International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Prediction of Pass Rate Using Neural Networks

Dr.T.Arun Kumar

Associate Professor of Mathematics,
Department of Mathematics, Government Degree College (Sciences), Adilabad 504001

Abstract:

In this paper, neural networks applied to predict the pass rate (percentage) for a particular examination of a particular student based on some auxiliary information. The data of 500 students is divided into three sets first two sets are used to training and validation of the neural networks and third set is used for testing. Training and validation is continued until the parameters of neural network are generalized. Then this network is applied on testing set to predict the pass rates. This data is analyzed through the feedforward neural networks and it presents promising results on prediction of pass rate. The prediction efficiency is measured by computing mean square error (MSE), mean absolute error (MAE) and mean absolute percentage error (MAPE) for the testing set.

Key words: neuron, neural networks, training, validation and testing.

1. Introduction

In this paper, neural network applied to predict the pass rate of a student based on the auxiliary information on number of assignments submitted, mock examinations marks and whether the student had attended tutorials or not. There are some autoregressive models and generalized linear models are available in the literature to handle such problems. These methods are not robust and it involves the assumptions on distribution and method of estimation of parameters. A neural network does not assume any assumptions but needs to handle very large numerical data. Neural networks extract the desired information from the given large data. Neural networks provide a non linear relationship among the variables extracted from the given data. Neural networks perform well in prediction of a study variable for the given auxiliary variables.

In section 2, a brief introduction to neural networks is presented. The problem and its modeling explained in section 3. Final conclusions are given in section 4 followed by references.

2. Neural Networks

The neural networks contain a large number of simple neuron like processing elements and a large number of weighted connections between the elements. The weights on the connections encode the knowledge of a network. Though biologically inspired, many of the neural network models developed do not duplicate the operation of the human brain. The intelligence of a neural network emerges from the collective behavior of neurons, each of which performs only very limited operations even though each individual neuron work slowly, they can still quickly find a solution by working in parallel.

A neural network has a parallel – distributed architecture with a large number of nodes and connections. Each connection points from one node to another are associated with a weight. The frame work is often specified by the number of layers (or slabs) and the number of nodes per layer. The types of layers include:

i. The input layer: The nodes in it are called input units, which encode the instance presented to the network for processing. It does not process information they simply distribute information to other units.
ii. The hidden layer: The nodes in it are called hidden units, which are not directly observable and hence hidden. They provide nonlinearities for the network.
iii. The output layer: The nodes in it are called output units, which encode possible concepts (or values) to be assigned to the instance under consideration.

According to the interconnection scheme, a network can be either feedforward or recurrent and its connections either symmetrical or asymmetrical. In this paper we used the backpropagation feedforward neural network. A Feedforward neural network has all the connections point in one direction (from the input towards the output layer). The name back propagation form the fact the error of hidden units are derived from propagating back ward the error associated with output units since the target values for the hidden units are not given. In the back propagation network, the activation function chosen is the sigmoid function, which compresses the output value into the range between 0 and 1. The sigmoid function is advantageous in that it can accommodate large signals without saturation while allowing the passing of small signals without excessive attenuation. Also, it is a smooth function so that gradient can be calculated, which are required for a gradient decent search the algorithm as follows:

Weight Initialization: Set all weights and node thresholds to small random numbers. Note that the node threshold is the negative of the weight from the bias unit.

Calculation of Activation:

1. The activation level of an input unit is determined by the instance presented to the network.
2. The activation level O_j of a hidden and output unit is determined by $O_j = F\left(\sum W_{ij} O_i - O_j\right)$

   Where W_{ji} is the weight from an input O_i, θ; is the node threshold, and F is a sigmoid function: $F(a) = \left(1 + e^{-a}\right)^{-1}$

Weight Training:

1. Start at the out put units and work back ward to the hidden layer recursively. Adjust weights by $W_{ji}(t+1) = W_{ji}(t) + \Delta W_{ji}$

   Where $W_{ji}(t)$ is the weight unit i to j at time t (or t^{th} iteration) and ΔW_{ji} is the weight adjustment.
2. The weight change is computed by $\Delta W_{ji} = \eta \delta_j O_i + \alpha\left[W_{ji}(t) - W_{ji}(t-1)\right]$

   Where $0 < \alpha < 1$.

3. The error gradient is given by

- for the output unit: $\delta_j = O_j\left(1 - O_j\right)\left(T_j - O_j\right)$

  where T_j is the desired (target) output activation and O_j is actual output activation at output unit j.

