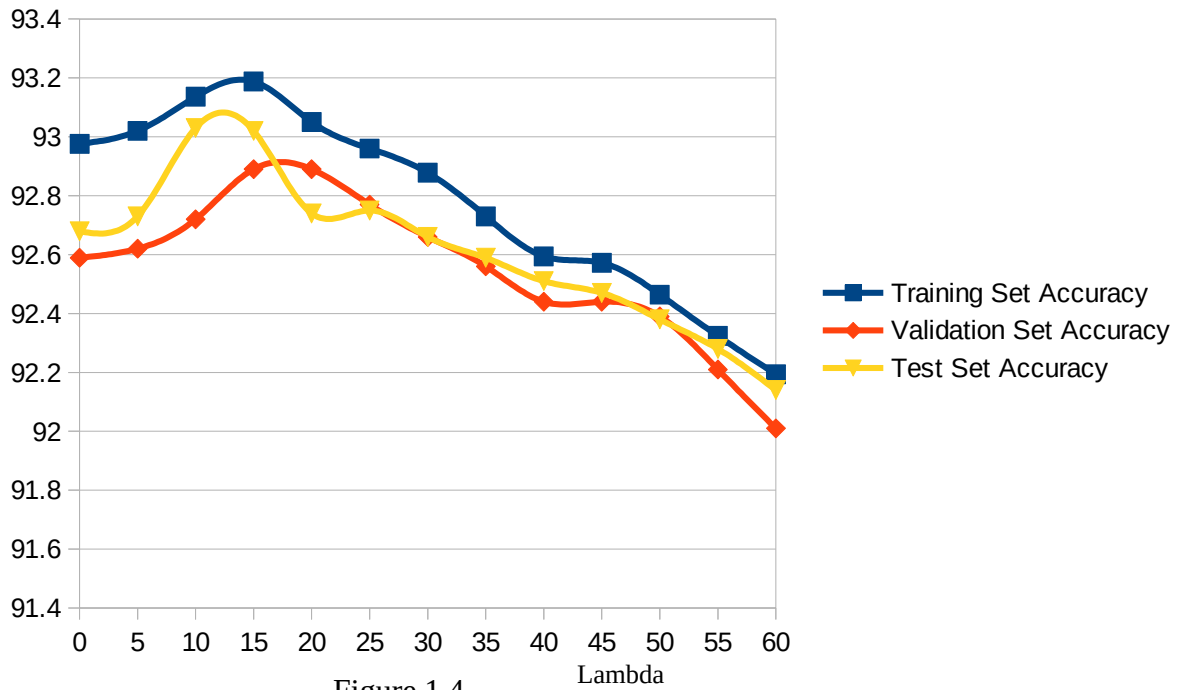


Figure 1.4 shows the accuracy obtained for Training, Validation and Test data sets with number of hidden nodes=16. From the figure we can see that the accuracy improved compared to hidden nodes=12. We can check whether the accuracy can be improved further by increasing the number of hidden nodes.

No of hidden nodes 20

Lambda Value	Training Set Accuracy	Validation Set Accuracy	Test Set Accuracy	Training Time
0	93.48	93.34	93.4	50.742
5	93.684	93.479	93.52	50.306
10	93.632	93.43	93.42	50.253
15	93.692	93.589	93.54	49.793
20	93.612	93.46	93.57	49.88
25	93.586	93.39	93.53	50.094
30	93.512	93.3	93.42	49.823
35	93.376	93.17	93.32	49.797
40	93.28	93.11	93.27	49.9234
45	93.168	92.97	93.26	49.753
50	93.068	92.81	93.179	49.924
55	92.938	92.71	93.04	49.861
60	92.738	92.57	92.87	51.037

Table 1.5

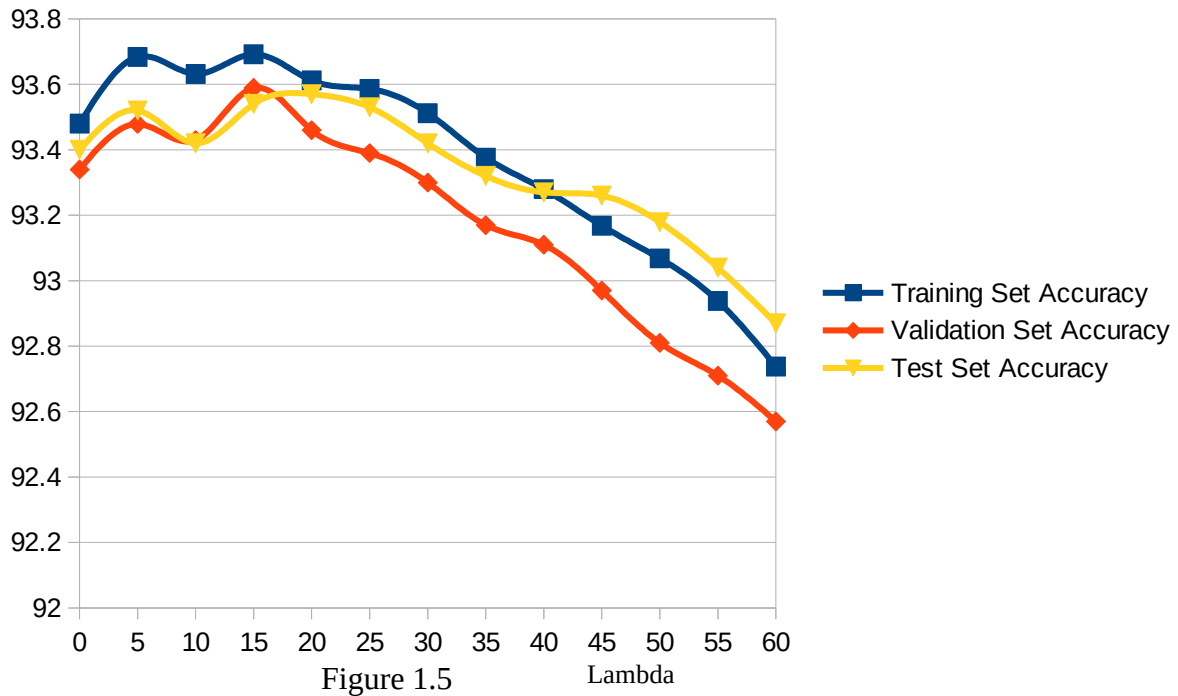


Figure 1.5 shows the accuracy obtained for Training, Validation and Test data sets with number of hidden nodes=20. From the figure we can see that the accuracy improved compared to hidden nodes=16.

