CSE-574
Programming Assignment Report-Team 34

Team Members:
Sagar Bansilal Shinde (UB#: 50134022)
Madni Vadhariya (UB#: 50098070)
Mohammad Shahid Fazal (UB#: 50133053)

Program Flow:

1) The preprocessor function loads the mat file provided and extracts the test files and the train files. It further divides the train data into validation data and the training data. Validation labels, Test Labels, Train Labels are created based on each digit. It also performs the normalization of the test data, train data and validation data.

2) Number of input nodes, hidden nodes and output nodes are initialized and the nnObjFunction is then called to perform machine learning operations like feed forward and back propagation on the different kinds of data in order to train the Neural Network to achieve better accuracy at predicting labels.

3) The nnPredict() function is called with weights w1(the weight vector between input layer and the hidden layer) and w2(the weight vector between hidden layer and output layer) as arguments alongside training data ,validation data and test data as arguments in separate passes. The nnPredict() function predicts a label based on the information provided.

4) The algorithm culminates in the neural network achieving better accuracy for predicting digit labels in a short time of **little less than 2 minutes**.

Table with lambda=0.5, tested with different number of hidden nodes over different datasets

Number of Hidden Nodes	Accuracy of Training Data	Accuracy of Validation Data	Accuracy of Test data
4	49.6	48.69	49.41
8	89.342	88.17	89.29
12	92.44	91.43	91.97
16	92.92	92.2	92.6
20	93.16	92.1	92.68
50	94.53	93.56	94.08
100	95.1	94.35	94.66

Graph 1: Number of Hidden Nodes (n_hidden) Versus Accuracy

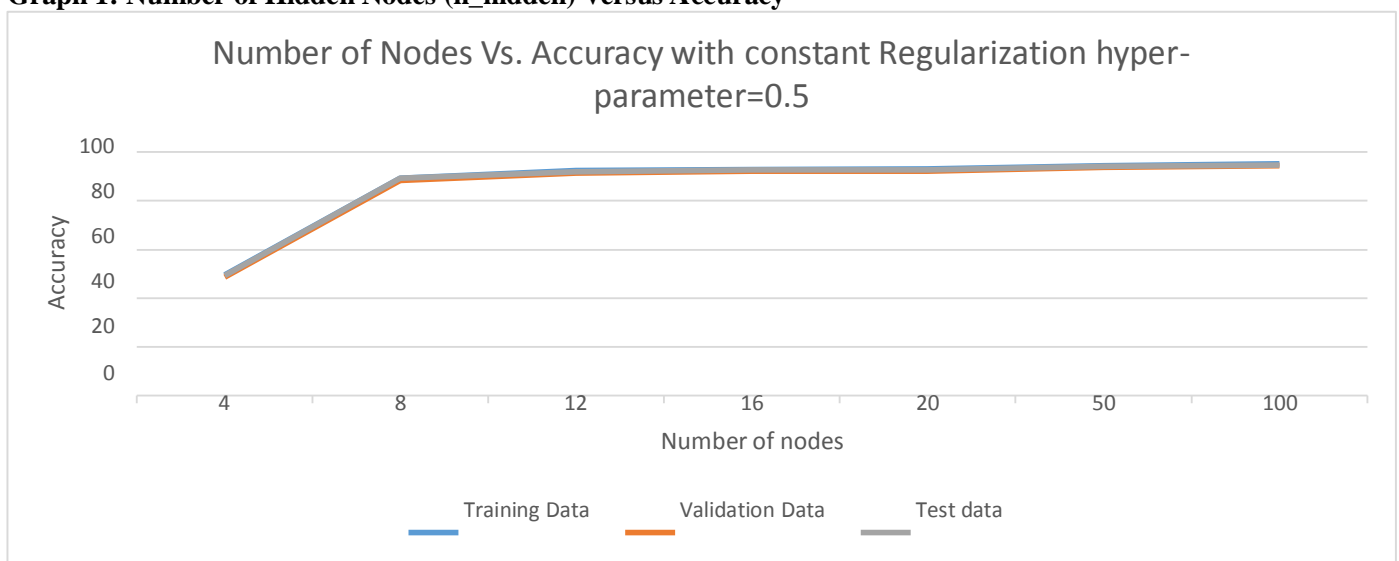Number of Nodes Vs. Accuracy with constant Regularization hyper-parameter=0.5

