University at Buffalo
CSE 574 Introduction to Machine Learning

Programming Assignment 2 Report

Submitted by: **Group 95**
Vishal Chaurasia – 50353581
Pushkar K Pandey – 50412529
Varun Sudarshan – 50360324

November 14, 2021

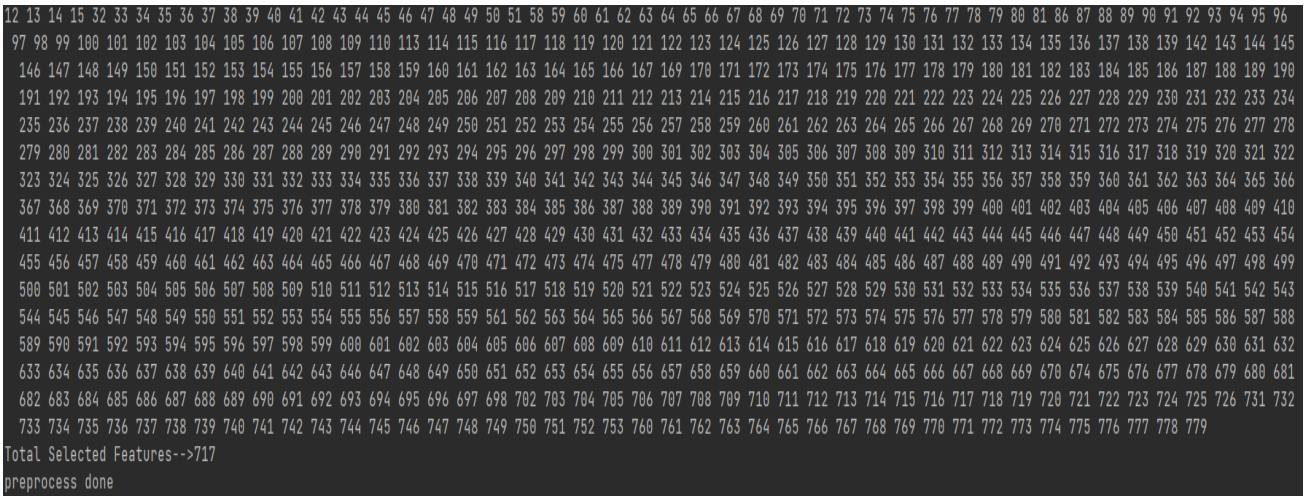
Introduction

In this assignment, we had to implement a Multilayer Perceptron Neural Network & evaluate its performance in classifying handwritten digits. Additionally, we used the same network to analyze a more challenging face dataset and compare the neural network's performance against a deep neural network and a convolutional neural network using the TensorFlow library.

Feature Selection

While pre-processing the dataset, we noticed many redundant features, i.e., their values were the same for all points in the training set. Since no additional information could be derived from the redundant features, we choose to ignore them.

Total number of features selected: **717**.
The following figure depicts the indices of the features selected.



How to decide hyper-parameter for Neural Network:

As explained below, we achieved the best combination by using different values of **λ** and no of hidden units (m) and comparing the test accuracy. We found the optimal hyperparameters (regularization coefficient, **λ** and no of hidden units **m**) combination with supporting data.

Choosing the best combination:

Ranging **λ** from 0-60 with an increment value of 10 & m from 4-20 with an increment value of 14:

λ	**m**	Train Accuracy	Validation accuracy	Test accuracy	Training Time
30	20	93.53	92.820	93.300	34.67
00	20	93.528	92.39	93.28	34.73
00	16	93.66	92.61	93.28	30.93
10	20	93.62	93.02	92.91	32.23
10	12	91.67	90.77	91.5	30.12
20	20	93.226	92.528	92.77	34.28
30	60	92.476	91.73	92.14	30.65

Above are the top hyperparameter combination that resulted in the best testing accuracy result