No of hidden nodes 50

Lambda Value	Training Set Accuracy	Validation Set Accuracy	Test Set Accuracy	Training Time
0	95.072	95.02	94.89	79.783
5	95.156	95.11	95.12	76.434
10	95.196	95.11	95.03	77.912
15	94.728	94.75	94.789	76.057
20	95.15	94.99	95	74.58
25	94.884	94.87	94.69	74.4612
30	94.612	94.55	94.45	73.3142
35	94.374	94.36	94.33	73.098
40	94.556	94.44	94.43	74.3548
45	94.364	94.179	94.24	74.496
50	94.166	94.05	94.13	74.7339
55	93.976	93.99	94.03	74.549
60	93.978	93.83	93.88	74.46628

Table 1.6

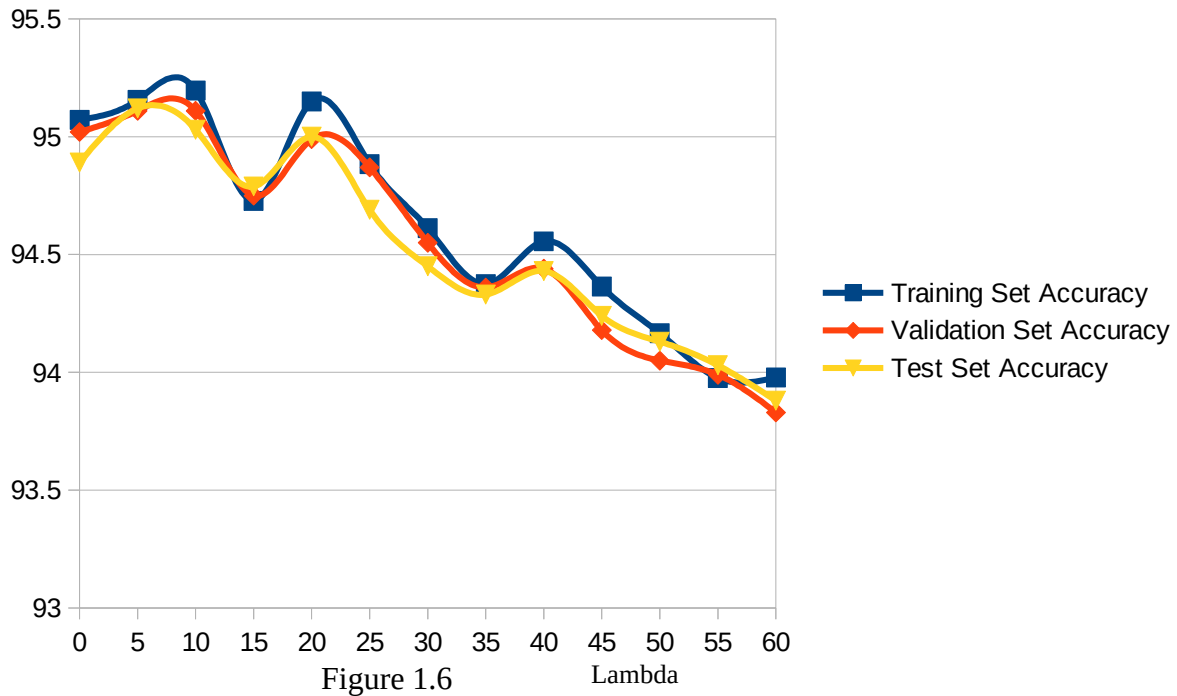


Figure 1.6 shows the accuracy obtained for Training, Validation and Test data sets with number of hidden nodes=50. From the figure we can see that the accuracy improved compared to hidden nodes=20. Also as lambda is getting increased, the variance is getting decreased. But for higher values of lambda, the accuracy is getting decreased. We have to make a trade-off between bias and variance. We can choose lambda=10 as optimal value because the accuracy is high with low variance.

No of hidden nodes 75

Lambda Value	Training Set Accuracy	Validation Set Accuracy	Test Set Accuracy	Training Time
0	95.664	95.54	95.12	95.5229
5	95.806	95.75	95.23	94.394
10	95.192	95.06	94.86	91.935
15	95.2659	95.24	95.009	90.927
20	95.072	95.009	94.91	91.06
25	95.17	95.009	95.05	91.027
30	95.004	94.94	94.91	91.149
35	94.776	94.77	94.71	91.128
40	94.676	94.65	94.76	90.988
45	94.52	94.53	94.68	90.9259
50	93.788	93.77	94.02	91.423
55	93.826	93.85	93.92	91.37
60	93.788	93.74	93.89	91.025

