CSE574 Spring 2019 Introduction to Machine Learning
Programming Assignment 1

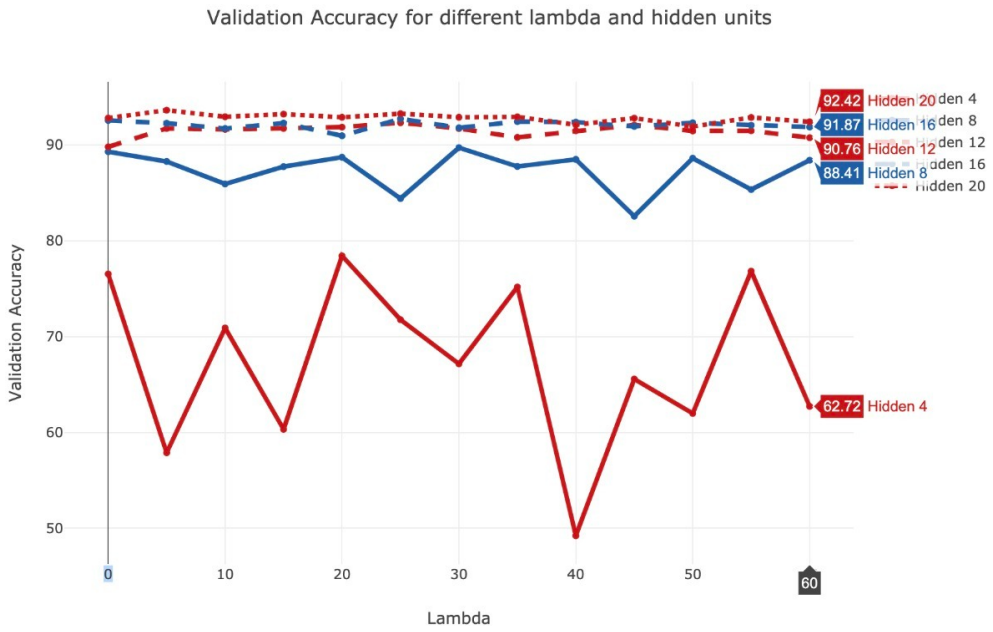
Name	UBIT
Narendra Badam	nbadam
Mohit Tripathi	mohittri
Saham	sahamati

NOTE: - For constructing interactive plots we have used plotly library and commented them out in our script. We generated the plots and posted them in our report. In order to view those plots IN SCRIPT, kindly uncomment the libraries, install them, and uncomment plotting code.

Ans – 1

Feature Selection: We have removed all the columns having only zeroes as they won't contribute towards the model building. After removing the columns the features in the data have been reduced from 784 to 718.

Diagrams to explain the relation between Lambda and performance of Neural Network



Lambda vs Validation Accuracy Plot

We can observe from the above plot that we get the best validation accuracy (92.42%) for lambda value=60 and 20 hidden units. The least validation accuracy which is approximately 49% can be observed for Lambda=40 and 4 hidden units.

