CSE 574 - Introduction to Machine Learning

Handwritten Digits Classification Using Neural Networks

Group Number: 48

Kirti Hari - 50208065

Shruti Kulkarni - 50207124

Shashank Suresh - 50208025

PROJECT DESCRIPTION AND DATA

The Regularization Factor (λ) is used to avoid the over-fitting problem i.e., to have an optimal learnt Neural Network which does pretty good on both the train and test data. The model that is learnt is said to be:

- Under fitting if it performs badly (achieving low prediction accuracy) on the training data set itself.
- Over fitting if it performs very well on the training data set but performs badly on the test data set.

EXPLANATION WITH SUPPORTING FIGURES OF HOW TO CHOOSE THE HYPER-PARAMETER FOR NEURAL NETWORK

The efficiency or accuracy of the trained neural network depends on the number of hidden layers present and also depends on number of nodes present in each layer.

The neural network which we have designed has only one hidden layer. Initially, we trained the neural network with number of hidden nodes as 50 for different values of λ.

Below is the table showing values of accuracy on the three data sets for different values of λ.

λ	Training Data Accuracy	Validation Data Accuracy	Test Data Accuracy
0	95.8	94.9	95.24
5	95.912	95.16	95.4
10	94.66	94.24	94.6
15	93.67	93.56	93.74
20	93.066	92.55	93.13
25	92.39	92.12	92.57
30	91.54	91.35	91.87
35	91.45	90.82	91.78
40	90.966	90.94	91.36
45	90.76	90.43	91.07
50	90.41	90.09	90.87
55	90.404	89.9	90.79
60	90.082	90.1	90.64

Below is the plot of the above table:

Lambda vs Accuracy with Hidden Nodes = 50
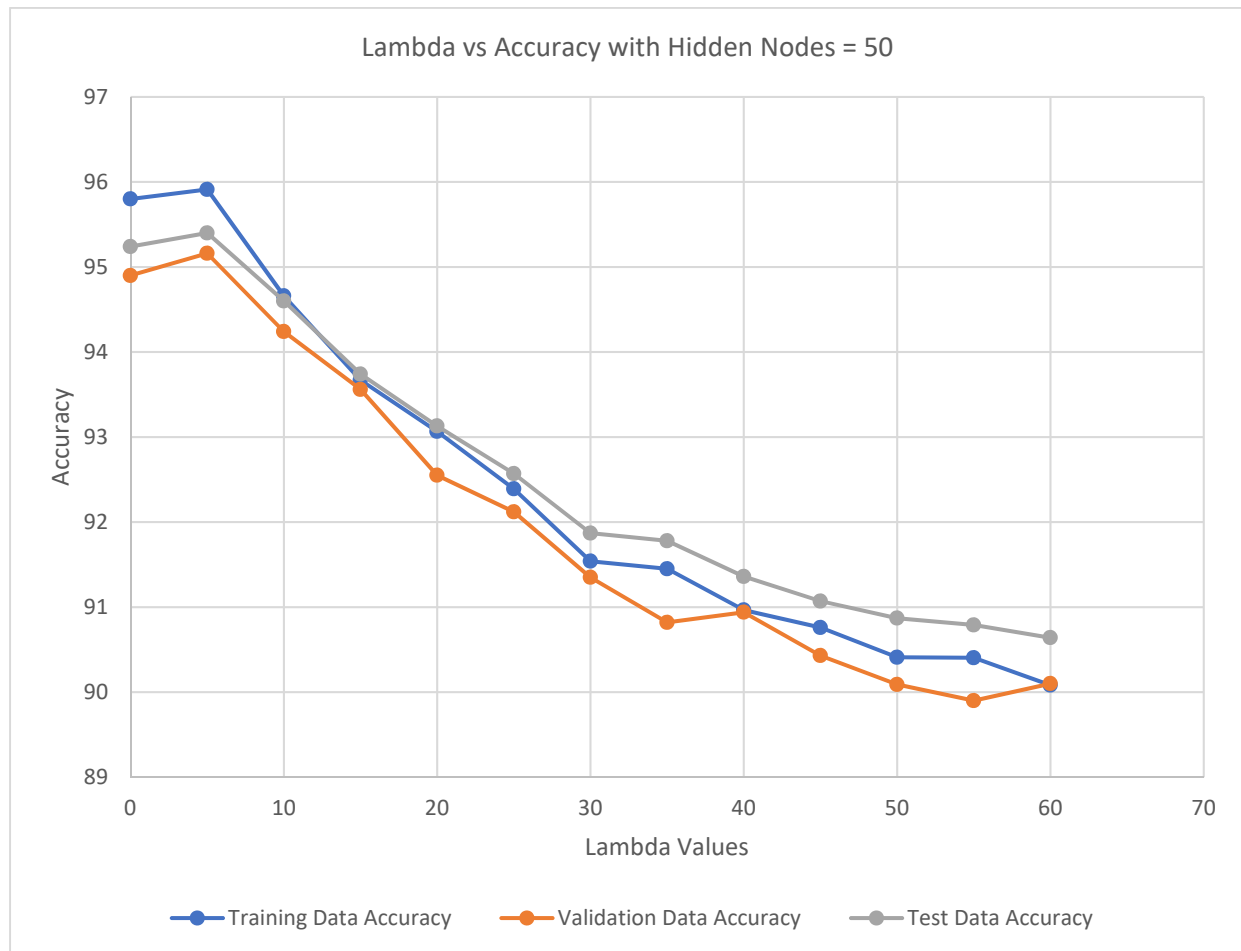
From the above plot, we see that as λ increases, its accuracy of prediction on the training data set decreases. We can clearly see the over fitting problem when λ is high and the under fitting problem when λ is low.