Table 1.7

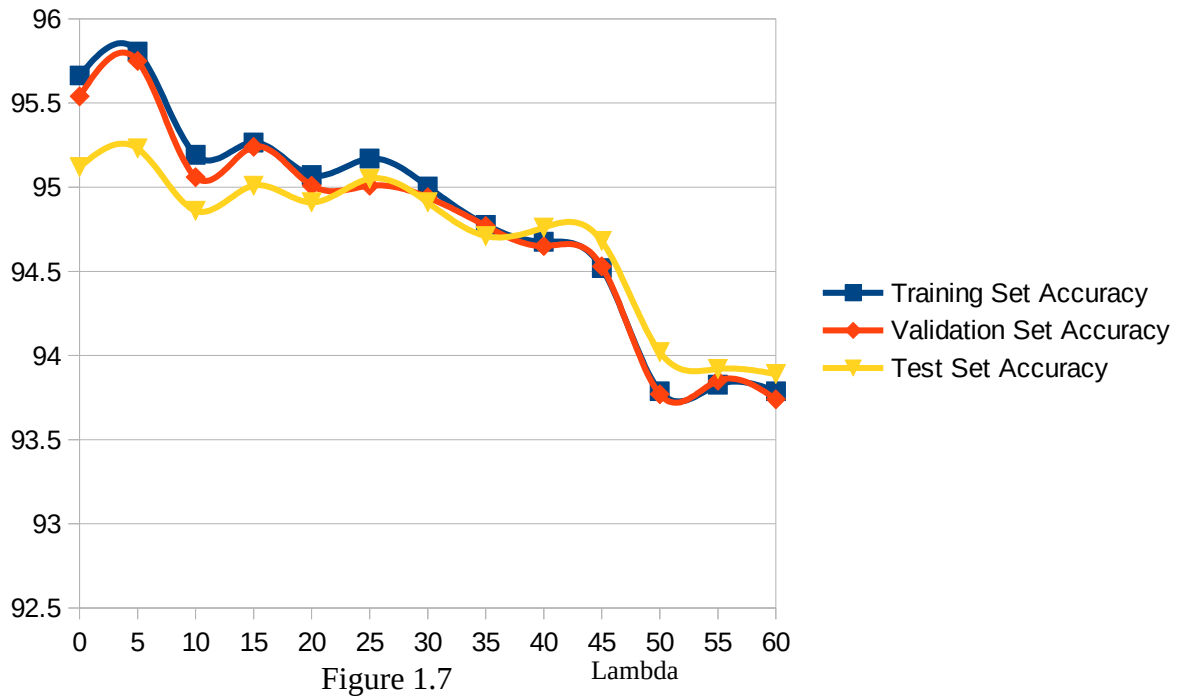


Figure 1.7 shows the accuracy obtained for Training, Validation and Test data sets with number of hidden nodes=75. The accuracy has no significant improvement compared to number of hidden nodes=50 at the cost of running time.

So we are choosing number of hidden nodes=50 and lambda=10 as optimum values.

Number of Hidden Units vs Accuracy and Time:

The below table and plot shows the accuracy of Training, validation and Test data sets for different number of hidden nodes .

No of Hidden Units (Lambda=10)	Training Set Accuracy	Validation Set Accuracy	Test Set Accuracy
4	64.368	62.71	64.22
8	89.646	89.49	89.48
12	92.066	91.75	91.75
16	93.136	92.72	93.03
20	93.632	93.43	93.42
50	95.196	95.11	95.03
75	95.192	95.06	94.86

Table 1.8

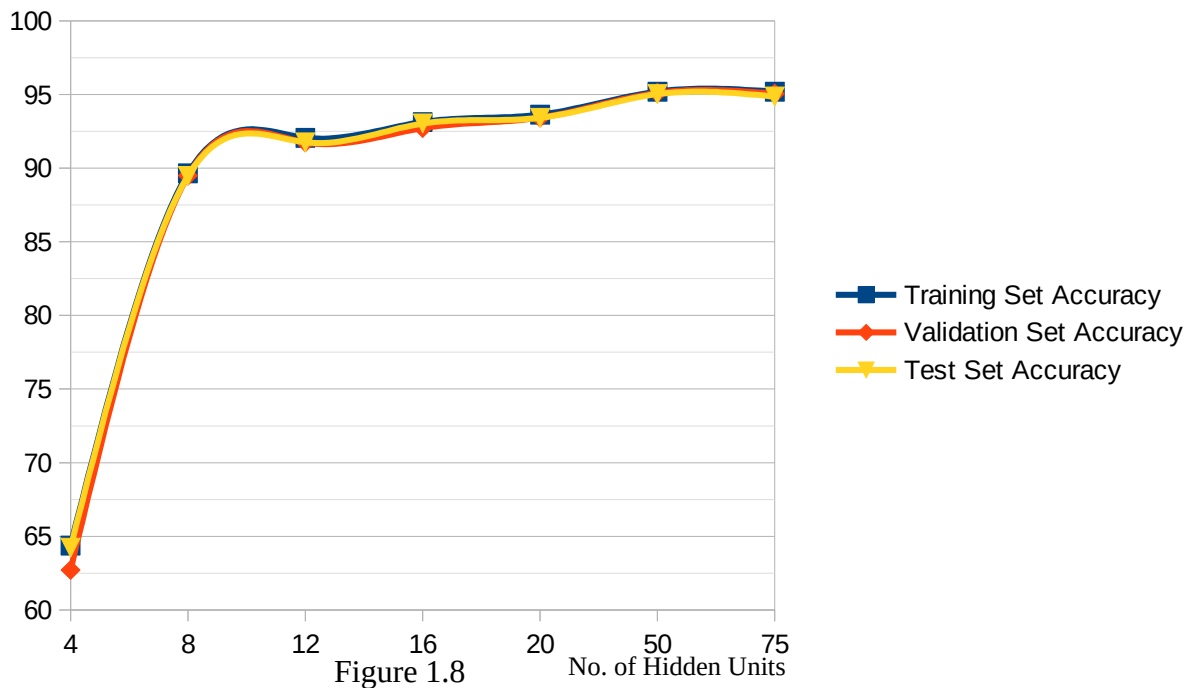


Figure 1.8 shows the accuracy obtained for Training, Validation and Test data sets for different number of hidden nodes. From the figure we can see that as the number of hidden nodes increases, the accuracy is getting increased. The rate of increase in accuracy is getting decreased as the number of hidden units is increased.