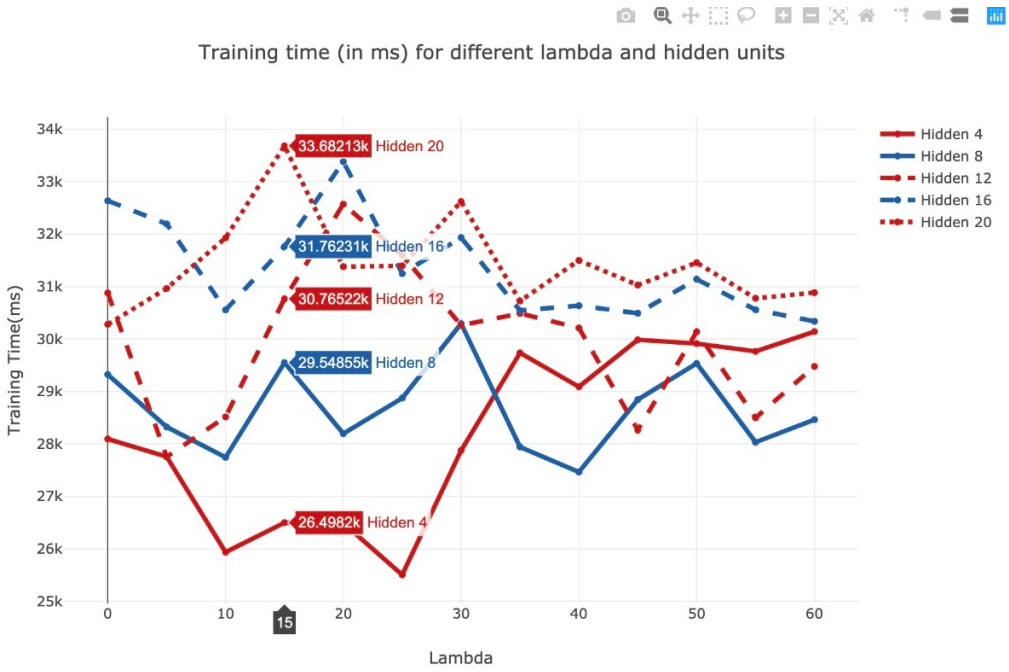
For Lambda

We can observe that with high value of lambda can lead to underfitting resulting in low validation and test accuracy, similarly at lambda=0 the model will overfit the data which leads to high training accuracy but low validation and test accuracy which would be disastrous for the model.

Hence, we select the optimum value of lambda (=60) in order to get the best test accuracy.

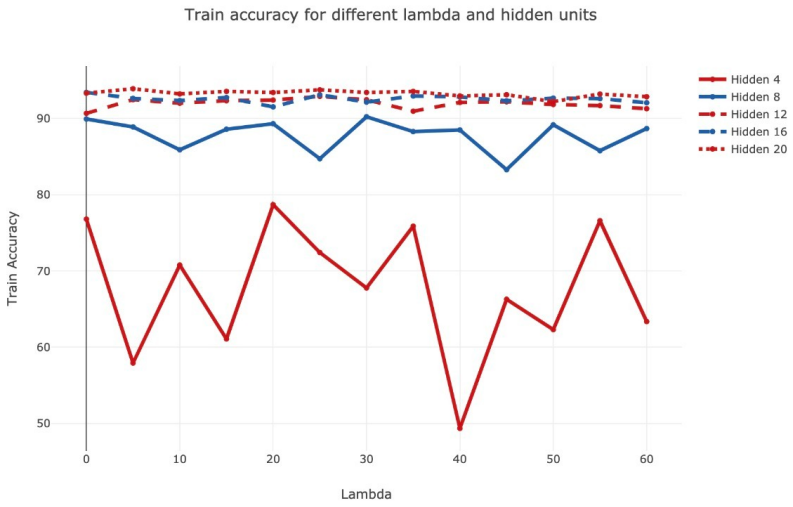
For Hidden Units: From the above graph we can infer that, as the number of hidden units increases accuracy also increases upto a certain point, however, after a certain limit, the model may overfit the data due to high hidden units which leads to high test error. Also, high number of hidden units is computationally challenging for calculation leading to a complex model. Hence, we plotted accuracy vs Number of hidden units in the above graph to decide the optimal number of hidden units ie 20

