International Journal of Advanced Science and Research
ISSN: 2455-4227
Impact Factor: RJIF 5.12
www.allsciencejournal.com
Volume 3; Special Issue 1; March 2018; Page No. 56-58

Implementation of back propagation algorithm using neuroph framework

Rashmi Lad
MIT ACSC, Alandi (D), Pune, Maharashtra, India

Abstract
Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous system, such as the brain, processes information. The key element of this is the novel structure of the information processing system. The purpose of this paper is to introduce the back propagation algorithm using new framework Neuroph. This paper provides clearer understanding of algorithm. Animal dataset is used for finding the minimum square root & iterations.

Keywords: back propagation algorithm, neural network, neuroph

Introduction

Artificial neural network is information processing model which works on the basis of human nervous system like brain. The ANN is a collection of large number of interconnected neurons. It is a network which learns by example. Neural network can be used in multidisciplinary fields such as mathematics, physics, image processing, data mining, time series, philosophy, linguistics, cognitive psychology, neurobiology, AI etc.
Human brain has large no of interconnected neurons. In the biological neurons the electrical signal is transmitted between the synapse and the dendrites. The signal involves a chemical process and transmits the signal to the receiver end. The synapses are to be inhibitory if they passes hinder the firing to the receiving cell or excitatory if they signal firing of the receiving cell.

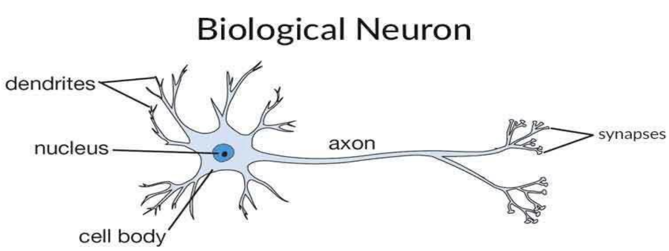

Fig 1

Two types of learning are mainly used in neural network: supervised and unsupervised. Supervised learning is the learning which is performed under any guidance and unsupervised learning is done without any guidance or help. There are many learning algorithms available. Back-propagation is the most important learning algorithm from supervised type. It is also known as multilayer feed forward algorithm. In The training back-propagation network there are three layers: input, output & hidden layers. Here the training is performed in three stages: the feed forward of the input learning pattern, calculations and updation of weights. The input layer always takes only binary & bipolar data.

Neuroph Framework

It is a new lightweight java based framework available for developing the object oriented neural network architecture. It is an open source project designed by SourceForge under the Apache License. Neuroph provide Java library with small number of basic classes for neural network concepts as well as GUI tools. It supports all common neural network architecture such as Back propagation, Perceptron, SOM etc. It also provides support for image recognition.

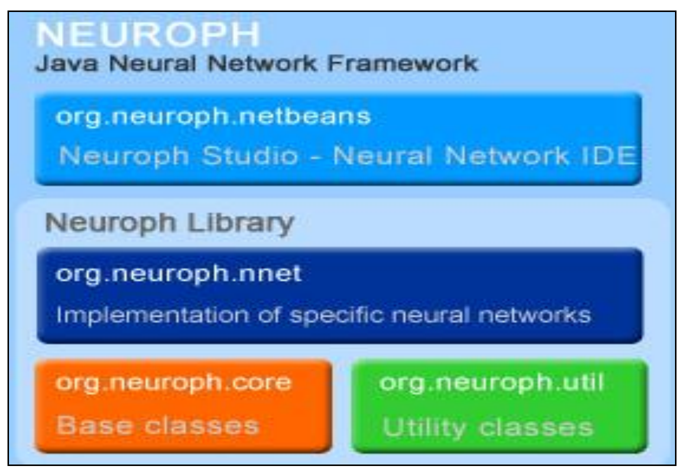

Fig 2

Computational Analysis of Neuroph

In this study we are using Animal dataset [5]. This dataset has 13 rows and 16 columns. All 13 rows show the names of the animals of different category and column shows their features. The dataset is already in normalized and binary form.