Next, we have tried for different number of hidden nodes with λ = 5 (as we got highest accuracy from the above case).

Hidden Nodes	Training Data Accuracy	Validation Data Accuracy	Test Data Accuracy	Training Time
4	82.216	81.25	80.78	186.77
8	90.442	89.52	90.25	190.38
12	93.19	91.8	92.63	228
16	94.148	93.07	93.68	272.59
20	94.67	93.63	94.13	358.23

Below is the plot of the above table:

Hidden Nodes vs Training Time with λ = 5
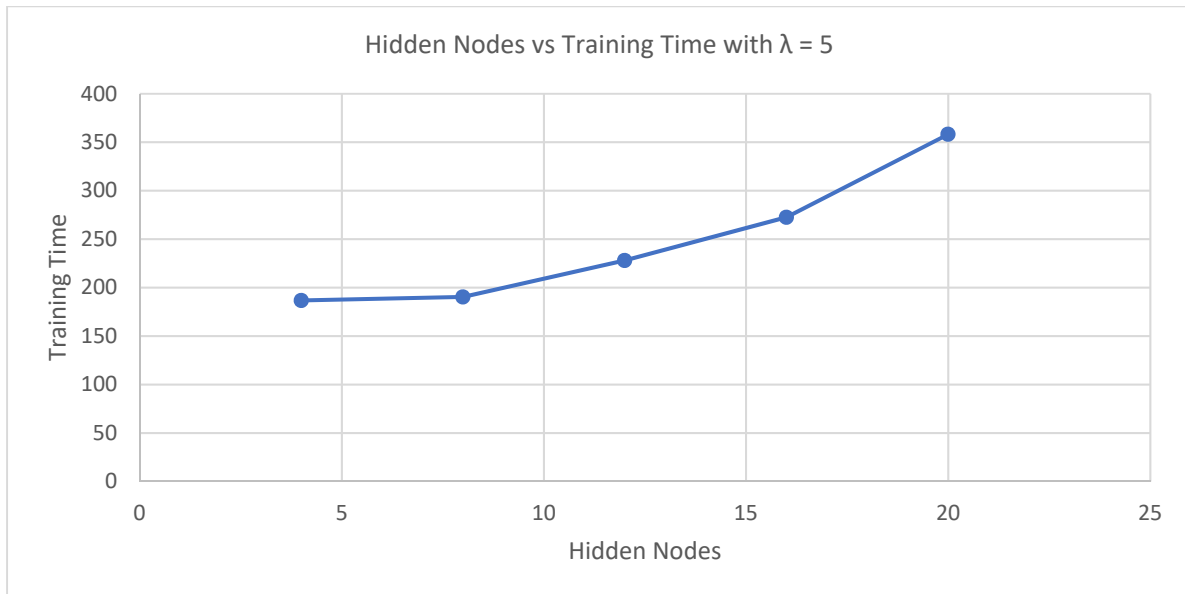
With this we clearly see that as the number of hidden nodes increases, the time taken for training the data also increases.

ACCURACY OF CLASSIFICATION METHOD ON THE HANDWRITTEN DIGITS TEST DATA

Final accuracy values with λ = 5 and hidden nodes = 50.

