CSE 574

INTRODUCTION TO MACHINE LEARNING

PROGRAMMING ASSIGNMENT 2

HANDWRITTEN DIGITS CLASSIFICATION

BY

HARIKRISHNA RANGINEENI	**50248741**	**hrangine**
AJAYSAI POTLURI	**50246594**	**ajaysaip**

UNIVERSITY AT BUFFALO

Spring 2018

CONTENTS

1. INTRODUCTION

This project implements a Multilayer Perceptron Neural Network and evaluate its performance in classifying handwritten digits. Further, same neural network implementation is used to analyze a more challenging face dataset and compare the performance of the neural network against a deep neural network using the TensorFlow library.

Our Machine Learning Model Implements Forward feed and back propagation methodology to adjust weights of model for predictive learning. The parameters in Neural Network model are the weights associated with the hidden layer units and the output layer units.

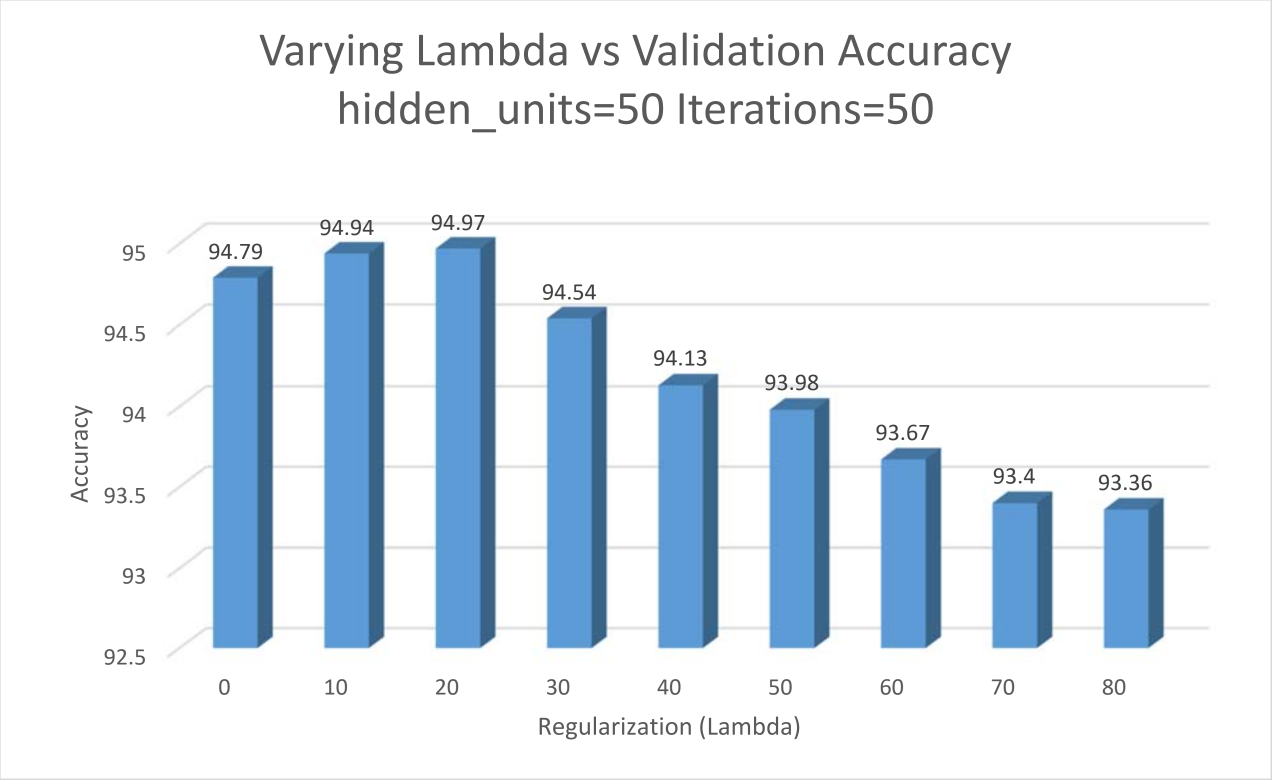
This project discusses relations and dependency of accuracy and training time of neural network on parameters such as number of Hidden nodes, regularization coefficient, and maximum iterations of conjugate gradient algorithm to perform optimization task.

2. CHOOSING HYPER PARAMETERS FOR NEURAL NETWORKS

The accuracy and speed of a neural network depends on multiple factors such as number of hidden layers, number of hidden units in each layer, size of the data set, and the regularization parameter lambda, and the number of iterations the neural network runs for. Our implementation of the neural network for classifying handwritten digits from the MNIST data set consists of a single hidden layer. Thus we have performed various tests to tune the regularization parameter (lambda) and the find the optimal number of hidden units so that the neural network gives a high accuracy. These tests include varying lambda from 0 to 80, changing the number of hidden units and also determining the effect of training the network with higher iterations.