The below table and plot shows the training times for different number of hidden nodes

No. of Hidden Units (Lambda= 10)	Training Time
4	43.61418
8	42.1026
12	44.3904
16	49.2044
20	50.253
50	77.912
75	91.935

Table 1.9

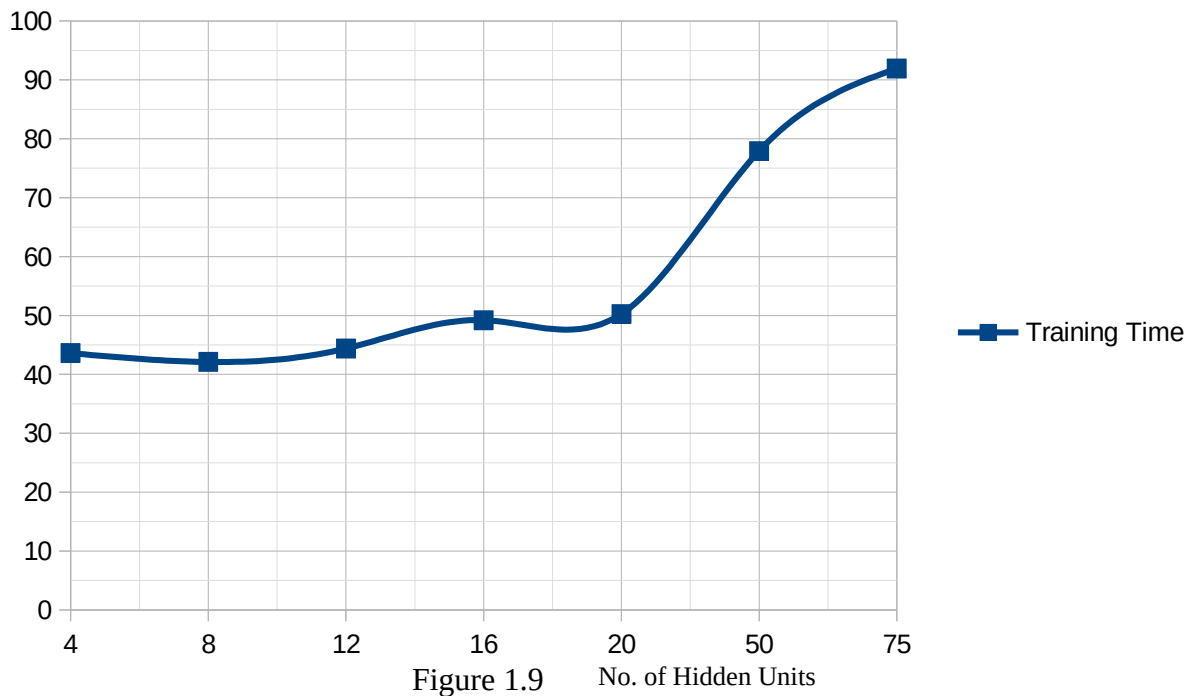


Figure 1.9 No. of Hidden Units

Figure 1.9 shows the Training time for different number of hidden nodes. As the number of hidden nodes increases, the training time increases.

Accuracy of classification method on the handwritten digits test data :

For $\lambda = 10$, number of hidden nodes = 50, the accuracy is given below

Training Set Accuracy	Validation Set Accuracy	Test Set Accuracy
95.196	95.11	95.03

Table 2.0

Accuracy of classification method on the CelebA data set:

For $\lambda = 10$, number of hidden nodes=50, the accuracy and training time is given below

Training Set Accuracy (%)	Validation Set Accuracy (%)	Test Set Accuracy (%)	Training Time (s)
84.5781990521327	83.67729831144464	84.36790310370931	220.3774597644806

Table 2.1

Comparison of Single Layer Neural Network(implemented by us) with a deep neural network in terms of accuracy and training time

The below table shows the test set accuracy obtained with single layer neural network(implemented by us), seep neural network with 3,5 and 7 layers

Parameters	Single Layer Neural Network(Impleme nted by us)	Deep Neural network with 3 Layers	Deep Neural network with 5 Layers	Deep Neural network with 7 Layers
Test Set Accuracy (%)	0.843679	0.810749	0.768736	0.750568