Table with number of hidden nodes=50, tested with different lambda(λ) over different datasets

LamdaValue λ	Accuracy of Training data	Accuracy of Validation Data	Accuracy of Test Data
0.01	94.25	93.67	94.06
0.1	94.12	93.39	93.8
1	95.36	94.34	94.73
10	94.01	93.66	93.64

Graph 2: Lamdavalue(λ) versus Accuracy
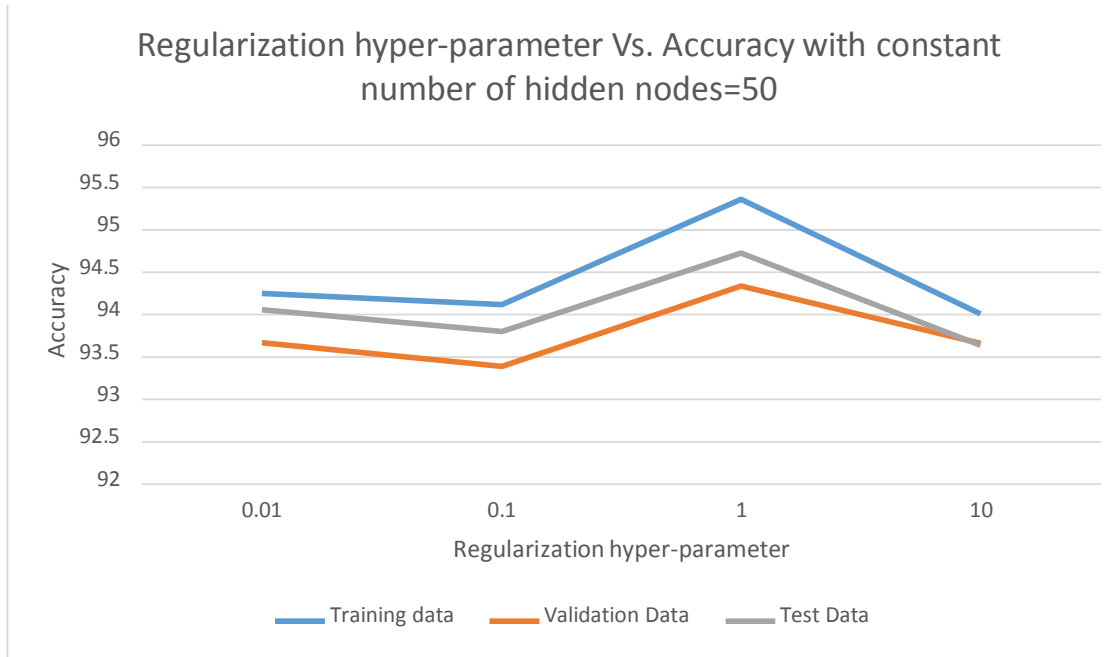
From graph 1 it is apparent that increasing number of nodes in hidden layer increases significantly until some saturation point beyond which increasing the number of nodes doesn't improve the accuracy anymore. From Graph 2: we deduce that for Lambdavalue(λ) = 1 we get the most accuracy. The whole program trains the weight matrices, predicts the output of all the three datasets and calculates the accuracy in less than 2 minutes. Training data has high accuracy is due to changing weights based on prediction of training data and for test data weights are not modified for wrong prediction. From our results on different λ and n_hidden values we find that accuracy is highest at λ = 1 and n_hidden(number of nodes in the hidden layer) = 50, which is 94.73%. To get the best accuracy, increase the number of nodes in hidden layer upto the saturation point which lets network learn the concept more accurately. LambdaValue λ should be chosen in such a manner that we get best accuracy for the test_data and at the same time not big to cause underfitting problem and not too small to cause overfitting problem and in this assignment after running analysis and data gathered for various values of λ we learn that for this problem λ = 1 gets most accurate results.