CSE474/574 Introduction to Machine Learning
Programming Assignment 1 - Report
Handwritten Digits Classification Using Neural Networks

Assignment Group 33
Abhijeet Deshpande – 50168465
Mayur Tale - 50169256
Vishwaksen Mane – 50168863

Introduction:

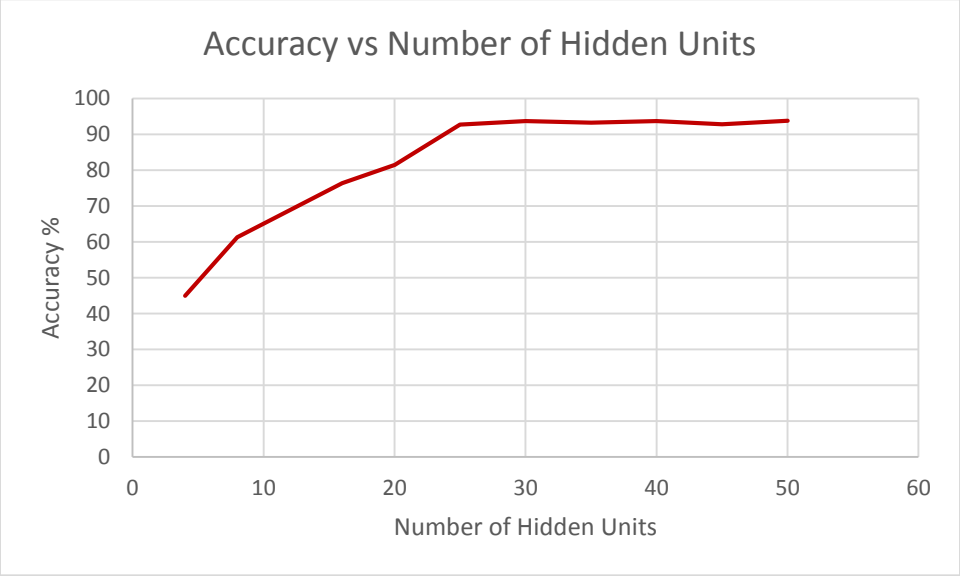
We have implemented multilayer perceptron neural network along with feed forward and back propagation and evaluated its performance in classifying handwritten digits. We also implemented regularization in the neural network so as to avoid the over fitting problem.

Performance Measures:

Accuracy: **94.04 %**
Number of Hidden Units: **50**
Regularization Factor λ**: 0.2**

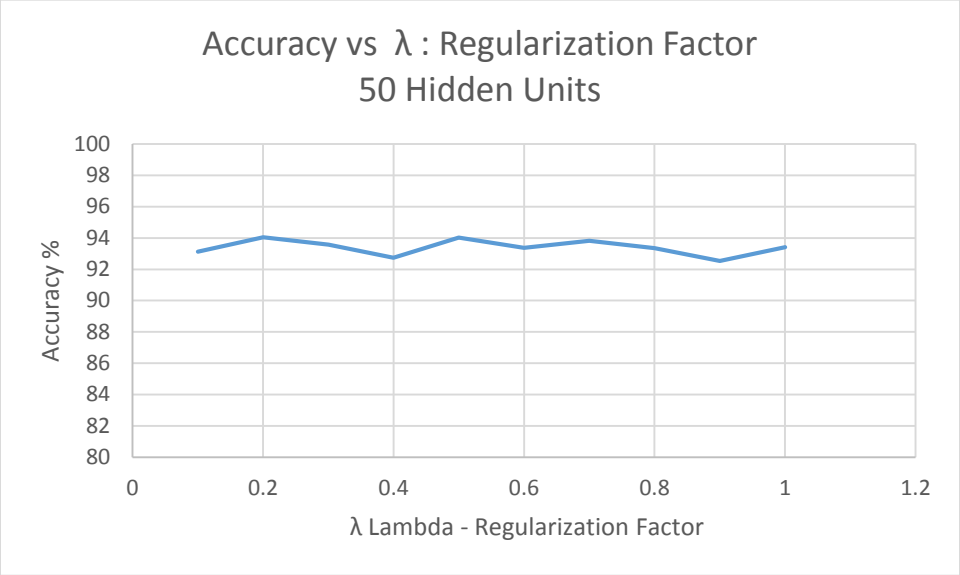
Graphical Analysis:

Accuracy vs Number of Hidden Units



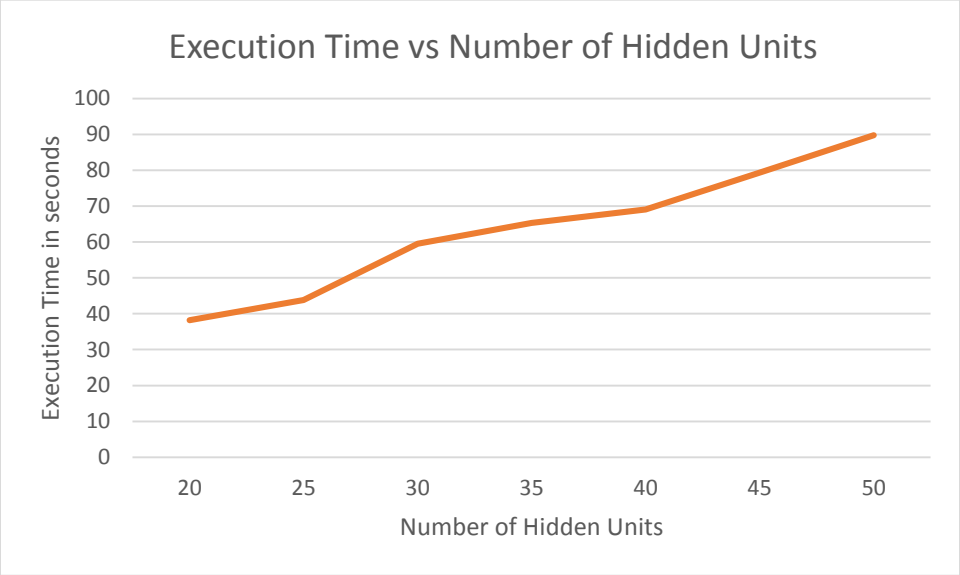
The above graph relates to the direct proportionality of accuracy with respect to the number of hidden units in the neural network.

Accuracy vs Regularization Factor λ where number of hidden units=50:



This plot describes the variation in the accuracy of our neural network in reference to λ – Regularization factor when number of hidden units is 50.

Execution Time vs Number of Hidden Units:



The above plot depicts direct proportionality between execution time and number of hidden unit.

Inference/Conclusion:

The number of hidden units is directly proportional to the accuracy of the neural network. i.e. increasing the number of hidden units led to increase in the accuracy and fine tune our prediction and thereby improving the efficiency of the neural network. We observed that increasing the number of hidden units from 16 to 50 led to increase in the accuracy from 76% to 94%. This significant increase in accuracy was a driving reason for us to set the number of hidden units to 50 where we achieved similar accuracy consistently.

Similarly, we also observed that the execution time is directly proportional to the number of hidden units in the neural network. This implies that as the number of hidden units increases, the execution time also increases due to the increase in the number of computations to be performed. We observed that increasing the number of hidden units from 30 to 50 resulted in increase in the execution time from 59 seconds to 90 seconds.

The Regularization factor λ can be used to avoid over-fitting problem so that the algorithm produces similar results on testing dataset. We experimented with different values for λ and observed that without regularization, the algorithm takes time to converge and learn weight vectors. Therefore, we experimented with different values of λ ranging between 0-1 and recorded the corresponding accuracy and performance of our neural network. Moreover, we observed very minor variation in accuracy against λ i.e. an accuracy of 93.13% for λ=0.1 and accuracy of 93.41% for λ=1. Thus we choose regularization factor **λ=0.2** which consistently gave us higher **accuracy of 94.04%** when number of **hidden units were 50**.