Table 2.2

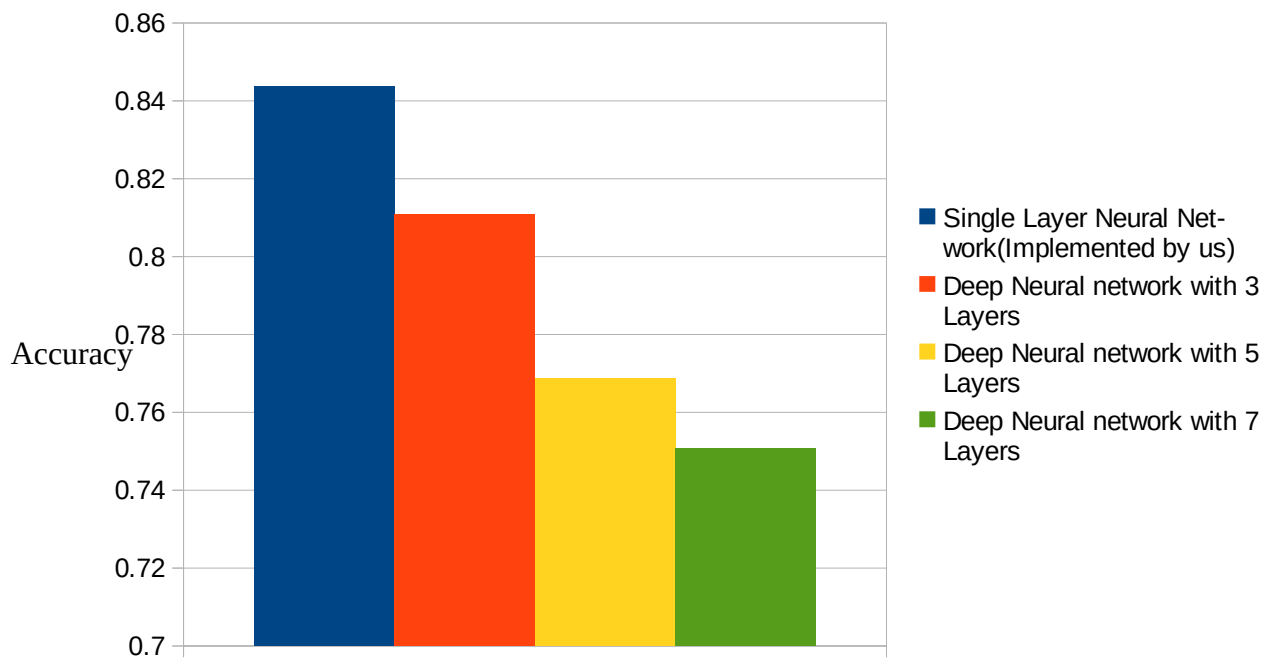


Figure 2.0

Figure 2.0 shows the accuracy on Test data set for our neural network, deep neural network with 3, 5 and 7 layers. From the figure we can see that our neural network with 1 hidden layer performs better when compared to deep neural networks with 3, 4 and 5 layers.

The below table shows the training time for single layer neural network(implemented by us), seep neural network with 3,5 and 7 layers

Parameters	Single Layer Neural Network(Impleme nted by us)	Deep Neural network with 3 Layers	Deep Neural network with 5 Layers	Deep Neural network with 7 Layers
Training time (s)	220.3774597644806	382.307737827301	454.8127586841583	530.6604232788086

Table 2.3

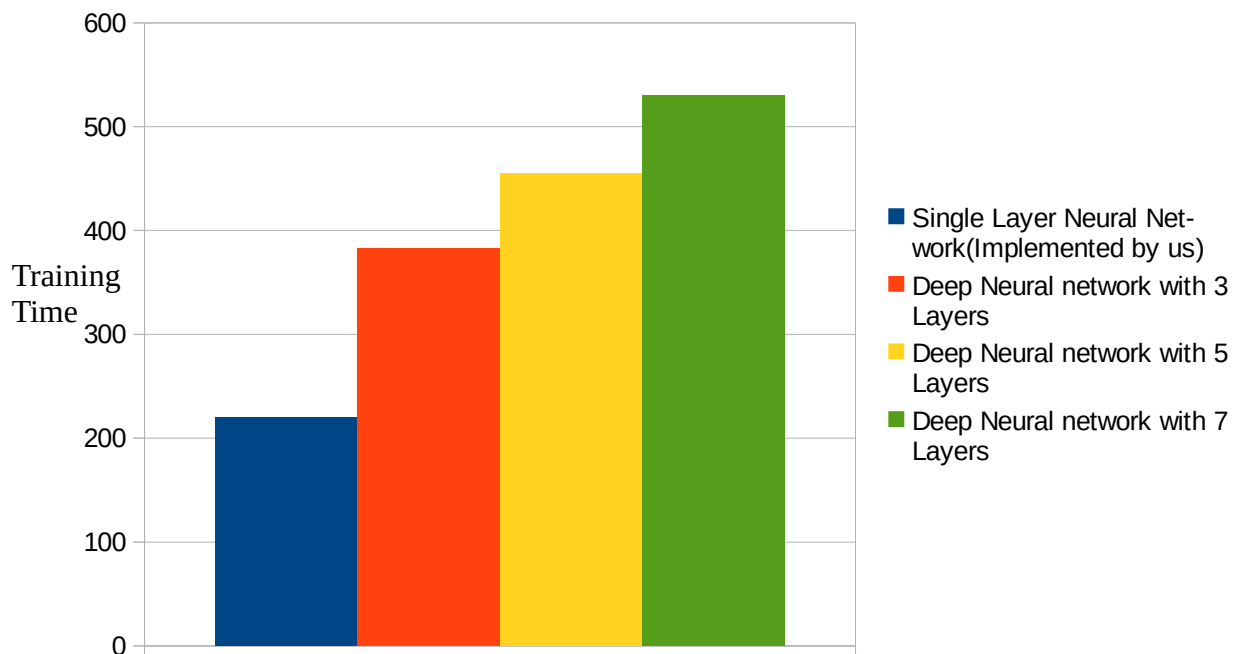


Figure 2.1

Figure 2.1 shows the training time taken by Single Layer Neural Network(implemented by us) and deep neural network with 3,5 and 7 layers. From the figure we can see that as the layers are getting increased, the training time is getting increased.

Results from Convolutional neural network in terms of accuracy and training time:

Size of:

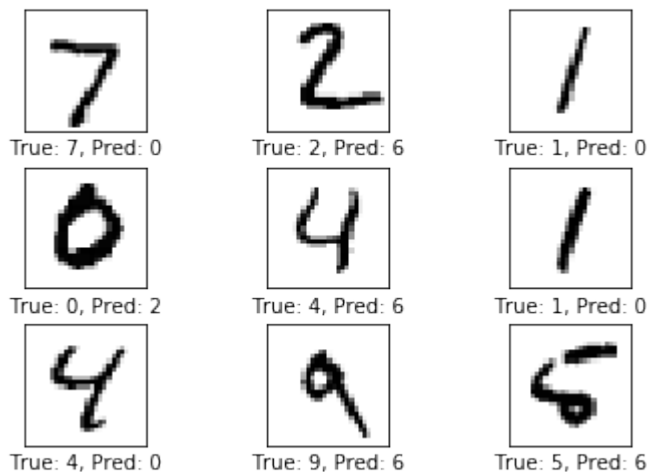
- Training-set: 55000

- Test-set: 10000

- Validation-set: 5000

Accuracy on Test-Set: 10.4% (1044 / 10000)