Training time(ms) for different number of Hidden units



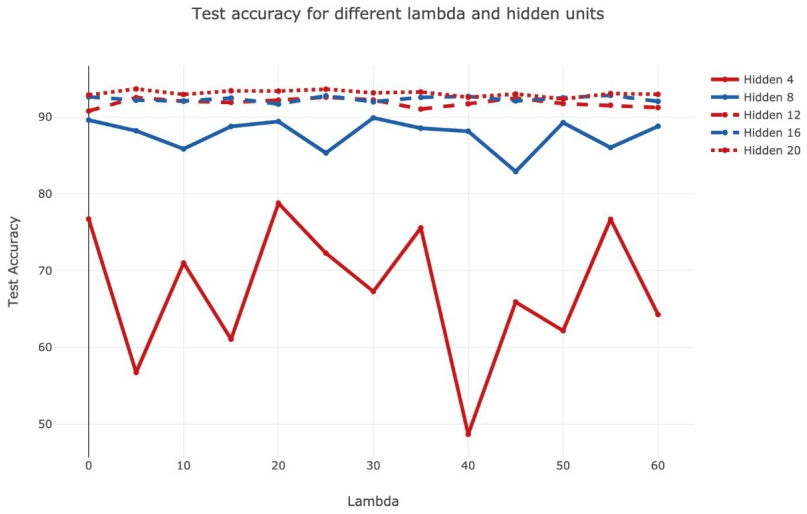
From the above graph we can observe that for a particular lambda(=15) ,time increases as the number of hidden units increases and the training time is maximum for model with 20 hidden units

Train Accuracy for various Lambda and Hidden units:



As lambda is increasing, train accuracy is decreasing slightly to counter overfitting

Test Accuracy for various Lambda and Hidden units:



By choosing the best parameters(lambda = 60 and hidden units = 20) below are the accuracies for test, training and validation data sets:

```
Training set Accuracy with hyperparameter:92.818%

Validation set Accuracy with hyperparameter:92.42%

Test set Accuracy with hyperparameter:92.81%
```

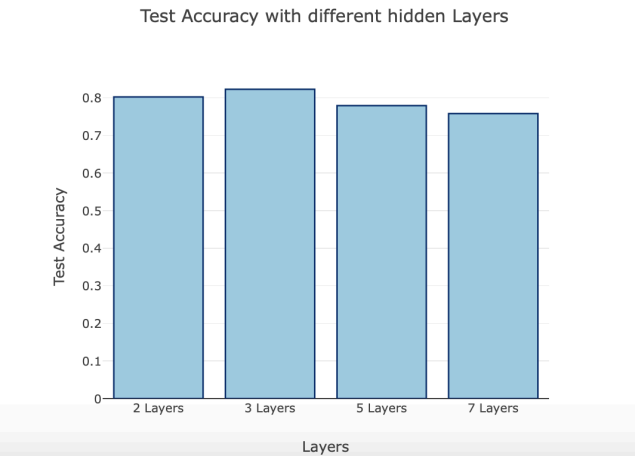
Accuracy of classication method on the CelebA data set:

Accuracy of single hidden layer Neural Network on CelebA data set (test data only), to distinguish between two classes - wearing glasses and not wearing glasses. Use facennScript.py to obtain these results.

The accuracy results for single layer Neural Network on CelebA which were obtained using facennScript.py . The test accuracy is 83.8%.