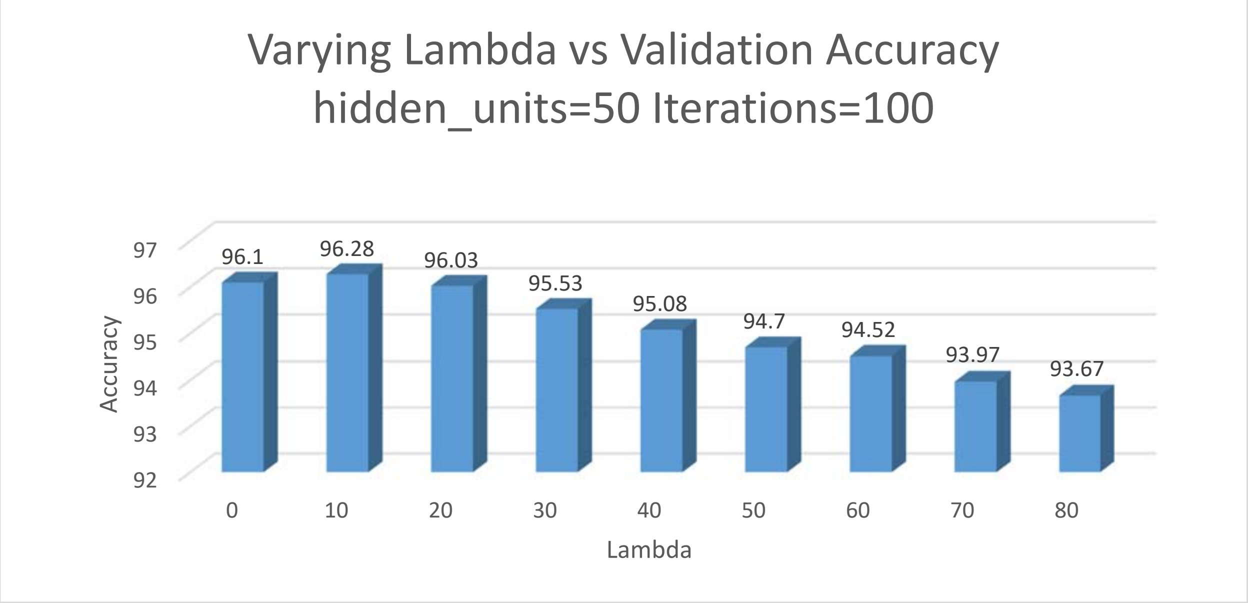
2.1. Training Data

2.1.1. Number of hidden units as 50 and number of iterations as 50



We vary lambda from 0 to 80. As observed above, increasing lambda gradually resulted in decreased accuracy for the validation data. The top 2 accuracy values were when Lambda was kept 10 and 20.

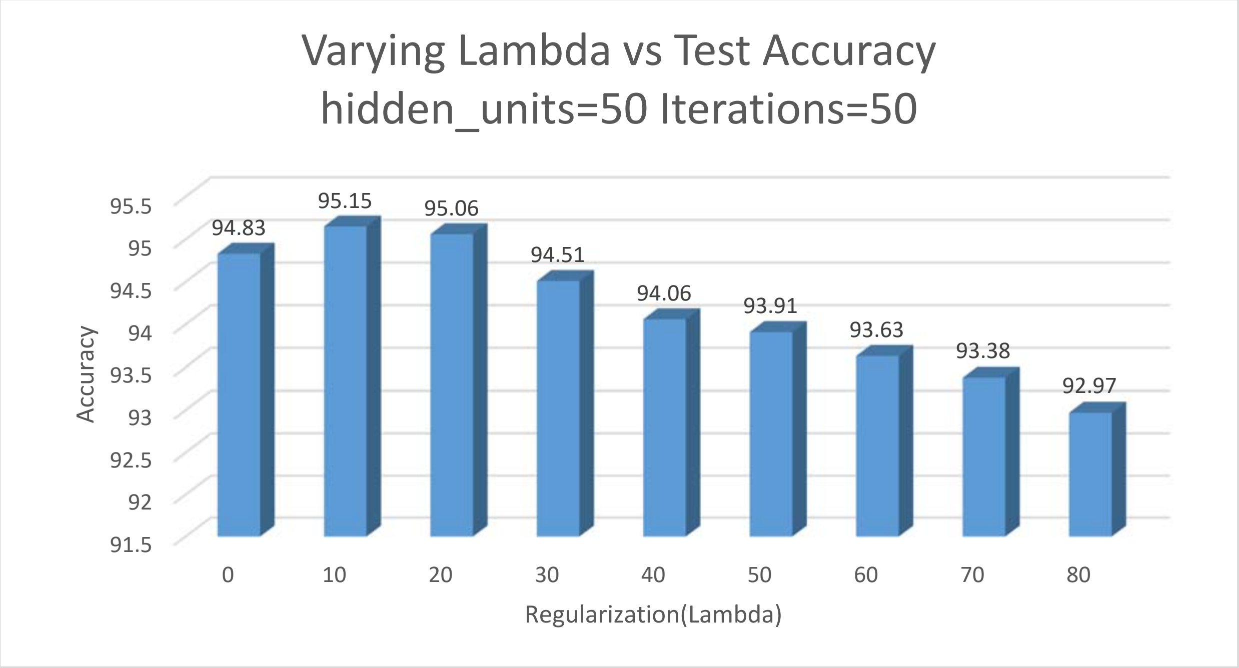
2.1.2. Increasing number of iterations to 100



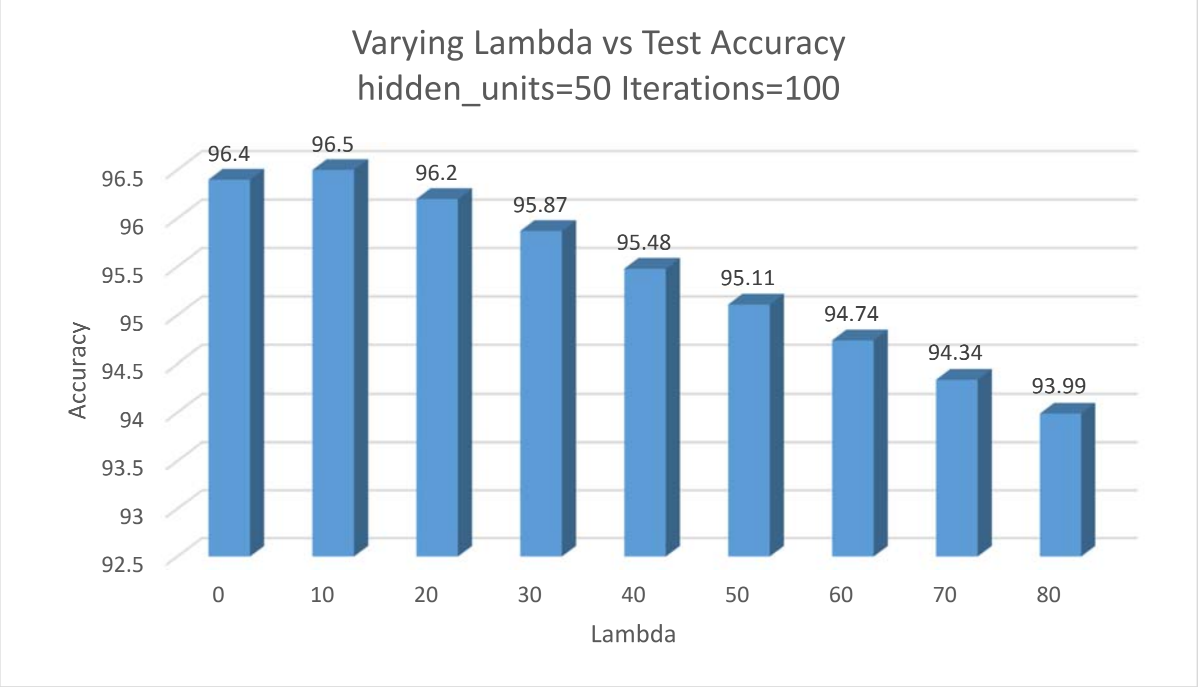
We increased the number of iterations to 100 to see if we see a similar pattern when varying lambda, which we were able to confirm from the results. With 100 iterations too, increasing the value of Lambda gradually led to a decrease in the Validation set accuracy.