Example errors:



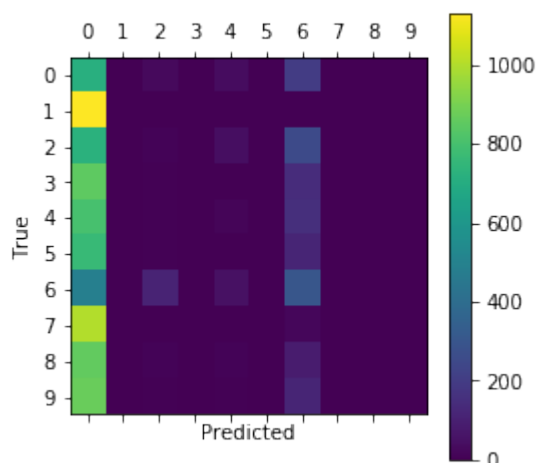
Confusion Matrix:

```
[[ 714  0  27  0  37  0 201  1  0  0]
 [1134  0  0  0  0  0  1  0  0  0]
 [ 726  0  11  0  41  0 254  0  0  0]]
```

```

[ 854  0  8  0  3  0 145  0  0  0]
[ 804  0  8  0 17  0 153  0  0  0]
[ 762  0  8  0  3  0 119  0  0  0]
[ 490  0 114  0 52  0 302  0  0  0]
[1003  0  2  0  4  0  19  0  0  0]
[ 860  0 13  0 13  0  88  0  0  0]
[ 877  0  7  0  7  0 118  0  0  0]

```



Optimization Iteration: 1, Training Accuracy: 14.1%

Time usage: 0:00:00

Accuracy on Test-Set: 10.3% (1026 / 10000)

Example errors:



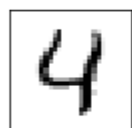
True: 7, Pred: 0



True: 2, Pred: 6



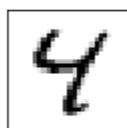
True: 1, Pred: 0



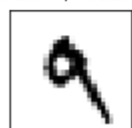
True: 4, Pred: 0



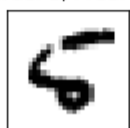
True: 1, Pred: 0



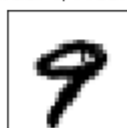
True: 4, Pred: 0



True: 9, Pred: 6



True: 5, Pred: 0



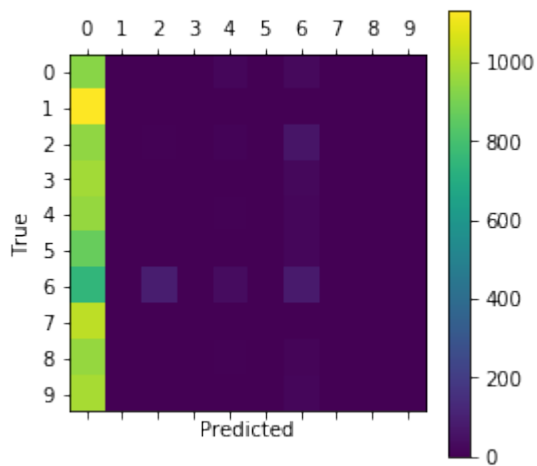
True: 9, Pred: 0

Confusion Matrix:

```

[[ 932  0  2  0 18  0 28  0  0  0]
 [1135  0  0  0  0  0  0  0  0  0]
 [ 946  0  6  0 13  0 66  0  1  0]
 [ 979  0  1  0  4  0 26  0  0  0]
 [ 955  0  1  0  5  0 21  0  0  0]
 [ 872  0  0  0  2  0 18  0  0  0]
 [ 746  0 90  0 38  0 83  0  1  0]
 [1024  0  0  0  1  0  3  0  0  0]
 [ 955  0  0  0  5  0 14  0  0  0]
 [ 986  0  1  0  4  0 18  0  0  0]]

```



Time usage: 0:00:08

Accuracy on Test-Set: 61.1% (6107 / 10000)

Example errors:



True: 2, Pred: 6



True: 4, Pred: 9



True: 4, Pred: 7



True: 5, Pred: 6



True: 9, Pred: 7



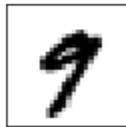
True: 6, Pred: 0



True: 5, Pred: 3



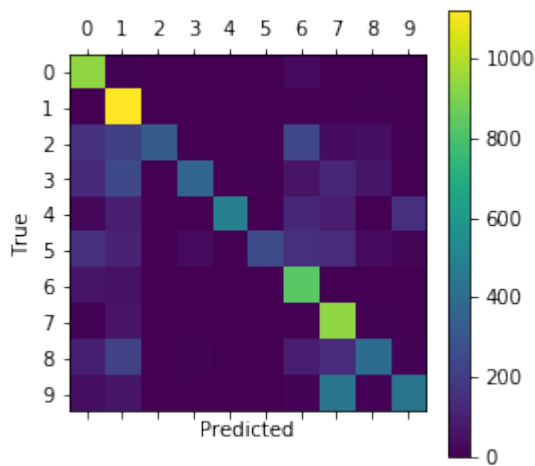
True: 3, Pred: 6



True: 9, Pred: 7

Confusion Matrix:

```
[[ 936  6  0  0  0  0  31  4  3  0]
 [  0 1123  1  0  0  0  3  0  8  0]
 [ 162 213 327  1  0  0 241 38 45  5]
 [ 140 249  0 365  0  5  58 120 70  3]
 [  25  94  0  0 491  0 129  84  2 157]
 [ 152 112  0  33  0 257 150 143 31 14]
 [  67  49  0  0  1  4 835  1  1  0]
 [  12  61  4  1  0  0  2 938  7  3]
 [  97 218  0  9  0  2  93 143 399 13]
 [  40  69  0  3  1  0  9 440 11 436]]
```