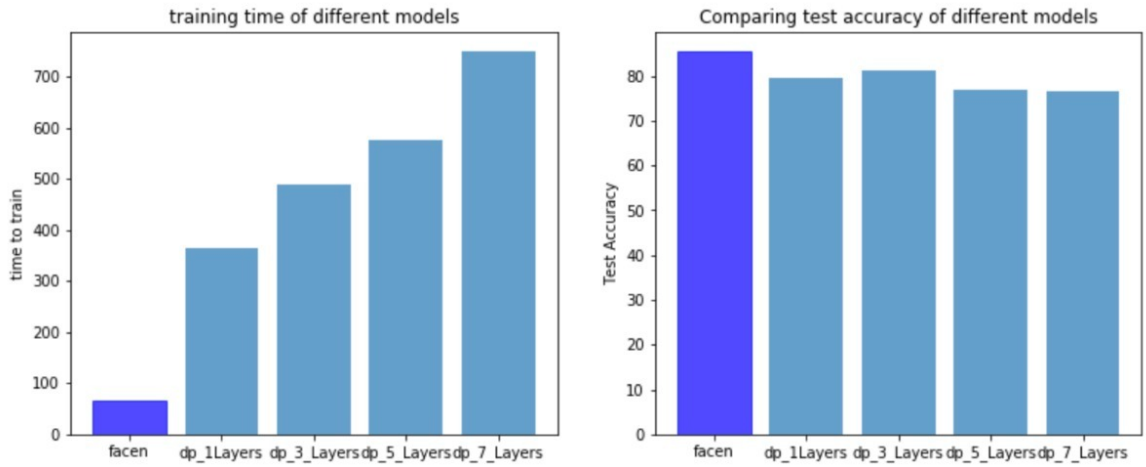
```
Training set Accuracy:83.23222748815165%

Validation set Accuracy:81.38836772983115%

Test set Accuracy:83.8001514004542%
```

Evaluate the accuracy of deep Neural Network (try 3, 5, and 7 hidden layers) on CelebA data set (test data only). Use deepnnScript.py to obtain these results.



From the above graph we can infer that accuracy increases as we increase number of layers from 2 to 3 but decreases as we increase the number of Layers more than 3 due to overfitting

Compare the performance of single vs. deep Neural Networks in terms of accuracy on test data and learning time.



From the "Training time of different models" (left) plot we can infer that for a simple 1 layer neural network built by us takes less time for training and as the number of layers increases, training time also increases.

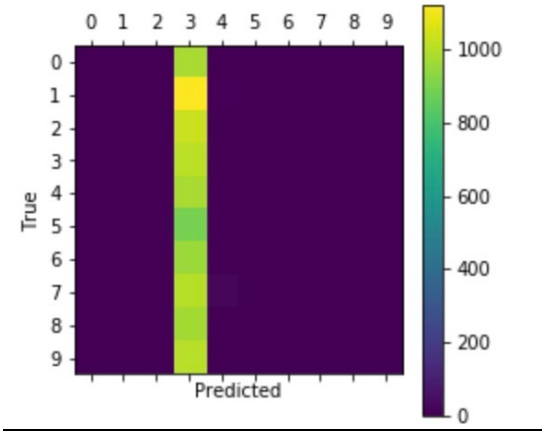
From the "Comparing Test Accuracy of different models" (right) plot we can observe that the one layer facenn model built by us has a better test accuracy than the one layer model test accuracy obtained through Tensor Flow. Also change in number of layers changes the test accuracy of models built by Tensor flow keeps fluctuating.

Run the convolutional neural network code and print out the results, for example the confusion matrix:

Confusion Matrix for different iterations at different time intervals:

```
Accuracy on Test-Set: 10.1% (1011 / 10000)
Confusion Matrix:
[[   0    0    0  980    0    0    0    0    0    0]
 [   0    0    0 1123   12    0    0    0    0    0]
 [   0    0    0 1032    0    0    0    0    0    0]
 [   0    0    0 1010    0    0    0    0    0    0]
 [   0    0    0  981    1    0    0    0    0    0]
 [   0    0    0  891    1    0    0    0    0    0]
 [   0    0    0  958    0    0    0    0    0    0]
 [   0    0    0 1001   27    0    0    0    0    0]
 [   0    0    0  973    1    0    0    0    0    0]
 [   0    0    0 1005    4    0    0    0    0    0]]
```