Table 1

	Small	Medium	Big	2legs	4legs	Hair	Hooves	Mane	Feathers	Hunt	Run	Fly	Swim
cow	0	0	1	0	1	1	1	0	0	0	0	0	0
Dove	1	0	0	1	0	0	0	0	1	0	0	1	0
Hen	1	0	0	1	0	0	0	0	1	0	0	0	0
Duck	1	0	0	1	0	0	0	0	1	0	0	0	1
Goose	1	0	0	1	0	0	0	0	1	0	0	1	1
Owl	1	0	0	1	0	0	0	0	1	1	0	1	0
Hawk	1	0	0	1	0	0	0	0	1	1	0	1	0
Eagle	0	1	0	1	0	0	0	0	1	1	0	1	0
Fox	0	1	0	0	1	1	0	0	0	1	0	0	0
Dog	0	1	0	0	1	1	0	0	0	0	1	0	0
Wolf	1	1	0	0	1	1	0	1	0	1	1	0	0
Cat	0	0	0	0	1	1	0	0	0	1	0	0	0
Tiger	0	0	1	0	1	1	0	0	0	1	1	0	0
Lion	0	0	1	0	1	1	0	1	0	1	1	0	0
Horse	0	0	1	0	1	1	1	1	0	0	1	0	0
Zebra	0	0	1	0	1	1	1	1	0	0	1	0	0

In this paper for the experiment purpose we have used back propagation supervised learning algorithm that uses sigmoid activation function. It has three layers input, output & hidden. It maps input to the output nodes with more than one hidden layers with appropriate weight. We have used some part of data for training purpose and some part of data for testing purpose. A training dataset is loaded in Neuroph with a number of inputs and output set.

Back propagation algorithm is supervised algorithm for showing different types of combinations of input, output & hidden neuron. Following picture shows the framework for setting different parameters to construct backprapogation neural network.

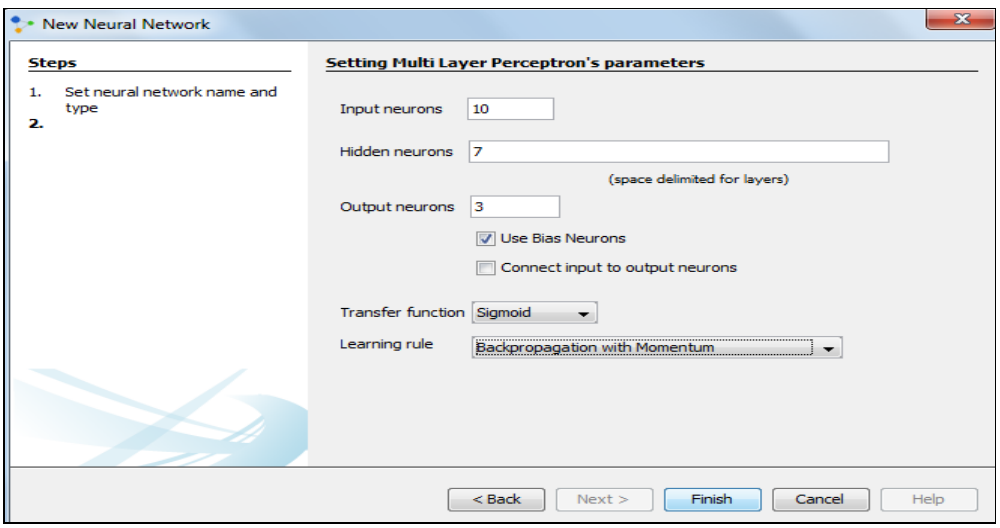

Fig 3

Problem occurs when we increase or decrease the hidden layers. Very less numbers of neurons in the hidden layers generate under fitting problem.

Computational Result Analysis of Neuroph

After constructing architecture for Neural network by Neuroph. It is trained by learning parameters. The maximum error rate stops the network training, if it is achieved. Now here we used same number of input neuron but changed the hidden neuron & learning rate. The momentum is also constant 0.7.

The whole dataset is divided into 2 parts training & testing. Out of 13 we used 10 dataset values for training purpose with changes in hidden layers & 3 dataset we used for testing purpose. The network was first trained and then tested to measure the accuracy.

