Programming Assignment 1
Handwritten Analysis
CSE 574

Group 23 - Members :

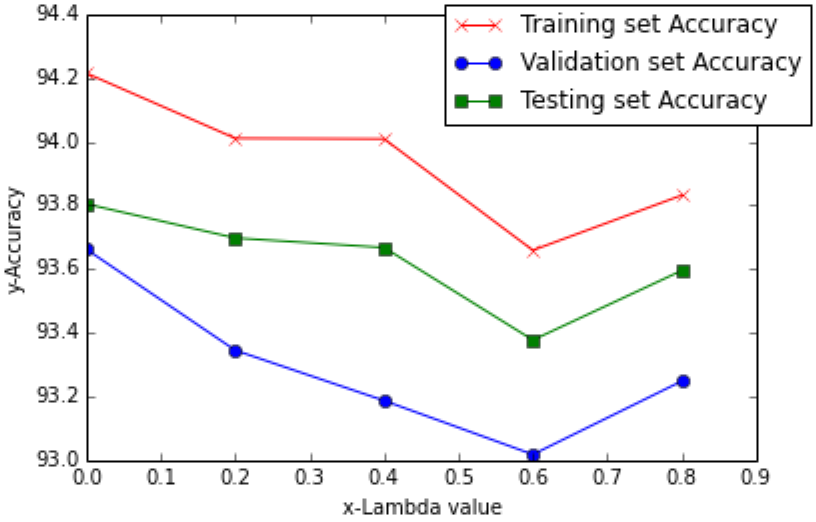
- Rakesh Balasubramanian
- Maira Saboia Da Silva
- Ramya Rao

Abstract :

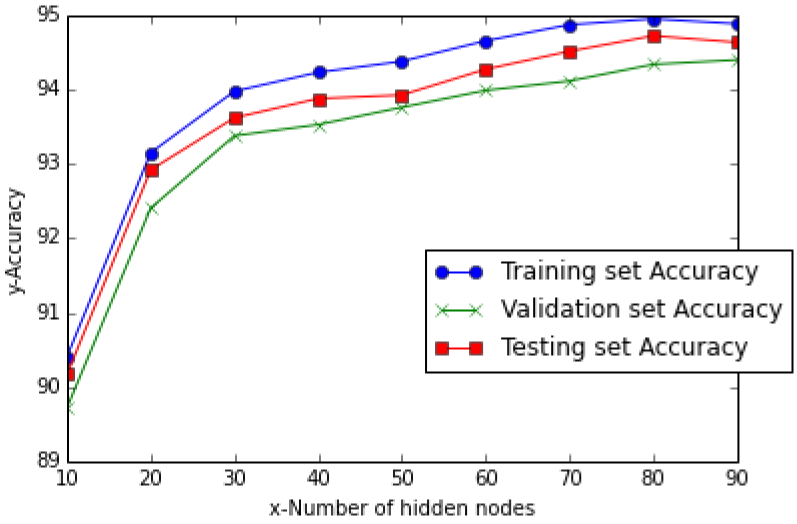
In this assignment, we have implemented a Multilayer Perceptron Neural Network to classify handwritten digits. In our preprocessing function, feature reduction effectively reduces 67 features from the training sample set. After implementing the feedforward pass with randomly initialized weights, we have normalized the error function by a factor λ and used it for back propagation calculations. Performance of the neural network for varying values of λ and no of hidden nodes have been observed in detail to check for optimal behavior.

Lambda vs. Accuracy :

The below graph shows the variation in accuracy of our implementation with variation in λ in the range 0 to 1 in steps of 0.2 (mean taken over hidden nodes). Ideally we expect increase in accuracy of testing data and decrease in accuracy of training data with increasing lambda value. However this is not observed in our implementation. This might be because there are not enough training samples.

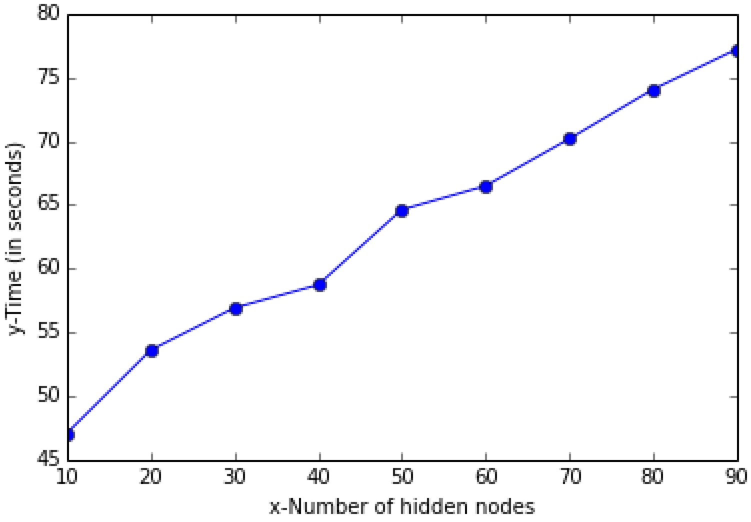


Hidden nodes vs Accuracy :



The accuracy of our implementation increases as the number of hidden nodes increases. This can be attributed to the fact that the hidden nodes represent the number of learned features. We also see that after a certain value of hidden nodes, the accuracy does not increase anymore. Beyond this point, accuracy cannot increase as the number of features to learn is a fixed value.

Hidden nodes vs Time



We see here that as we increase the number of hidden nodes, the learning time of the neural network increases. The number of weight updations increase with each additional node, and this value is being computed several times before arriving at optimal weights. Hence, more hidden units implies more complexity and therefore more time.

Best Observed Output of Our Implementation:

We have tried to avoid for-loops as much as possible and used vectorized dot products for most calculations. This code optimization led us to a much better learning time.
Based on observation we have :
λ = 0.4
No. of hidden units = 80
Time taken for computation = 1 min 3 sec
Training Data Accuracy = 95.658 %
Validation Data Accuracy = 94.95 %
Test Data Accuracy = 95.35 %

System specs used for testing :
Processor make: Intel core i7 quad core
processor speed: 2Ghz
RAM : 8 GB