A Project Report

On

Artificial Neural Networks

BY

AMAN BADJATE	2017B3A70559H
GARVIT SONI	2017B3A70458H
KSHITIJ VERMA	2017B1A71145H

UNDER THE SUPERVISION OF

DR. BHANU MURTHY

SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS OF

BITS F464: MACHINE LEARNING

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI (RAJASTHAN)

HYDERABAD CAMPUS

APRIL, 2021

Contents

List of Figures

1 Introduction

Neural Network is a network of interconnected “neurons” in a similar way as our brain. The neurons are arranged in layers which are connected by edges having some weights. An example of a 2 layer neural network is shown below.

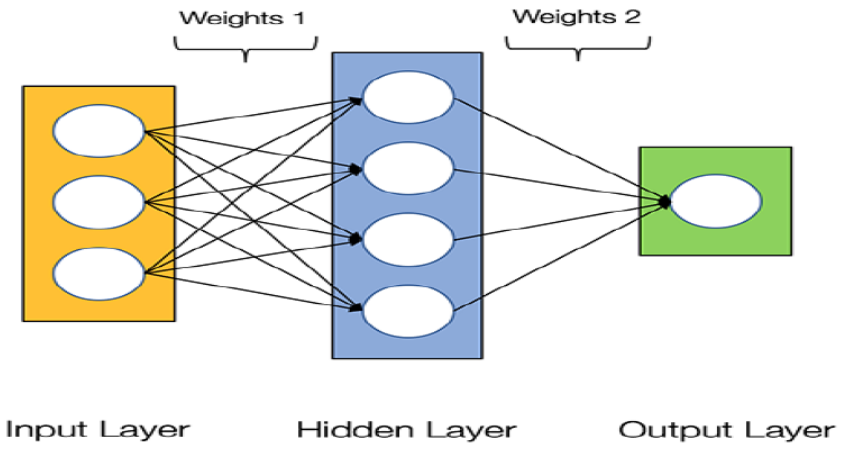

Figure 1: Simple ANN model

For the training process there are 2 steps: Feed Forward and Backward Propagation. In Feed Forward the inputs are multiplied by various weights and added to biases and pushed forward layer by layer to finally the output layer.

In Back Propagation the change in error is calculated with respect to the weights and biases and those weights and biases are adjusted so as to decrease the error. This is done by gradient descent.

These two processes are repeated several times to finally fit our model good enough.

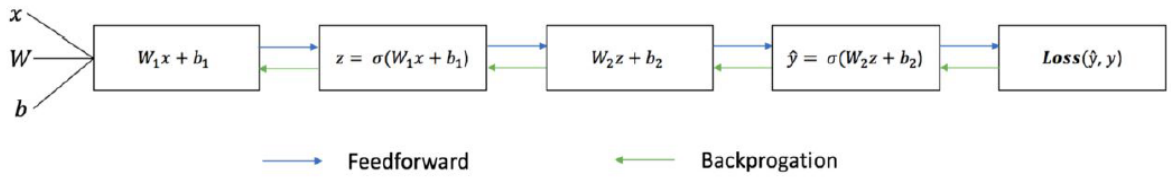

Figure 2: Feed Forward and back propagation

2 Design Decisions

1. Firstly the data was scaled as the attributes were of varying scales. Z-score normalization was used to scale the data so that their mean and variance are the same.
   $z = (x - \mu) / \sigma$

2. Train and test data was split in 70 - 30 ratio uniformly randomly so as to avoid any pre-bias in the data.

3. The number of neurons in each layer was chosen around two-thirds of the previous layer as mentioned because choosing too few or too many neurons can lead to underfitting and overfitting respectively.

4. The number of epochs was kept short to avoid the model to overfit to the train-data and to remain generalized.

3 Comparison of Different Learning rates

We have experimented with different learning rates. We have plotted accuracy versus epochs for 3 different learning rates for both 1 hidden layer model and 2 hidden layer model.
Results are shown below for each learning rate

3.1 Small Learning Rate

We have plotted the graphs for error and accuracy vs epochs for learning rate = 0.000001.