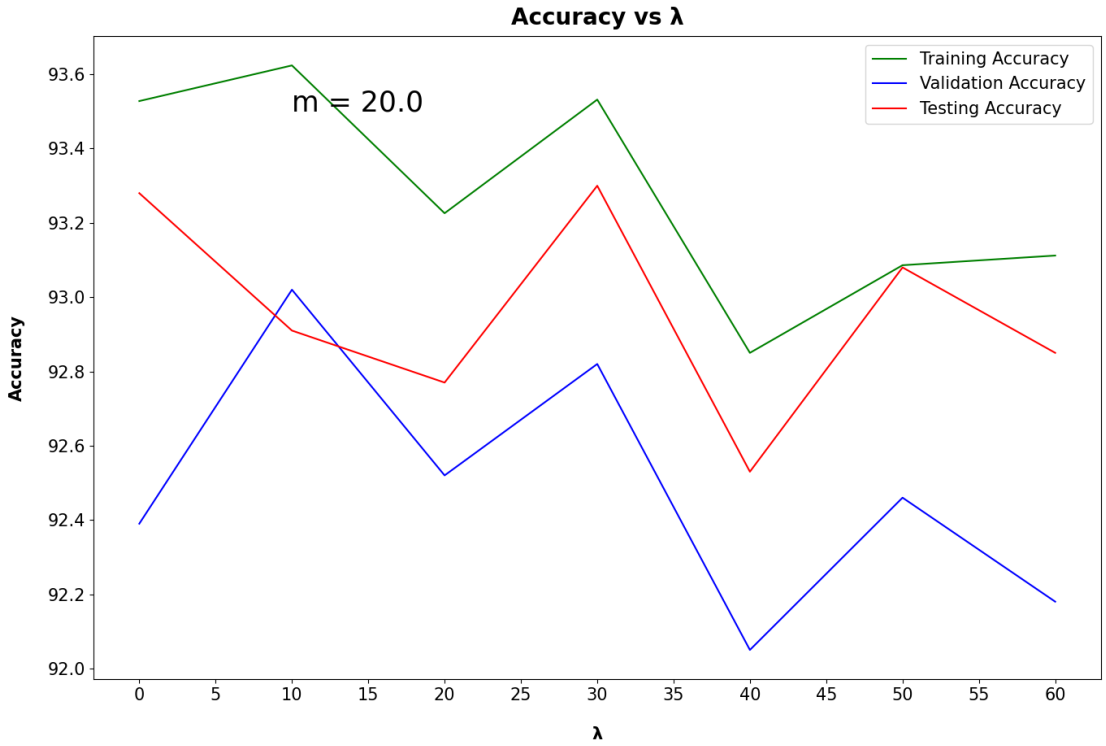
We can notice from the above table that our optimum **λ** and m combination is 30 and 20; this gives us the best test accuracy, 93.30 %.

We try changing **λ** by fixing the value of m to 20.

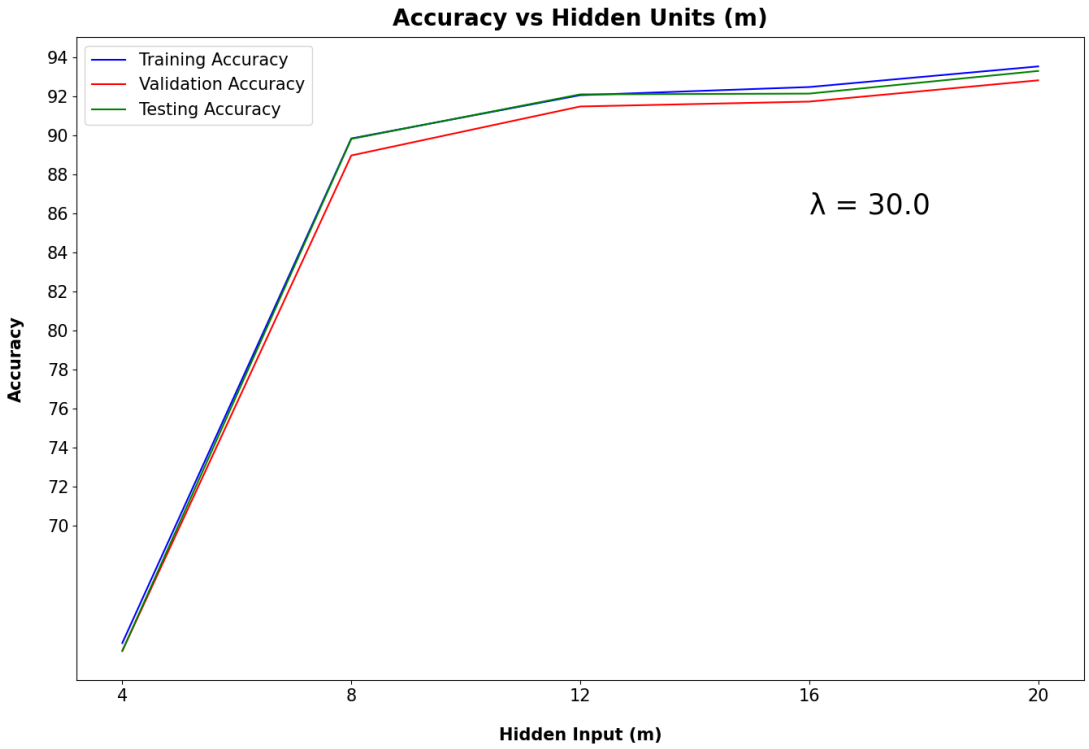
We can find from the resulting table that the value of **λ** = 30 gives us the optimum value.