Optimization Iteration: 101, Training Accuracy: 60.9%
 Optimization Iteration: 201, Training Accuracy: 85.9%
 Optimization Iteration: 301, Training Accuracy: 92.2%
 Optimization Iteration: 401, Training Accuracy: 81.2%
 Optimization Iteration: 501, Training Accuracy: 82.8%
 Optimization Iteration: 601, Training Accuracy: 87.5%
 Optimization Iteration: 701, Training Accuracy: 96.9%
 Optimization Iteration: 801, Training Accuracy: 98.4%
 Optimization Iteration: 901, Training Accuracy: 93.8%

Time usage: 0:01:13

Accuracy on Test-Set: 93.7% (9365 / 10000)

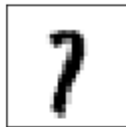
Example errors:



True: 4, Pred: 6



True: 7, Pred: 9



True: 7, Pred: 1



True: 7, Pred: 1



True: 3, Pred: 1



True: 6, Pred: 5



True: 8, Pred: 7



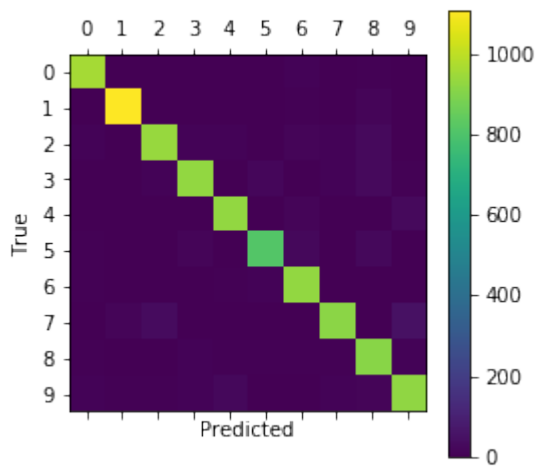
True: 9, Pred: 8



True: 3, Pred: 5

Confusion Matrix:

```
[[ 960  0  1  0  0  2 11  1  5  0]
 [  0 1109  3  2  0  1  5  0 15  0]
 [ 12  1 940 11 12  1 16  9 29  1]
 [  2  4 12 929  0 20  0  9 26  8]
 [  0  3  3  0 929  0 16  1  3 27]
 [  6  3  0 17  1 812 23  1 25  4]
 [  5  4  2  0  6  9 931  0  1  0]
 [  2 16 32  4  4  1  0 917  4 48]
 [  8  4  4 10  7  7  8  5 911 10]
 [ 12  6  5  9 24  4  0  9 13 927]]
```

Optimization Iteration: 1001, Training Accuracy: 100.0%

Optimization Iteration: 1101, Training Accuracy: 87.5%

Optimization Iteration: 1201, Training Accuracy: 93.8%

Optimization Iteration: 1301, Training Accuracy: 93.8%

Optimization Iteration: 1401, Training Accuracy: 90.6%

Optimization Iteration: 1501, Training Accuracy: 92.2%

Optimization Iteration: 1601, Training Accuracy: 92.2%

Optimization Iteration: 1701, Training Accuracy: 96.9%

Optimization Iteration: 1801, Training Accuracy: 96.9%

Optimization Iteration: 1901, Training Accuracy: 98.4%

Optimization Iteration: 2001, Training Accuracy: 96.9%

Optimization Iteration: 2101, Training Accuracy: 98.4%

Optimization Iteration: 2201, Training Accuracy: 98.4%

Optimization Iteration: 2301, Training Accuracy: 85.9%

Optimization Iteration: 2401, Training Accuracy: 95.3%

Optimization Iteration: 2501, Training Accuracy: 98.4%

Optimization Iteration: 2601, Training Accuracy: 96.9%

Optimization Iteration: 2701, Training Accuracy: 98.4%

Optimization Iteration: 2801, Training Accuracy: 96.9%

Optimization Iteration: 2901, Training Accuracy: 95.3%

Optimization Iteration: 3001, Training Accuracy: 95.3%

Optimization Iteration: 3101, Training Accuracy: 100.0%

Optimization Iteration: 3201, Training Accuracy: 98.4%

Optimization Iteration: 3301, Training Accuracy: 98.4%

Optimization Iteration: 3401, Training Accuracy: 96.9%

Optimization Iteration: 3501, Training Accuracy: 98.4%

Optimization Iteration: 3601, Training Accuracy: 98.4%

Optimization Iteration: 3701, Training Accuracy: 93.8%

Optimization Iteration: 3801, Training Accuracy: 98.4%

Optimization Iteration: 3901, Training Accuracy: 100.0%

Optimization Iteration: 4001, Training Accuracy: 96.9%

Optimization Iteration: 4101, Training Accuracy: 100.0%

Optimization Iteration: 4201, Training Accuracy: 96.9%

Optimization Iteration: 4301, Training Accuracy: 98.4%

Optimization Iteration: 4401, Training Accuracy: 100.0%

Optimization Iteration: 4501, Training Accuracy: 98.4%

Optimization Iteration: 4601, Training Accuracy: 98.4%

Optimization Iteration: 4701, Training Accuracy: 100.0%

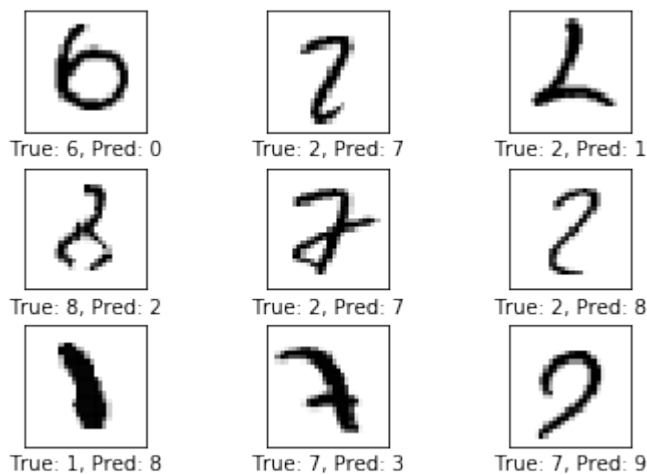
Optimization Iteration: 4801, Training Accuracy: 98.4%