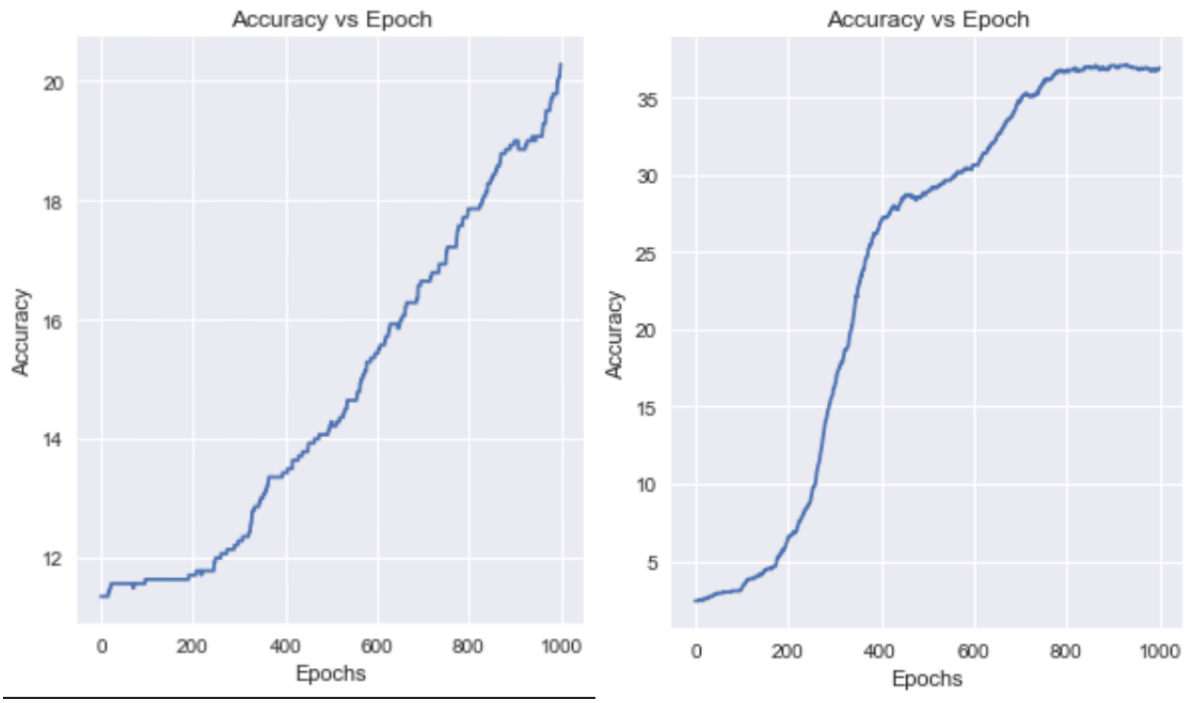

Figure 3: Accuracy vs Epoch for both 1 and 2 hidden for small eta

3.2 Large Learning Rate

We have plotted the graphs for error and accuracy vs epochs for learning rate = 5.