λ	Training Data Accuracy	Validation Data Accuracy	Test Data Accuracy
5	95.912	95.16	95.4

ACCURACY OF CLASSIFICATION METHOD ON THE CELEBA DATA SET

By running the default facennScript.py script, with λ = 10 and n_hidden nodes = 256, we observed the following results.

| λ	Training Data Accuracy	Validation Data Accuracy	Test Data Accuracy
10	83.64	82.25	84.14

On changing these hyper parameters to the optimal values observed in nnScript.py, which are λ = 5 and n_hidden = 50, we noticed the following:

λ	Training Data Accuracy	Validation Data Accuracy	Test Data Accuracy
5	85.44	83.93	85.54

COMPARISON OF YOUR NEURAL NETWORK WITH A DEEP NEURAL NETWORK (USING TENSORFLOW) IN TERMS OF ACCURACY AND TRAINING TIME

By varying the number of hidden layers in the Deep Neural Network (deepnnScript.py) we observed the following:

Number of Hidden layers	Accuracy on Test Data
2	0.82
3	0.777
5	0.771
7	0.741
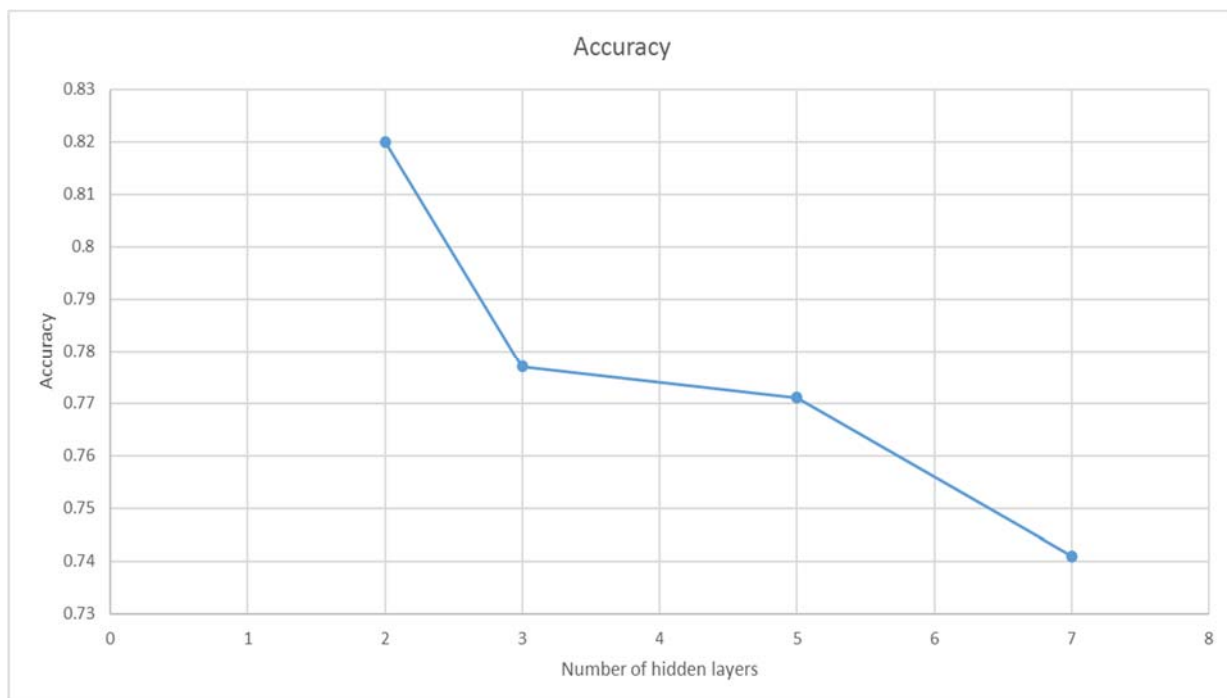
We observed that the best accuracy of the Deep Neural Networked is achieved with 2 hidden layers as 82% which takes a training time of 300.28 seconds.

We then ran our Neural Network with hyper parameters as those obtained in the nnScript.py and got an accuracy of 85.54%, with a training time of 102.86 seconds.

Type of NN	Accuracy	Training Time in Seconds
Our Neural Network(facennScript.py)	85.54	102.86
Deep Neural Network (deepnnScript.py)	82	300.28

We observed that our single layered Neural Network outperformed the Deep Neural Network in terms of accuracy and training time for the given test data.