2.2. Testing Data

2.2.1. Number of hidden units as 50 and number of iterations as 50

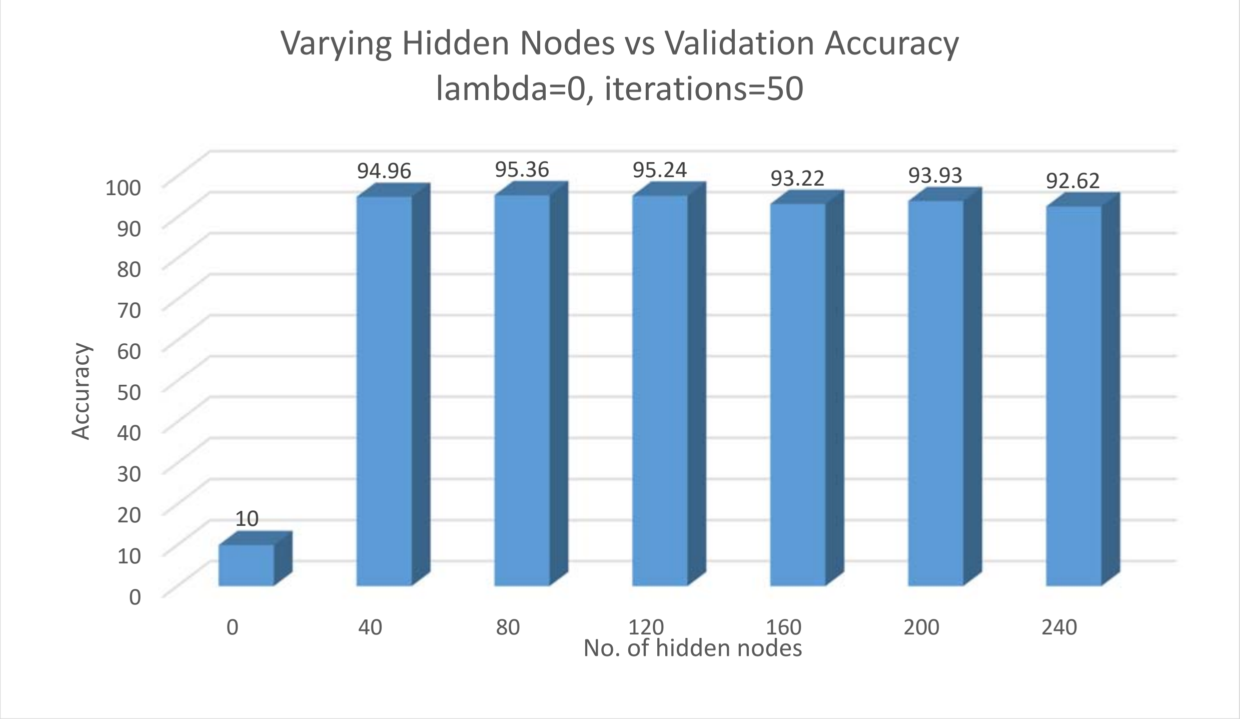


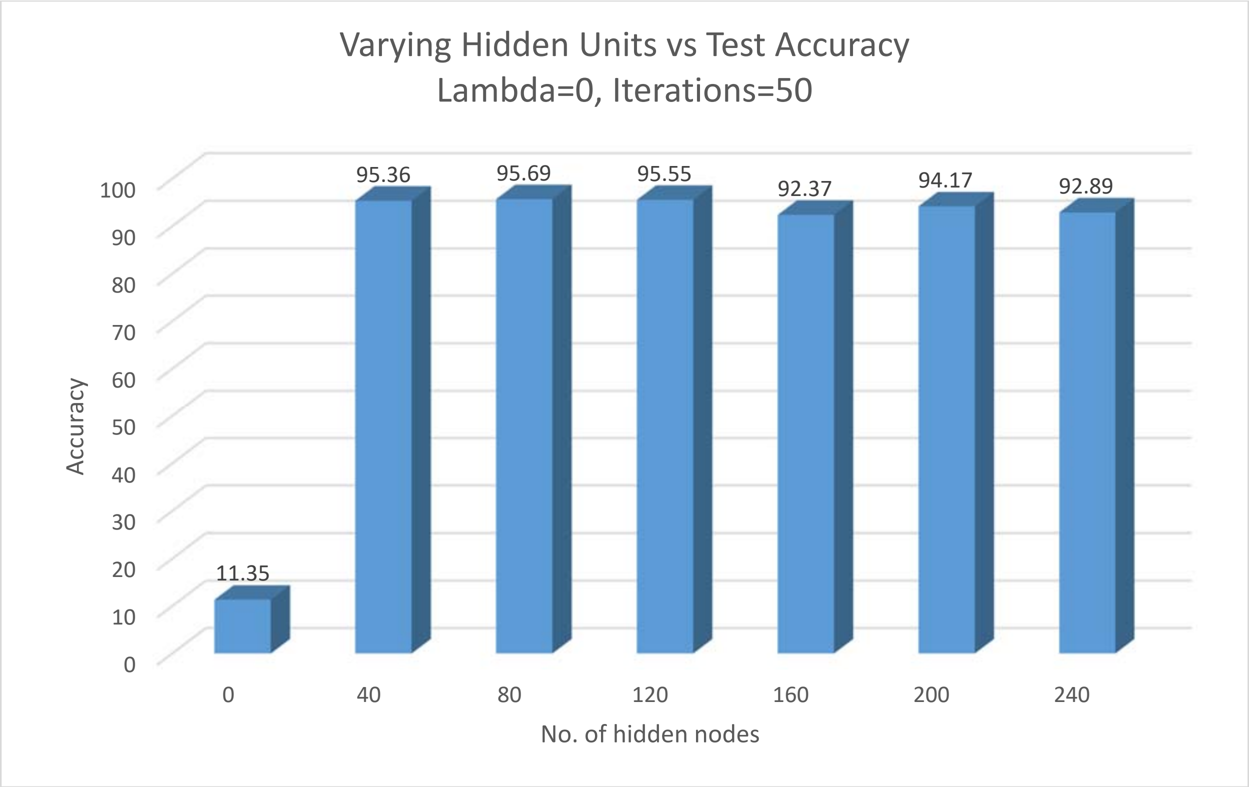
2.2.2. Increasing No. of Iterations to 100