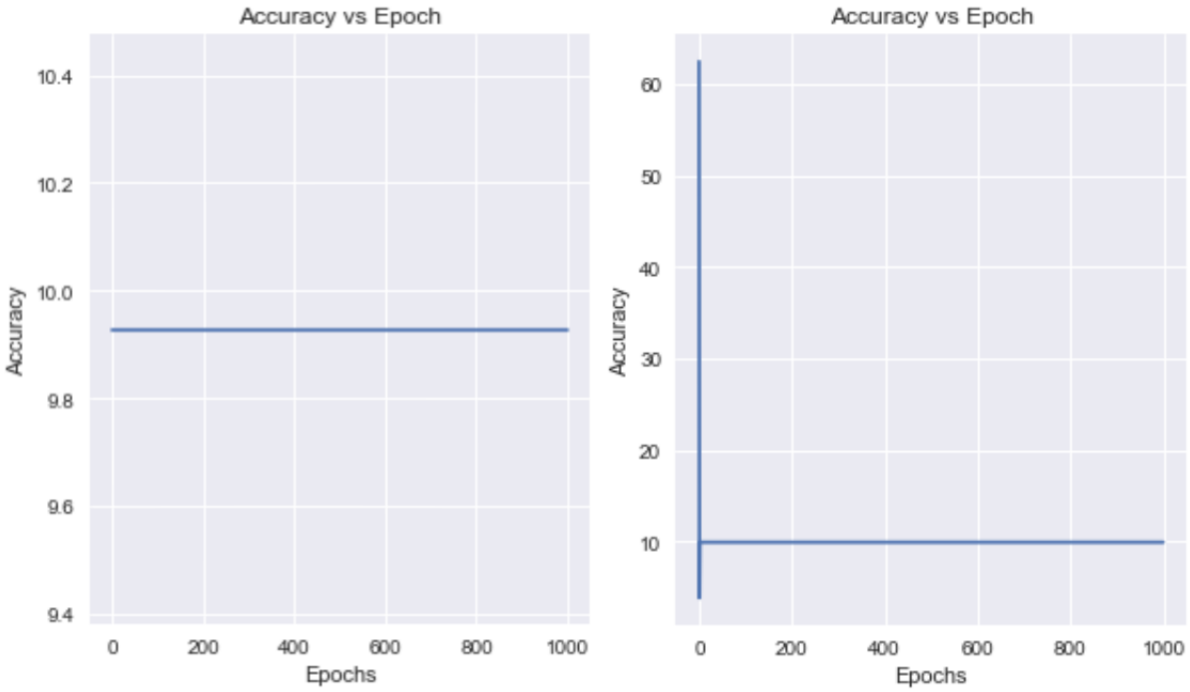

Figure 4: Accuracy vs Epoch for both 1 and 2 hidden for large eta

3.3 Optimal Learning Rate

We have plotted the graphs for error and accuracy vs epochs for learning rate = 0.001.

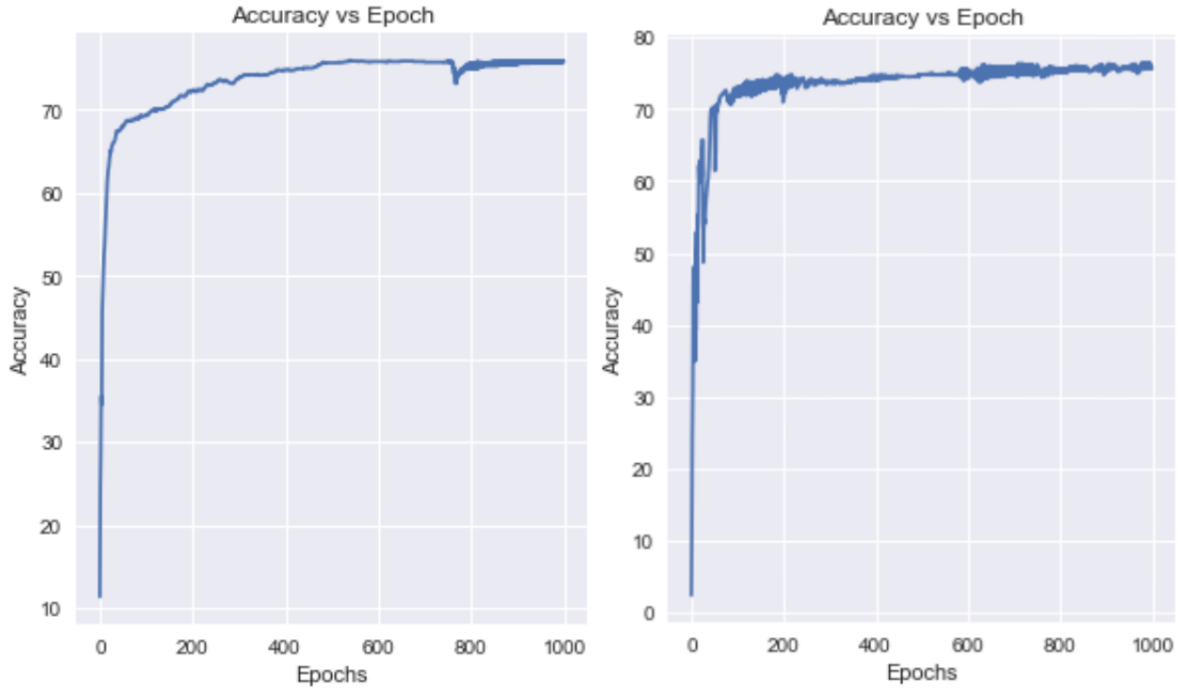

Figure 5: Accuracy vs Epoch for both 1 and 2 hidden for Optimal eta

4 Results

4.1 For 1 hidden Layer

Number of neurons in the hidden layer = 6

Activation Function for hidden layer = tanh

Optimal eta = 0.001

Training Accuracy: 75.78

Testing Accuracy: 71.33

Since graph for accuracy is already shown in the above section (optimal case), we will show the loss vs epoch graph here.