λ	**m**	Train Accuracy	Validation accuracy	Test accuracy	Training Time
00	20	93.528	92.39	93.28	34.73
10	20	93.624	93.02	92.91	32.23
20	20	93.226	92.52	92.77	34.28
30	20	93.53	92.82	93.30	34.67
40	20	92.85	92.05	92.53	33.11
50	20	93.08	92.46	93.08	32.51
60	20	93.11	92.17	92.85	36.51

From the below graph, we can say that we get the best testing accuracy at **λ** = 30.



We try changing **m** by fixing the value of **λ** to 30.



We can find from the resulting table that the value of m = 20 gives us the optimum value.

λ	**m**	Train Accuracy	Validation accuracy	Test accuracy	Training Time
30	04	63.99	63.59	63.59	26.15
30	08	89.84	88.97	89.82	27.11
30	12	92.06	91.47	92.10	28.15
30	16	92.47	91.73	92.14	30.65
30	20	93.53	92.82	93.30	34.67

By taking Hyper Parameter combination: λ = 30, m = 20 we get the best test accuracy as 93.30%

Accuracy on the celebA dataset:

We reused our implementations of the functions - sigmoid(), nnObjFunc( ) and nnPredict( ) from the nnScript.py in the facennScript.py. Below are the accuracy obtained by running the facennScript.py for $\boldsymbol{\lambda} = 10$:

Training set Accuracy:	**84.22%**
Validation set Accuracy:	**83.23%**
Test set Accuracy:	**84.44%**
Training Time:	**158.43 s**

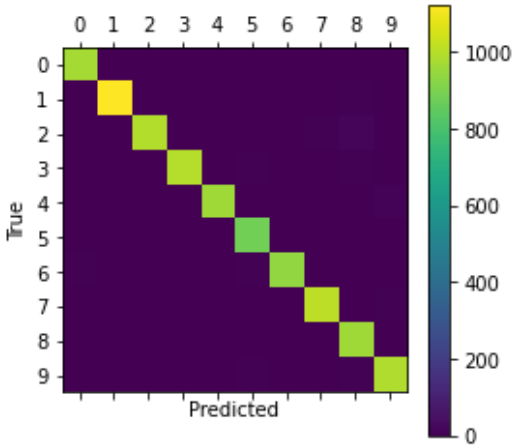
Deep NN Observations:

Hidden Layers	Accuracy (%)	Training time (s)
3	81.33	465.05
5	81.86	438.45
7	82.70	462.07
9	83.34	543.26
11	80.58	399.01

- The accuracy increases with an increase in the number of hidden layers up to a certain point and then decreases due to overfitting
- The Peak Accuracy is 83.34%
- The peak accuracy is less than the accuracy we observed for a NN with one hidden layer (84.44%).
- The training time for the NN (158.43 s) was significantly lesser than the deep NN's training time as there is only 1 hidden layer.

CNN Observations:

```
Time usage: 0:11:04
Accuracy on Test-Set: 98.5% (9847 / 10000)
Confusion Matrix:
[[ 973    0    0    0    0    1    2    1    3    0]
 [   0 1124    1    0    0    0    1    1    8    0]
 [   4    4  995    4    1    0    0    5   19    0]
 [   1    0    0  997    0    5    0    2    5    0]
 [   1    0    1    0  965    0    1    2    1   11]
 [   2    0    0    2    0  885    1    0    2    0]
 [   5    1    0    1    1    7  942    0    1    0]
 [   0    1    3    1    0    0    0 1014    2    7]
 [   3    0    1    2    0    3    0    0  963    2]
 [   1    3    0    1    2    5    1    2    5  989]]
```



- Accuracy of CNN is 98.5 percent
- Training Time: 11 Minutes 04 Seconds