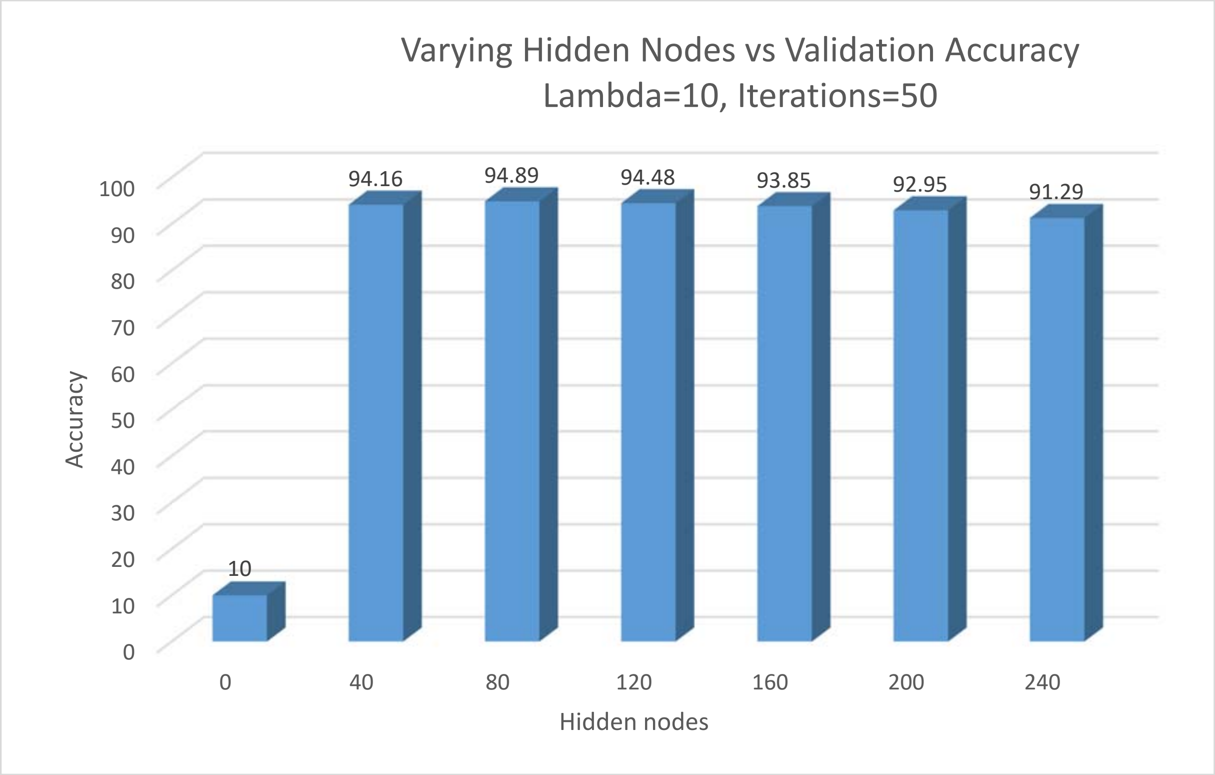
2.3. Changing hidden_units

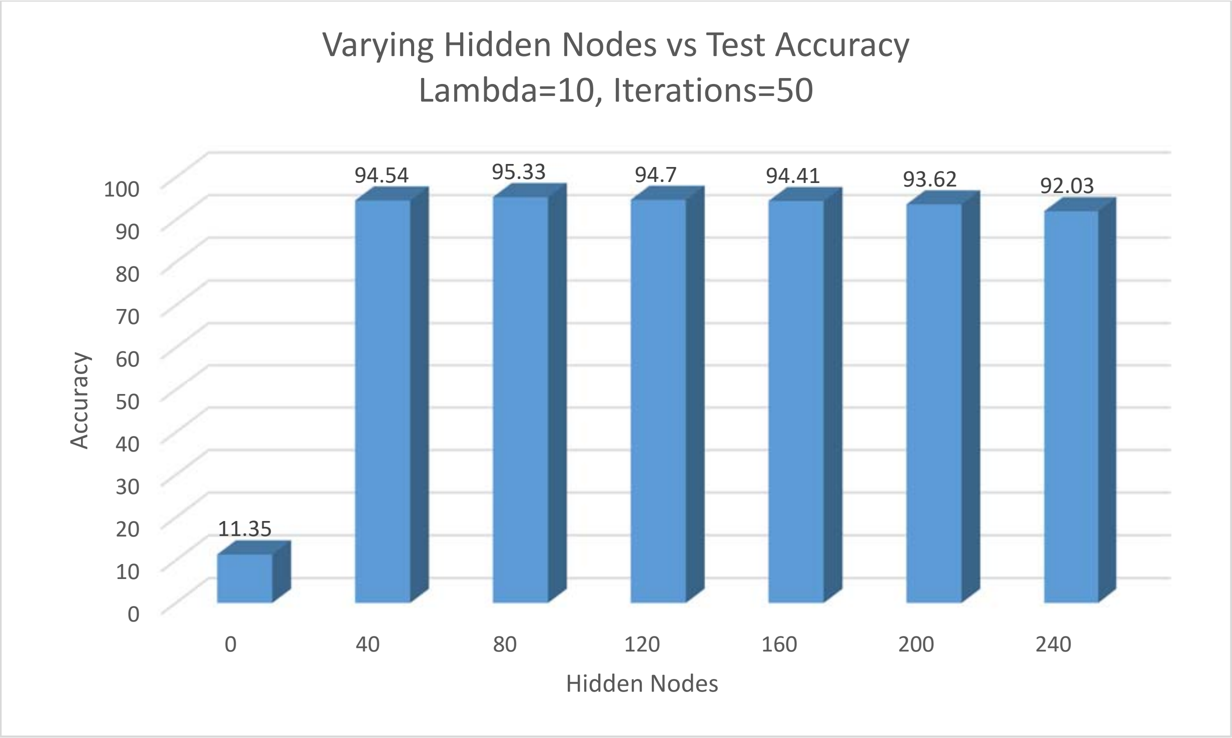
2.3.1. Lambda=0, Iterations=50





2.3.2. Lambda=10, Iterations=50





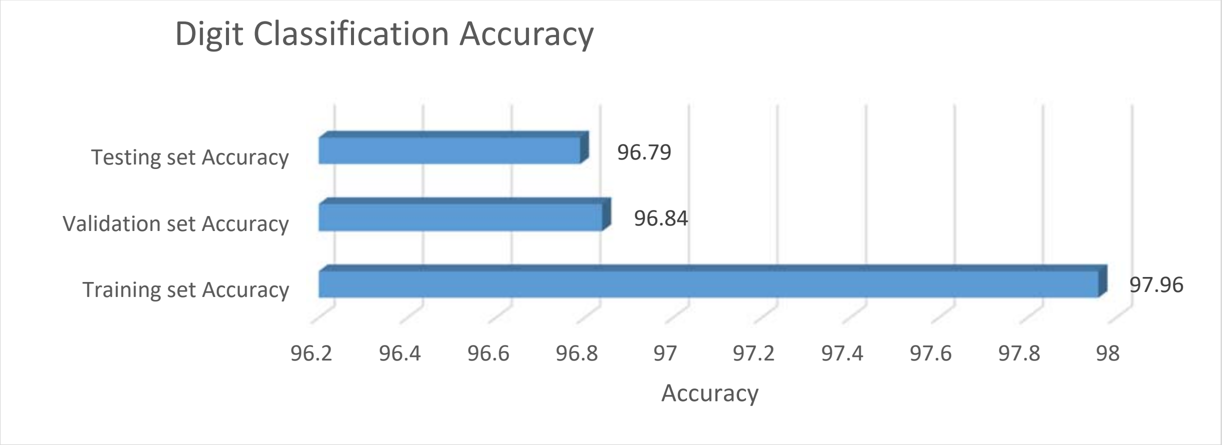
SELECTING HYPER-PARAMETERS BASED ON ABOVE TESTS

Based on the above results, we chose Lambda as 10, and the number of hidden units as 80. At 100 iterations, having even higher number of hidden units gives even better accuracy but hampers performance