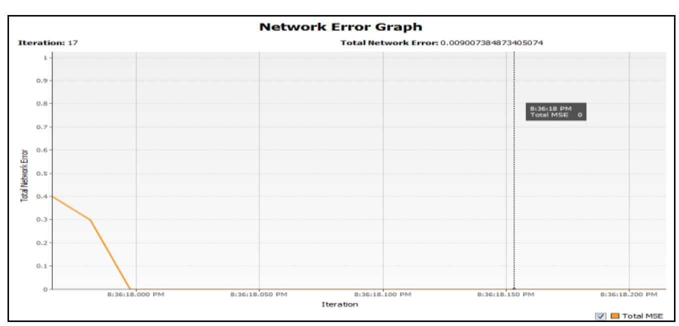

Fig 4

Train Data using Neruoph

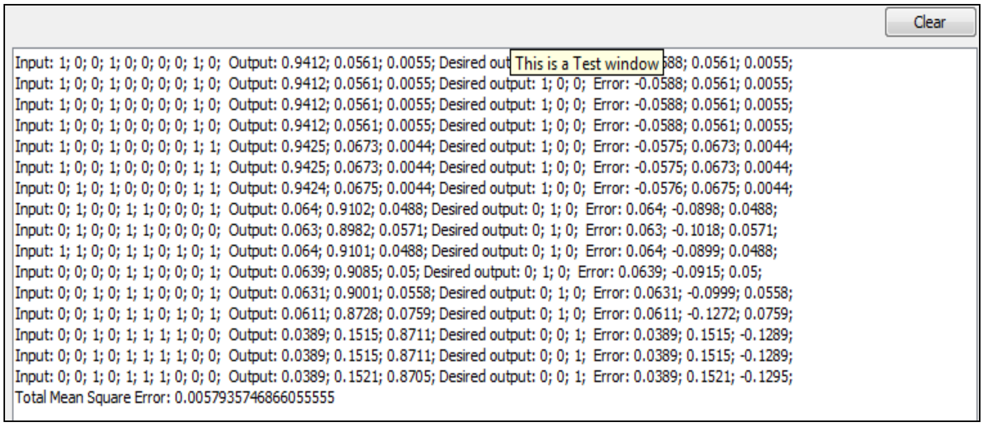

Fig 5

Table 2: Training Data

Sr. No	Input Neuron	Hidden Neuron	Learning Rate	Momentum	Max Error	No. of Iterations	Total Network Error	Total Mean Square Error
1	10	3	0.2	0.7	0.01	50	0.00984	0.00626
2	10	3	0.4	0.7	0.01	15	0.00995	0.00560
3	10	2	0.2	0.7	0.01	66	0.00972	0.00604
4	10	2	0.4	0.7	0.01	29	0.00956	0.00579
5	10	1	0.2	0.7	0.01	1108	0.07865	0.0502
6	10	1	0.4	0.7	0.01	231	0.08388	0.0516
7	10	5	0.2	0.7	0.01	18	0.00949	0.00576
8	10	7	0.2	0.7	0.01	18	0.00926	0.00554
9	10	10	0.2	0.7	0.01	17	0.00900	0.00524
10	10	12	0.2	0.7	0.01	14	0.00902	0.00504

Table 3: Testing Data

Sr. No	Input Neuron	Hidden Neuron	Learning Rate	Momentum	Max Error	No. of Iterations	Total Network Error	Total Mean Square Error
	3	3	0.2	0.7	0.01	626	0.17564	0.1078
	3	3	0.4	0.7	0.01	138	0.18594	0.1088
	3	2	0.2	0.7	0.01	394	0.17644	0.1084
	3	2	0.4	0.7	0.01	151	0.19524	0.1122
	3	5	0.2	0.7	0.01	320	0.17728	0.1080
	3	8	0.2	0.7	0.01	347	0.17704	0.1079

Conclusion

In this paper, we observed that mean square error is decreasing when we increase the size of hidden layers. Here we increased the hidden layers by keeping input & output neurons fixed and found that minimum mean square error gets changed. We found the minimum mean square error for 10 inputs by changing the hidden layers from 5 to 12 and by decreasing the no. of iterations with learning rate 0.2, momentum 0.7 for max error 0.01.

Reference

1. Kahkeshan B, Hassan SI. Oriental Journal of Assessment of Accuracy Enhancement of Back Propagation Algorithm by Training the Model using Deep Learning, 2017.
2. Ehret A, Hochstuhl D, Gianola D, Thaller G. Application of neural networks with back-propagation to genome-enabled prediction of complex traits in Holstein-Friesian and German Fleckvieh cattle, Genet. Sel. Evol. 2015; 47:22.
3. Norouzzadeh MS, *et al.* Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning, 2017, 1-17.
4. Fan F, Cong W, Wang G. General Backpropagation Algorithm for Training Second-order Neural Networks, 2017, 1-5.
5. Pasini A. Artificial neural networks for small dataset analysis, J. Thorac. Dis. 2015; 7(5):953-960.
6. Li J. Information Visualization with Self-Organizing Maps.