Machine Learning

Project 2

SagarVishwakarma

Per. No.: 50098079

sagarvis@buffalo.edu

Contents

Problem Statement

This project is to implement and evaluate classification algorithms. You will implement two Different methods of classification and compare their performance with a publicly available Package.

Theory

The above derivation of the backpropagation procedure allowed for general forms for the error function, the activation functions, and the network topology. In order to illustrate the application of this algorithm, we shall consider a particular example. This is chosen both for its simplicity and for its practical importance, because many applications of neural networks reported in the literature make use of this type of network. Specifically, we shall consider a two-layer network of the form illustrated in Figure 5.1, together with a sum-of-squares error, in which the output units have linear activation functions, so that $y_k = a_k$, while the hidden units have logistic sigmoid activation functions given by

$$h(a) \equiv \tanh(a) \tag{5.58}$$

where

$$\tanh(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}}. \tag{5.59}$$

A useful feature of this function is that its derivative can be expressed in a particularly simple form:

$$h'(a) = 1 - h(a)^2. \tag{5.60}$$

We also consider a standard sum-of-squares error function, so that for pattern n the error is given by

$$E_n = \frac{1}{2}\sum_{k=1}^{K}(y_k - t_k)^2 \tag{5.61}$$

where y_k is the activation of output unit k, and t_k is the corresponding target, for a particular input pattern $\mathbf{x}_n$.

For each pattern in the training set in turn, we first perform a forward propagation using

$$a_j = \sum_{i=0}^{D} w_{ji}^{(1)} x_i \tag{5.62}$$

$$z_j = \tanh(a_j) \tag{5.63}$$

$$y_k = \sum_{j=0}^{M} w_{kj}^{(2)} z_j. \tag{5.64}$$

Implementation of logistic Regression

Importing the Dataset

The whole dataset in imported in an excel file.
The dimension of the file obtained is thus 19978 by 513.

Training

With **n= 0.0005,** a set of parameters w are obtained and with that w again wnew is calculated. The whole process in ran in a loop which breaks **at crossentropyerror=300.**

Testing

The optimal obtained is passed into testing dataset. The error and accuracy is calculated.

Implementation of Neural Networks

Importing the Dataset

The whole dataset in imported in an excel file.
The dimension of the file obtained is thus 19978 by 513.

Training

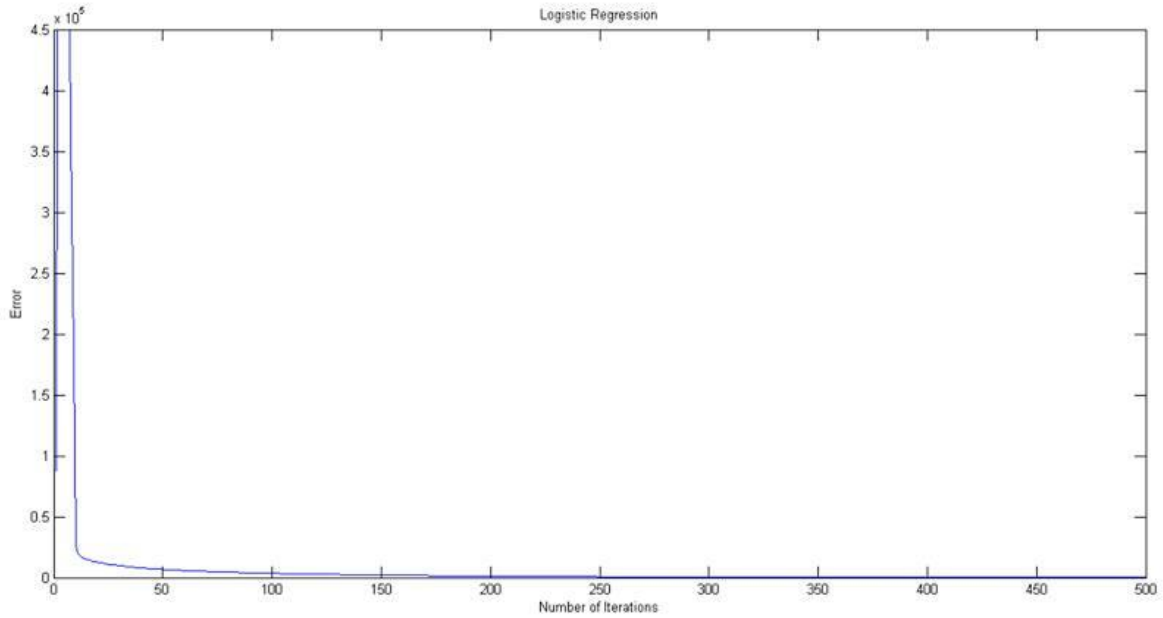
With **n1= 0.0003 and n2=0.0006,** a set of parameters w1 and w2 are obtained and with that again wnew1 and wnew2 is calculated. The whole process in ran in a loop which breaks **at crossentropyerror=500.**