as training time is increased by a lot. Hence we came to the conclusion that we should choose the optimal lambda as 10, and optimal number of hidden units as 80.

3. ACCURACIES OF DIGIT CLASSIFICATION AND FACE RECOGNITION

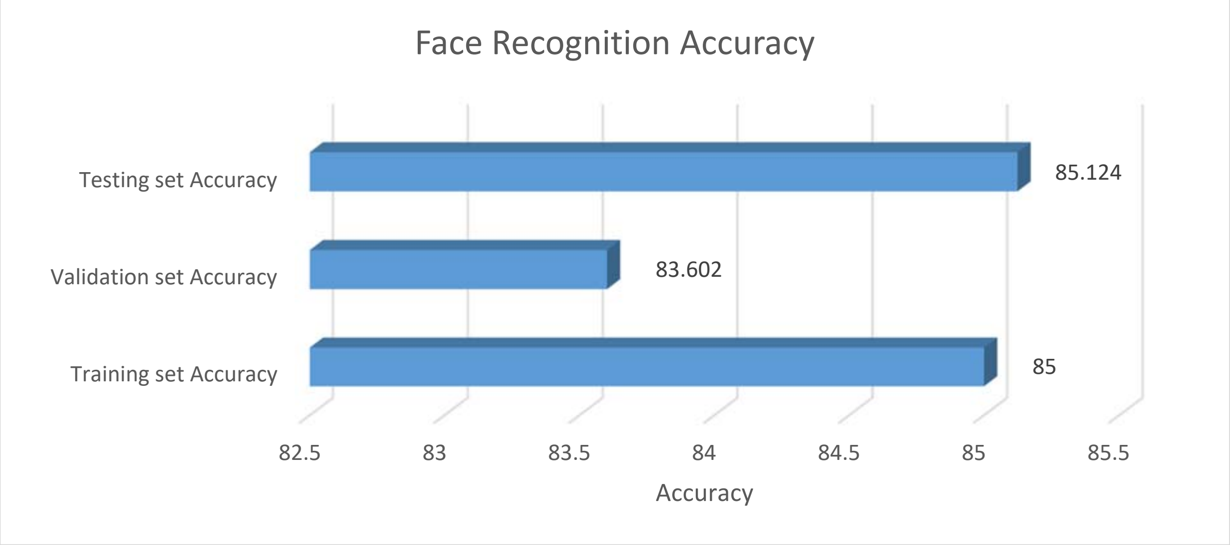
3.1. Digit Classification Accuracy

Based on the above tests the optimal hyper parameters are lambda=10, iterations=100, hidden nodes=80. After running the nnScript we got the following accuracy of 96.79.