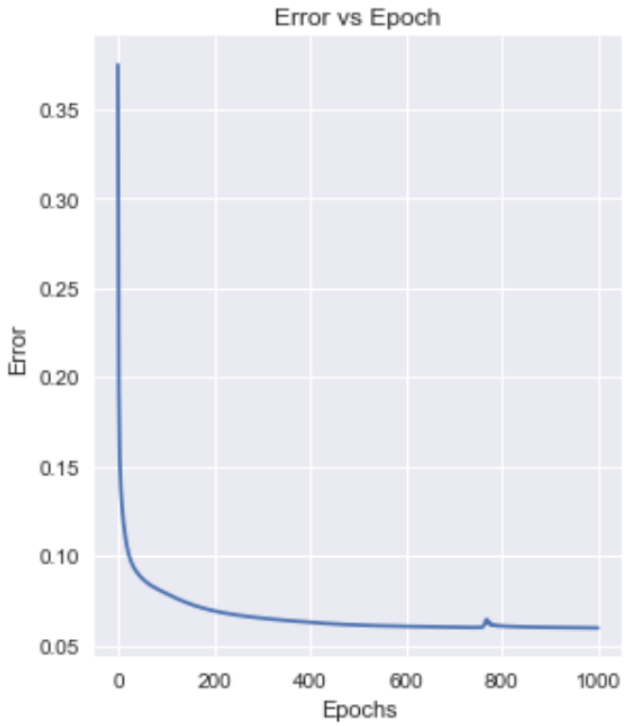

Figure 6: Training vs Epoch for 1 hidden layer

4.2 For 2 hidden Layer

Number of neurons in the hidden layers = (8,6)

Activation Function for hidden layers = tanh, tanh

Optimal eta = 0.001

Training Accuracy: 75.42

Testing Accuracy: 73.16

Since graph for accuracy is already shown in the above section (optimal case), we will show the loss vs epoch graph here.

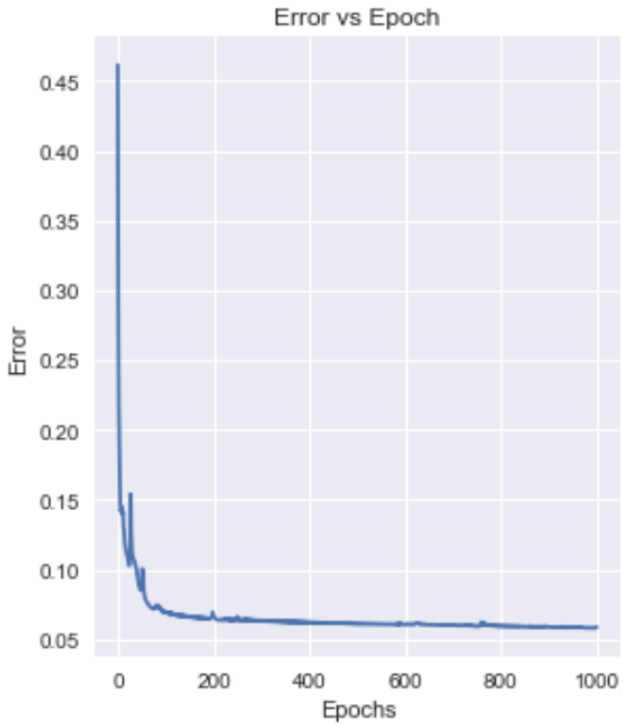

Figure 7: Training vs Epoch for 2 hidden layer

5 Conclusions

1. By seeing all graph, it is clear that learning rate should be not so high and not so low so that results are accurate.

2. The overall best accuracy achieved by the neural net model was 73.16% for 2 hidden layer model.

3. The number of neurons in each layer was chosen around two-thirds of the previous layer as mentioned because choosing too few or too many neurons can lead to underfitting and overfitting respectively.

4. The number of epochs was kept short to avoid the model to overfit to the train-data and to remain generalized.