Testing

The optimal obtained is passed into testing dataset. The error and accuracy is calculated.

Observations

The following graph is obtained.



Here, the optimum values of M and Lambda are obtained are:

N = 0.0005

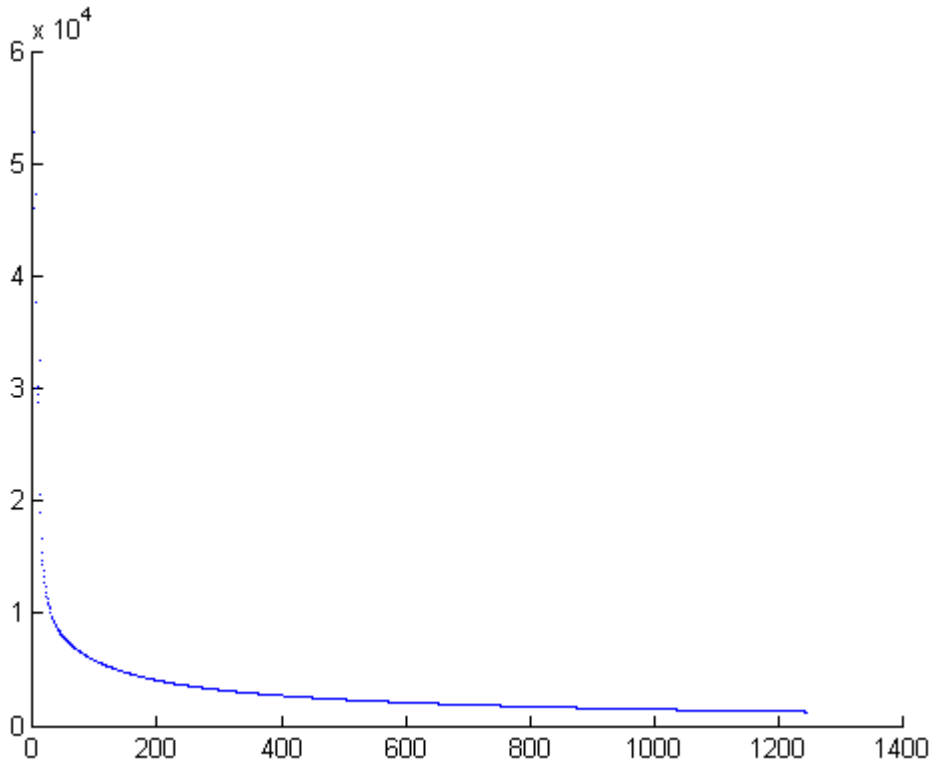
M = 25

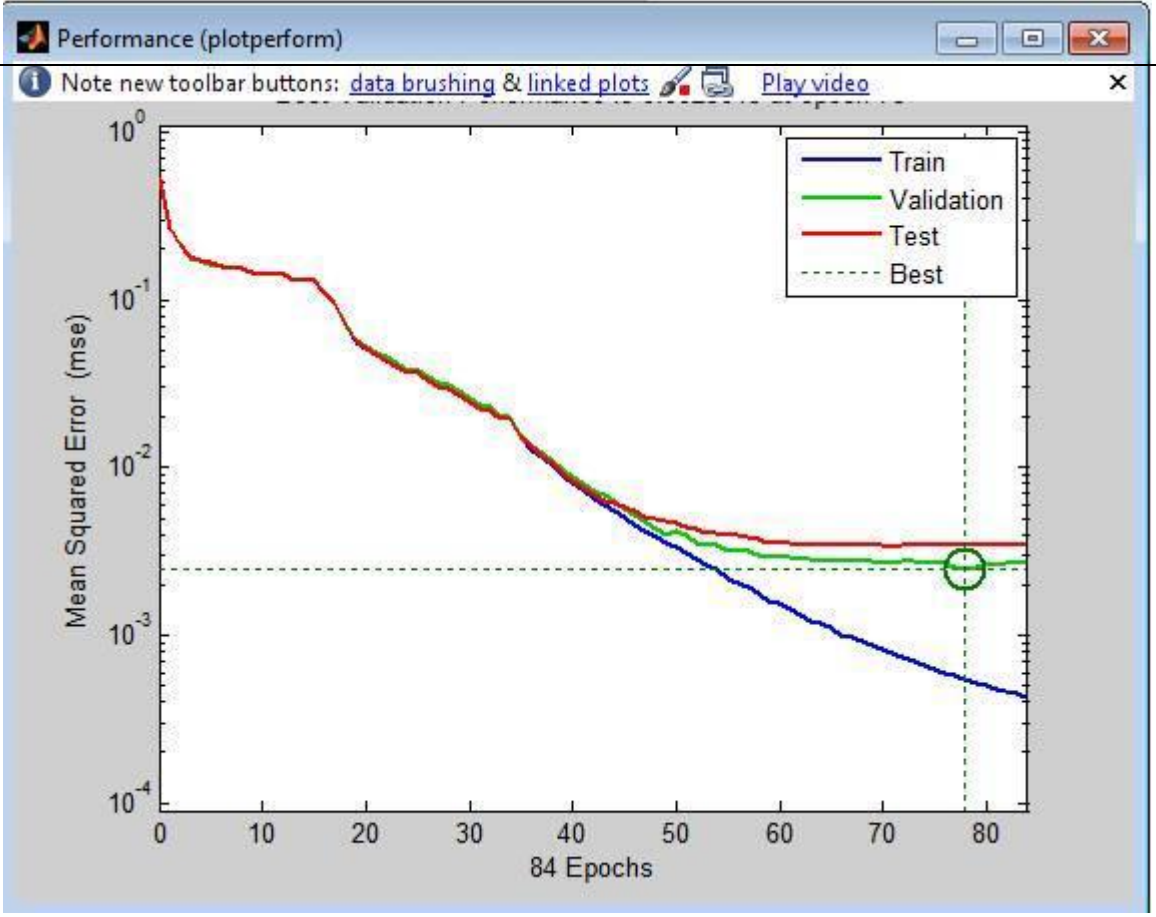
Using these values,

The cross entropy error obtained = 300

The Error rate obtained in testing is = 2.53 %

The following graph is obtained:


x 10^4
6
5
4
3
2
1
0
0
200
400
600
800
1000
1200
1400



Here, the optimum values obtained are:

N1 = 0.0003
N2 = 0.0006

M = 25

Using these values,

The cross entropy error obtained = 300

The Error rate obtained in testing is = 3.6 %

For Neural networks, the Succesrate obtained for testing data set = 99.7 %

Conclusion

The Logistic regression network has a good rate as compared to neural network.

The package offers a very good accuracy as compared to our project as we have terminated the loop at a point.