3.2. CelebA dataset Accuracy

When we executed the facennScript we gort the accuracy as

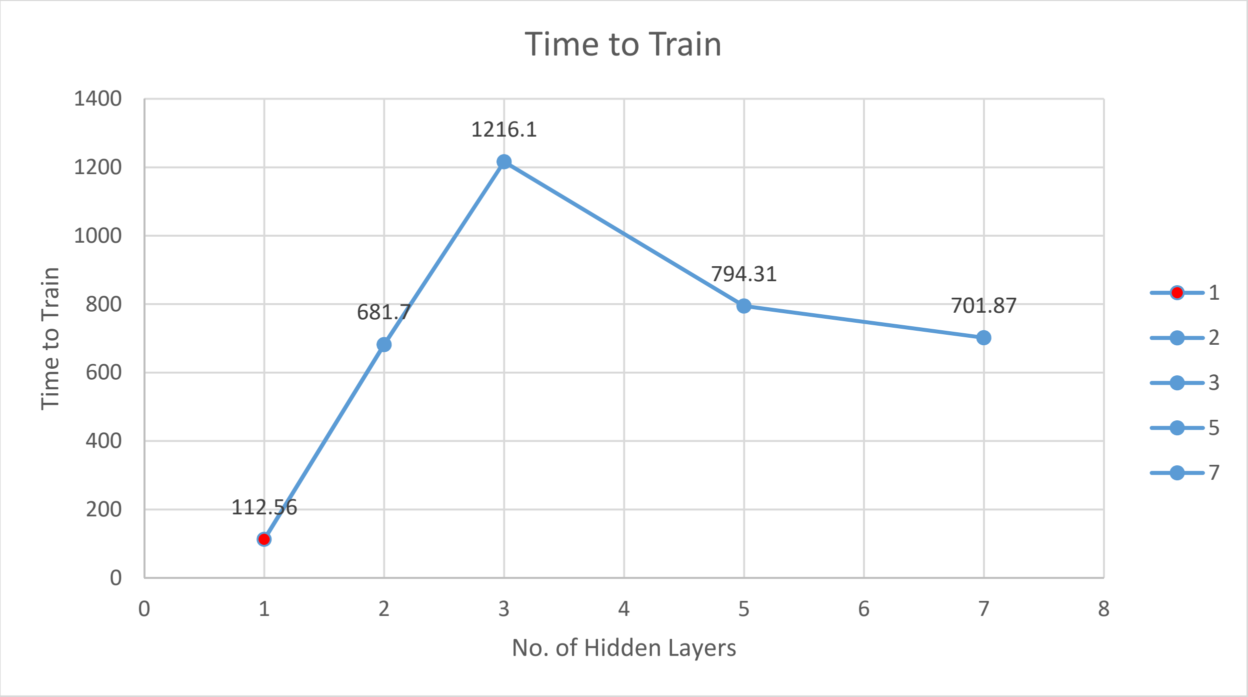


4. COMPARISON OF CelebA RESULTS WITH TENSORFLOW NEURAL NETWORK

Running the Deep Neural Network with 3, 5 and 7 hidden layers, we got the following results:

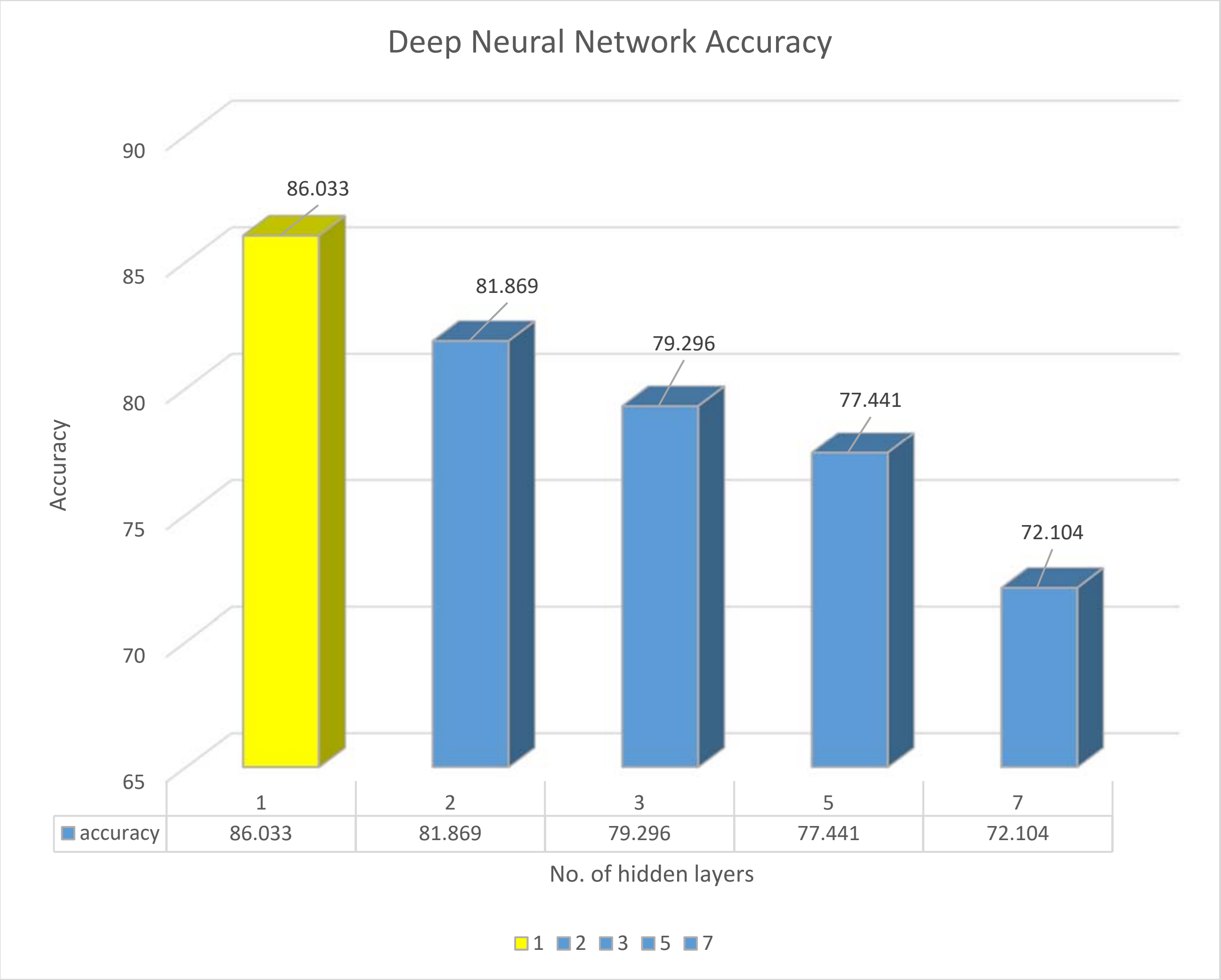
4.1. Training Time vs Hidden Layers

The first graph gives us the diffeence in training time with varying number of hidden layers. The scripts facennScript.py and deepnnScript.py were run on the UB CSE server Springsteen. We tried 2, 3, 5 and 7 hidden layers for the deep neural network. **The red point indicates the time taken by our neural network. A general observation was that the facennScript (neural network) ran faster than the deep neural network script.** Also, increasing the number of hidden layers increased the training time. Note that training time also depends on the available resources on the server at the time the script was being run, so in case a lot of users were running their respective jobs at the same time, this might have had some effect on the training time, but it would be safe to say that increasing the number of hidden layers would increase the training time.



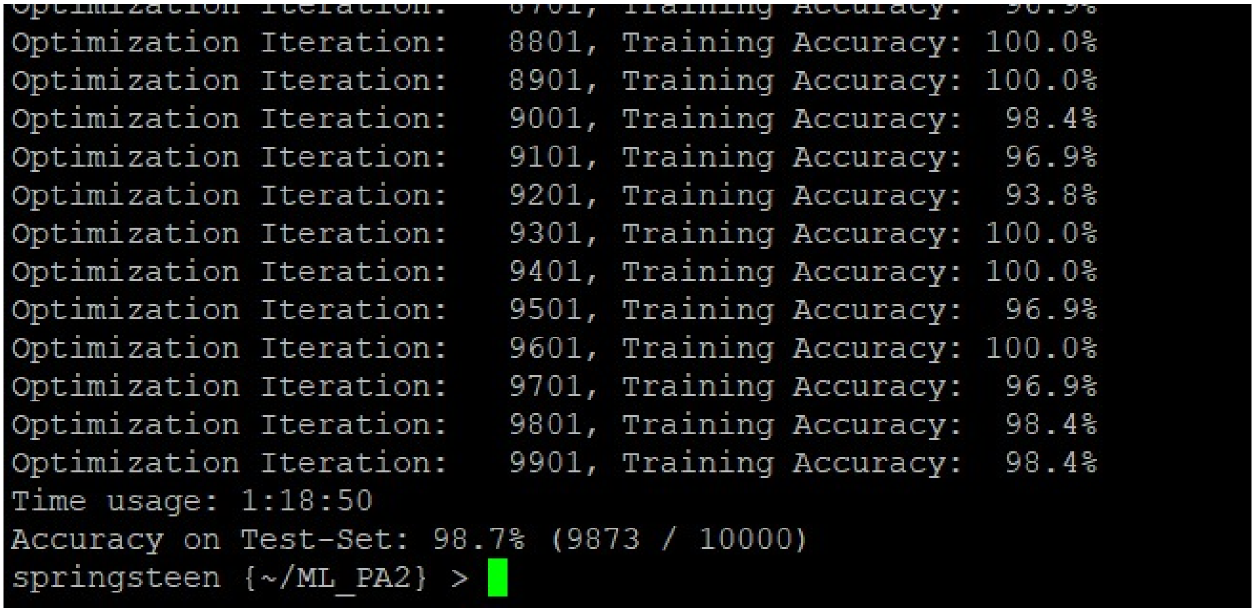
4.2. Accuracy vs Hidden Nodes

We see that increasing the number of hidden layers doesn't necessarily increase the accuracy of the deep neural network, in fact, we see a decrease when we test it with 7 hidden layers. Also, comparing the results between the Single Layer Neural Network and the TensorFlow Deep Neural Network, it is surprising to see that the accuracy of the single layer neural network comes out to be higher than that of a deep Neural Network. Perhaps this may change depending on other factors such as learning rate, number of iterations and number of hidden nodes used.



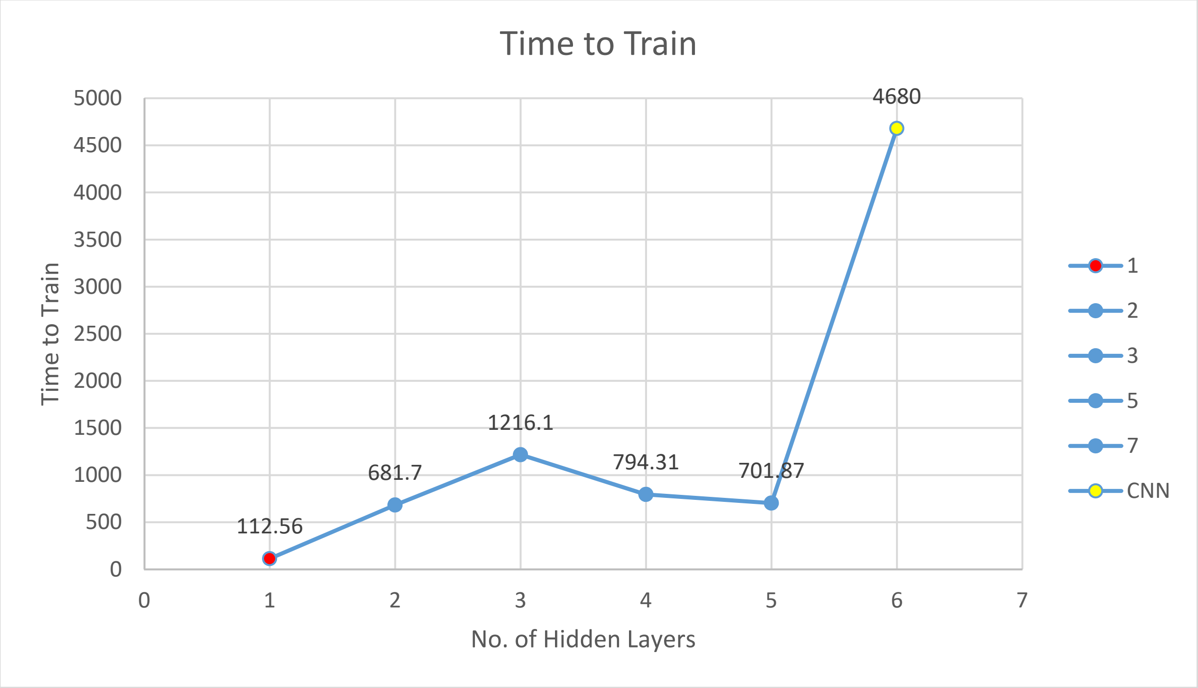
The Yellow Bar represents the accuracy for neural network wheras the remaining bars indicate the accuracies for the deep neural network. From the above graphical representation we can clearly see that **our neural network's accuracy is better than the deep neural network and also the accuracy was decreasing with the increase in number of hidden layers.**