- For the hidden unit: $\delta_j = O_j(1 - O_j)\sum \delta_k W_{kj}$

  where δ_k is the error gradient at unit k to which a connection points from hidden unit j.

4. Repeat iterations until convergence in terms of the selected error criterion. An iteration includes presenting an instance, calculating activations and modifying weights.

3. Prediction of pass rate using neural networks

A three factor model is set up to predict the pass rate (percentage) for a particular student in a particular exam. The pass rate (Y) is a function of:

i) The number of assignments (X_1) submitted by the student (a value from 0 to 4).
ii) The student's mark on the mock exam (X_2) (on a scale from 0 to 100).
iii) Whether the student had attended tutorials or not (X_3) (Yes=1/No=0).

Here the data is collected from 500 students in a college on the factors pass rate (Y), number of assignments submitted (X_1) marks on the mock exam (X_2) and the student attendance of tutorials(X_3). We have applied neural networks to predict the pass rate for given three factors.

The data is divided in to three sets 1.Training set, 2. Validation set and 3. Testing set. In this study we have taken 75% of data for training 15% of data for validation and remaining 10% of data for testing set. This testing set is an out of sample set.

A feed forward neural network with one input layer, one hidden layer and an output layer. An input layer consists of three neurons and ten hidden neurons and an output neuron. The following Figure 1 of feedforward neural network gives clear idea about the selected model for the given data.

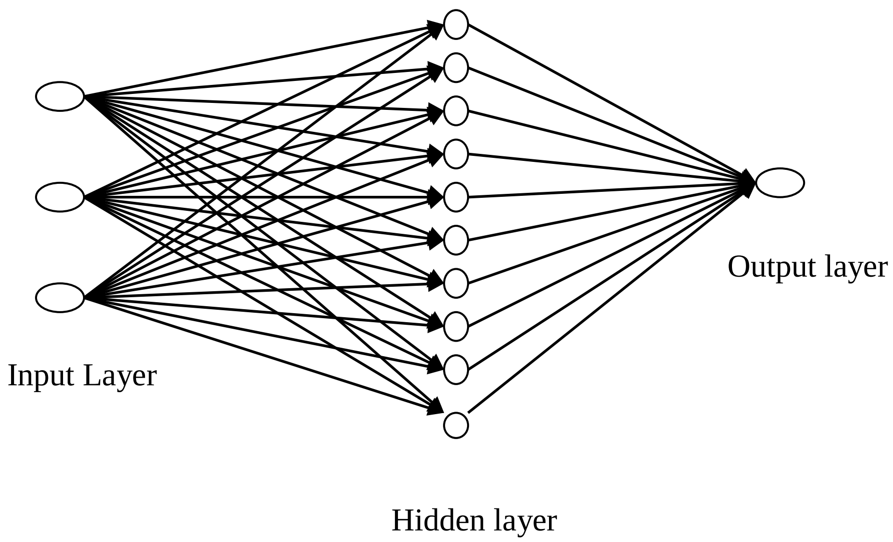

Figure 1. A feedforward neural network with 3 input units, 10 hidden units and an output unit

Backpropagation algorithm for feedforward neural network is used to train the network. The network is trained with learning rate 0.1, momentum rate 0.3 and mean of weights is zero with range 0.5 and number of epochs per training is 1000. Training is continued until the MSE of training set and validation set is less than 0.005 and the model is generalized. Neural network performance is measured using the following measures of errors.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|Y_i - \hat{Y}_i\right|$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|Y_i - \hat{Y}_i\right|}{Y_i}\times 100$$

where Y_i is observed pass rate from testing set and $\hat{Y}_i$ is predicted pass rate of i^{th} student for i=1, 2, … , N(=500). The generalized model is applied on testing set to predict the pass rate and observed the performance of the network by computing the MSE, MAE and MAPE for the predicted and observed values of pass rate of the students.

4. Conclusions

After training and testing the network we have observed the values of MSE=0.0075, MAE=0.0125 and MAPE=0.0042. Small value of MAPE indicates the neural network performance is good and the neural network presents a promising alternative approach in prediction tasks. One can compare these results with any other methods using the above measures of errors.

References

1. Haykin. S. (1999), “Neural Networks; A Comprehensive Foundation”, 2^{nd} ed., Macmillan, New Yark.
2. Laurene Fausett (1994). “Fundamentals of Neural Networks”, Pearson Education.
3. LeMin Fu (2003), “Neural Networks in Computer Intelligence”, Tata McGraw –Hills, Delhi.