Optimization Iteration: 4901, Training Accuracy: 98.4%
Optimization Iteration: 5001, Training Accuracy: 98.4%
Optimization Iteration: 5101, Training Accuracy: 98.4%
Optimization Iteration: 5201, Training Accuracy: 100.0%
Optimization Iteration: 5301, Training Accuracy: 98.4%
Optimization Iteration: 5401, Training Accuracy: 100.0%
Optimization Iteration: 5501, Training Accuracy: 96.9%
Optimization Iteration: 5601, Training Accuracy: 95.3%
Optimization Iteration: 5701, Training Accuracy: 95.3%
Optimization Iteration: 5801, Training Accuracy: 98.4%
Optimization Iteration: 5901, Training Accuracy: 96.9%
Optimization Iteration: 6001, Training Accuracy: 98.4%
Optimization Iteration: 6101, Training Accuracy: 100.0%
Optimization Iteration: 6201, Training Accuracy: 96.9%
Optimization Iteration: 6301, Training Accuracy: 96.9%
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: 98.4%
Optimization Iteration: 6801, Training Accuracy: 100.0%
Optimization Iteration: 6901, Training Accuracy: 98.4%
Optimization Iteration: 7001, Training Accuracy: 98.4%
Optimization Iteration: 7101, Training Accuracy: 98.4%
Optimization Iteration: 7201, Training Accuracy: 96.9%
Optimization Iteration: 7301, Training Accuracy: 100.0%
Optimization Iteration: 7401, Training Accuracy: 100.0%
Optimization Iteration: 7501, Training Accuracy: 98.4%
Optimization Iteration: 7601, Training Accuracy: 100.0%
Optimization Iteration: 7701, Training Accuracy: 98.4%
Optimization Iteration: 7801, Training Accuracy: 95.3%
Optimization Iteration: 7901, Training Accuracy: 98.4%
Optimization Iteration: 8001, Training Accuracy: 100.0%
Optimization Iteration: 8101, Training Accuracy: 100.0%
Optimization Iteration: 8201, Training Accuracy: 100.0%
Optimization Iteration: 8301, Training Accuracy: 100.0%
Optimization Iteration: 8401, Training Accuracy: 100.0%
Optimization Iteration: 8501, Training Accuracy: 100.0%
Optimization Iteration: 8601, Training Accuracy: 98.4%
Optimization Iteration: 8701, Training Accuracy: 100.0%
Optimization Iteration: 8801, Training Accuracy: 100.0%
Optimization Iteration: 8901, Training Accuracy: 100.0%
Optimization Iteration: 9001, Training Accuracy: 98.4%
Optimization Iteration: 9101, Training Accuracy: 100.0%
Optimization Iteration: 9201, Training Accuracy: 100.0%
Optimization Iteration: 9301, Training Accuracy: 100.0%
Optimization Iteration: 9401, Training Accuracy: 96.9%
Optimization Iteration: 9501, Training Accuracy: 100.0%
Optimization Iteration: 9601, Training Accuracy: 100.0%
Optimization Iteration: 9701, Training Accuracy: 98.4%
Optimization Iteration: 9801, Training Accuracy: 100.0%
Optimization Iteration: 9901, Training Accuracy: 100.0%

Time usage: 0:12:09

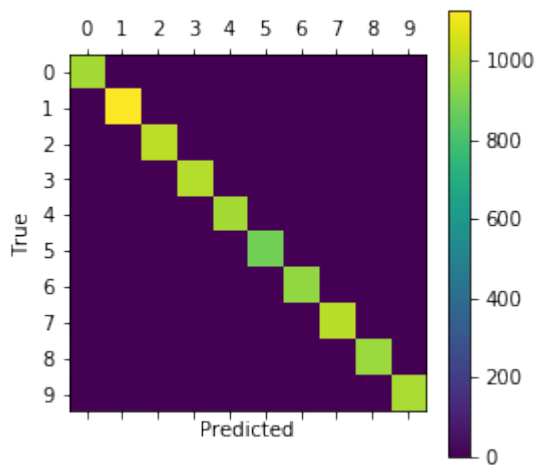
Accuracy on Test-Set: 98.8% (9876 / 10000)

Example errors:



Confusion Matrix:

```
[[ 974  0  0  0  0  0  2  0  3  1]
 [ 0 1128  0  0  0  1  1  0  5  0]
 [ 2  1 1014  6  1  0  0  3  5  0]
 [ 1  0  0 1001  0  4  0  2  2  0]
 [ 0  0  0  0 978  0  0  0  1  3]
 [ 1  1  0  2  0 886  1  1  0  0]
 [ 4  3  0  0  2  4 945  0  0  0]
 [ 0  1  5  3  0  0  0 1010  2  7]
 [ 5  0  1  2  2  1  1  2 957  3]
 [ 4  5  0  2  5  3  0  3  4 983]]
```



From the above results, we can see that initially when the weight vectors are initialized the accuracy on test set is very low as we did not train the neural network for the data set. The confusion matrix reflects this. As we are training the neural networks, the weight vectors are getting changed to suit the data set. After 9900 iterations, we can see from the confusion matrix that the class to which the data belongs is being identified correctly.