5. COMPARING CNN WITH OTHERS (Extra Credit)



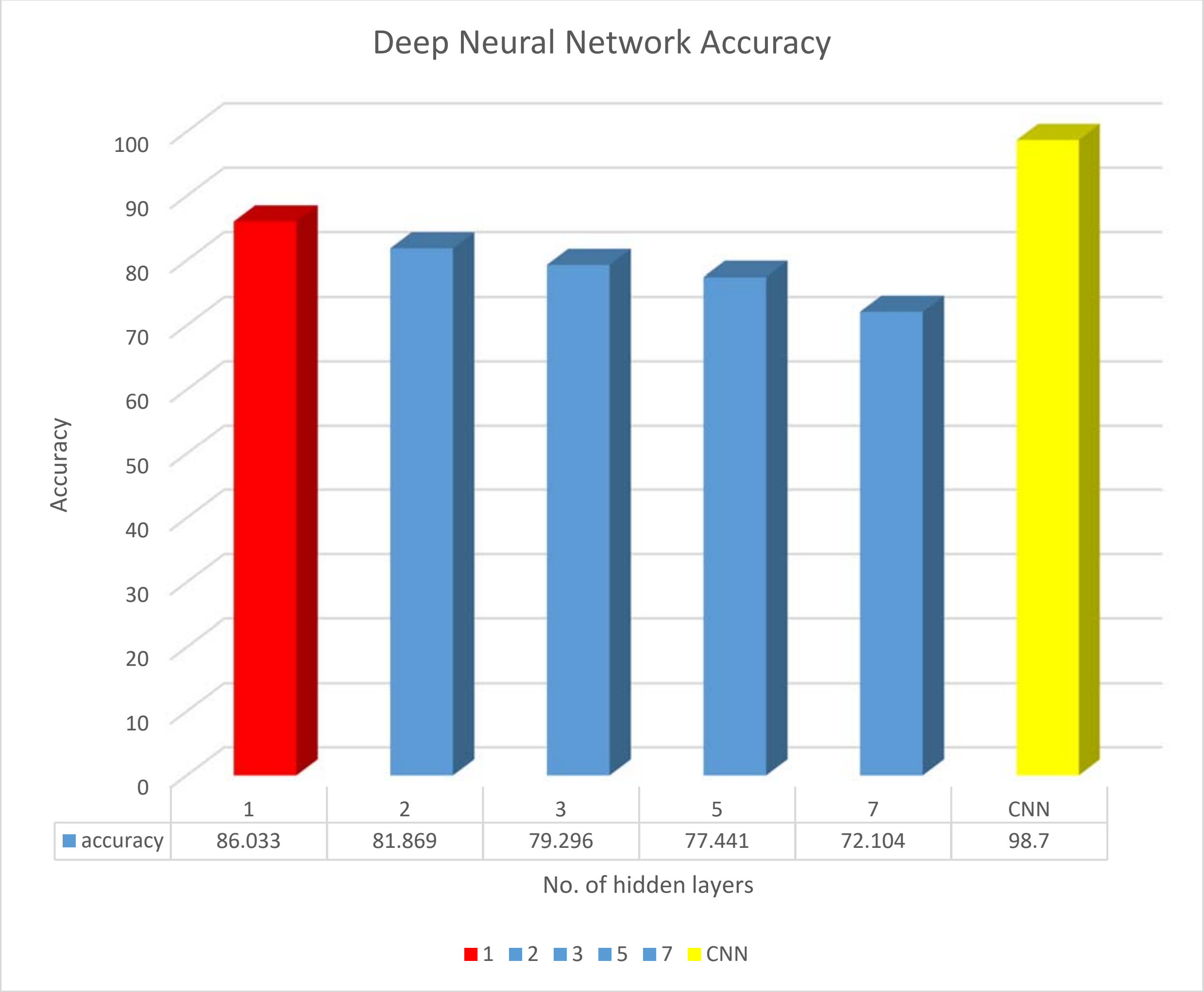
```
Optimization Iteration:   8701, Training Accuracy:  96.9%
Optimization Iteration:   8801, Training Accuracy: 100.0%
Optimization Iteration:   8901, Training Accuracy: 100.0%
Optimization Iteration:   9001, Training Accuracy:  98.4%
Optimization Iteration:   9101, Training Accuracy:  96.9%
Optimization Iteration:   9201, Training Accuracy:  93.8%
Optimization Iteration:   9301, Training Accuracy: 100.0%
Optimization Iteration:   9401, Training Accuracy: 100.0%
Optimization Iteration:   9501, Training Accuracy:  96.9%
Optimization Iteration:   9601, Training Accuracy: 100.0%
Optimization Iteration:   9701, Training Accuracy:  96.9%
Optimization Iteration:   9801, Training Accuracy:  98.4%
Optimization Iteration:   9901, Training Accuracy:  98.4%
Time usage: 1:18:50
Accuracy on Test-Set: 98.7% (9873 / 10000)
springsteen {~/ML_PA2} >
```

5.1. Comparing CNN training time with DEEPNN and Neural Network.



We can see that the Training time(yellow point) of CNN is very high compared to the DeepNN and neural network.

5.2. Comparing CNN accuracy with DEEPNN and Neural Network.



We can see that the Accuracy of CNN is pretty high when compared to the other DeepNN and Neural Network. Acuracy of CNN is 98.7.