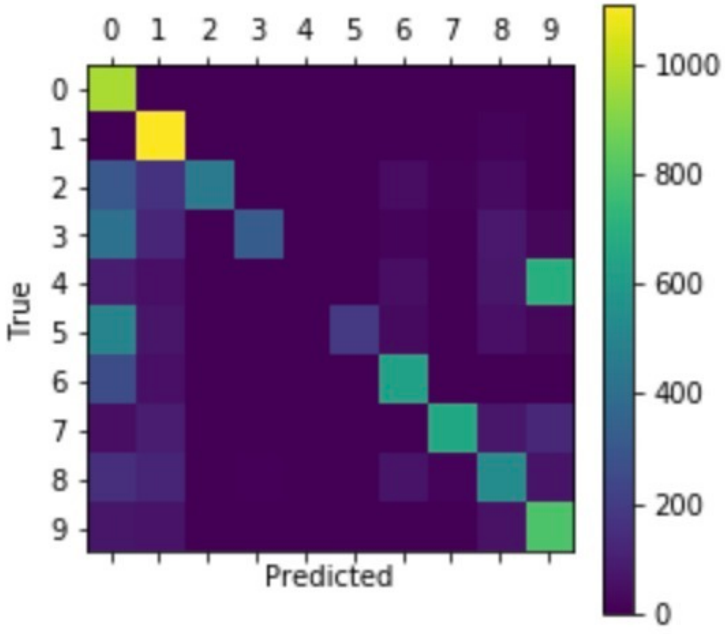
Confusion Matrix Plot



Confusion Matrix:

```
Accuracy on Test-Set: 56.9% (5692 / 10000)
Confusion Matrix:
[[ 967    1    0    0    0    1    7    0    4    0]
 [   1 1110    0    2    0    0    4    0   18    0]
 [ 307  163  456    0    0    0   45   14   42    5]
 [ 408  130    8  329    0    7   21    4   80   23]
 [  94   55    1    0    0    0   50    8   75  699]
 [ 501   72    0    3    0  191   35    5   55   30]
 [ 259   55    0    0    0    3  637    1    1    2]
 [  52   94    6    1    0    0    1  662   78  134]
 [ 151  119    1    9    0    3   67   18  538   68]
 [  73   65    0    0    0    0    2    7   60  802]]
```

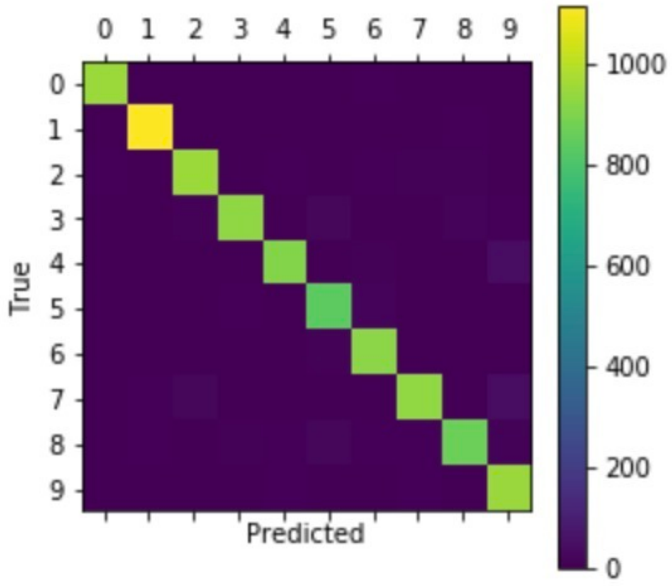
Confusion Matrix Plot:



Confusion Matrix:

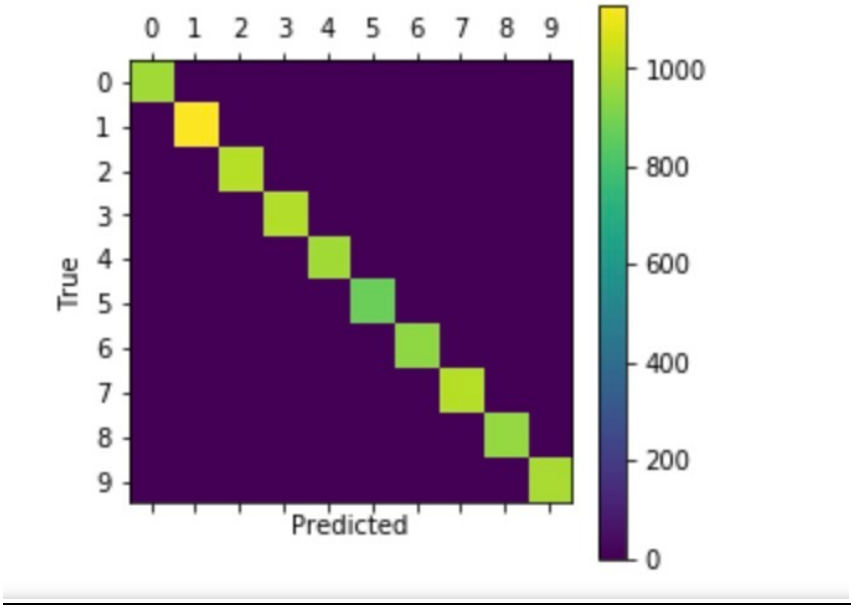
```
Accuracy on Test-Set: 93.7% (9372 / 10000)
Confusion Matrix:
[[ 952    0    4    1    0    5   13    2    3    0]
 [   0 1116    3    2    0    1    4    0    9    0]
 [  10    2  952    8   12    3   12   15   16    2]
 [   1    4   14  932    0   23    1    8   18    9]
 [   1    3    3    0  911    0   15    1    3   45]
 [   4    3    0   12    3  837   18    1    8    6]
 [   6    4    2    0    8   14  921    1    2    0]
 [   1   11   24    4    4    0    0  933    4   47]
 [   7    9    5   16    9   25   10   10  868   15]
 [   7    6    6    8    9    7    0   11    5  950]]
```

Confusion Matrix Plot:

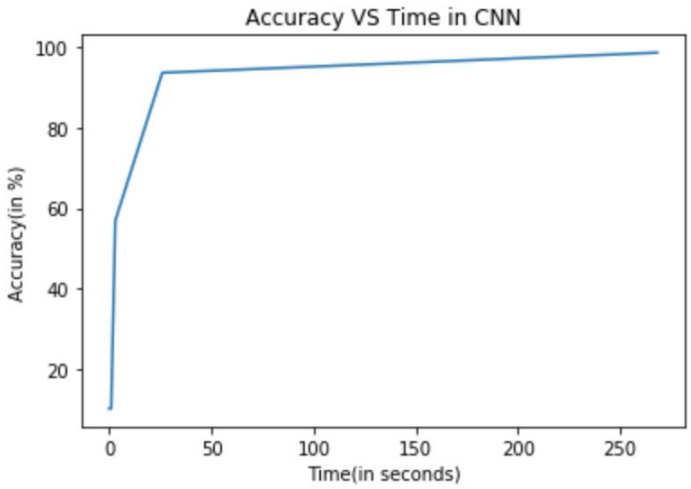


Confusion Matrix

```
Accuracy on Test-Set: 98.7% (9865 / 10000)
Confusion Matrix:
[[ 972    0    1    1    0    1    2    0    3    0]
 [   0 1131    1    0    0    0    1    0    2    0]
 [   5    5 1009    5    1    0    0    2    5    0]
 [   1    0    0 1007    0    0    0    0    2    0]
 [   0    0    1    1  973    0    0    1    1    5]
 [   2    0    0    8    0  879    1    0    0    2]
 [   5    2    0    1    1    4  944    0    1    0]
 [   1    3    8    3    0    0    0 1010    1    2]
 [   5    0    1    6    1    3    0    2  952    4]
 [   3    5    0    5    4    2    0    1    1  988]]
```



Report the results from convolutional neural network in terms of accuracy and training time. (20 extra points)



As the number of iterations increases (as well as time), the accuracy of the of CNN model is increasing.