Expert Systems with Applications 37 (2010) 7449–7455

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Learning of geometric mean neuron model using resilient propagation algorithm

Md. Shiblee [a], B. Chandra [b,*], P.K. Kalra [a]

[a] Department of Electrical Engineering at IIT Kanpur, Kanpur, India
[b] Department of Industrial Engineering and Management at IIT Kanpur, Kanpur, India

ARTICLE INFO

Keywords:
Neuron model
Geometric mean
Resilient propagation
Functional approximation

ABSTRACT

The paper proposes a new neuron model (geometric mean neuron model) with an aggregation function based on geometric mean of all inputs. Performance of the geometric mean neuron model was evaluated using various learning algorithms like the back-propagation and resilient propagation on various real life data sets. Comparison of the performance of this model was made with the performance of multilayer perceptron. It has been shown that the geometric mean based aggregation function with resilient propagation (RPROP) performs the best both in terms of accuracy and speed.

1. Introduction

Various neuron models have been purposed in the literature (Basu & Ho, 1999; Labib et al., 1999; McCulloch & Pitts, 1943; Plate, 2000; Rumelhart, Hinton, & Williams, 1986, chap. 8; Zhang, Zhao, & Wang, 2000). In 1986, Rumelhart et al. (1986, chap. 8) presented the most popular multilayer perceptron (MLP) model. The aggregation function of the MLP computes the weighted arithmetic mean of the inputs. It is well known that the number of neurons required to solve any problem depends on the mathematical structure of the neuron model. When real life problems are solved using standard artificial neural network, it requires significantly large number of neurons in the architecture. A neuron having higher order statistics can produce superior neural network with comparatively lesser number of neurons. For this, higher order neural networks (HONN) have been suggested in the literature (Chaturvedi, Mohan, Singh, & Kalra, 2004; Giles & Maxwell, 1987; Homma & Gupta, 2002; Sinha, Kumar, & Kalra, 2000; Taylor & Commbes, 1993). Higher order neurons have demonstrated improved computational power and generalization ability. However, these are difficult to train because of a combinatorial explosion of higher order terms as the number of inputs to the neuron increases. Geometric mean based neuron model shown in Fig. 1 is based on a polynomial architecture. Instead of considering all the higher order terms, a simple aggregation function is used. The resulting neuron has fewer parameters than the higher order neurons and is much easier to train. The geometric mean based neuron model is based on weighted geometric mean of all inputs. Nonlinearities in the geometric mean based neuron model is depicted with the parameters being multiplied together. The order of hyperplane in geometric mean based neuron (GMN) model is higher than that of MLP, thus the GMN captures non-linearity more efficiently. Back-propagation with steepest gradient descent and resilient back-propagation algorithm is used for training of neural network.

Rest of the paper is organized as follows. In Section 2, we discuss the mathematical representation of the proposed model. In Section 3 approximation capability of geometric mean based neuron has been proved. The network architecture of feed-forward neural network using geometric mean based model and training of NN with back-propagation (BP) using gradient descent and resilient propagation (RPROP) algorithm are discussed in Section 4. Comparative performance evaluation with geometric mean neuron model using benchmark data sets has been given in Section 5. Conclusions are given in Section 6.

2. Geometric mean neuron model

Neuron model concerns with relating function to the structure of the neuron on the basis of its operation. The MLP model is based on the concept of weighted arithmetic mean of the N input signal (Hornik, Stinchombe, & White, 1989)

$$\text{Weighted arithmetic mean} = \frac{1}{N}\sum_{i=1}^{N} w_i \cdot x_i \tag{1}$$

The proposed neuron model is based on the concept of geometric mean. The weighted geometric mean (Opic & Gurka, 1992) of the N inputs can be found by the summing operation as follows:

$$\begin{aligned}\exp\left(\sum_{i=1}^{N} w_i \cdot \log(x_i)\right) &= \exp\left(\sum_{i=1}^{N} \log(x_i^{w_i})\right)\\ &= \exp\left(\log(x_1^{w_1}\cdots x_N^{w_N})\right) = x_1^{w_1}\cdots x_N^{w_N}\\ &= \text{weighted geometric mean}\end{aligned} \tag{2}$$

* Corresponding author. Tel.: +91 11 26591998; fax: +91 051 22697553.
E-mail address: bchandra104@yahoo.co.in (B. Chandra).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.04.018

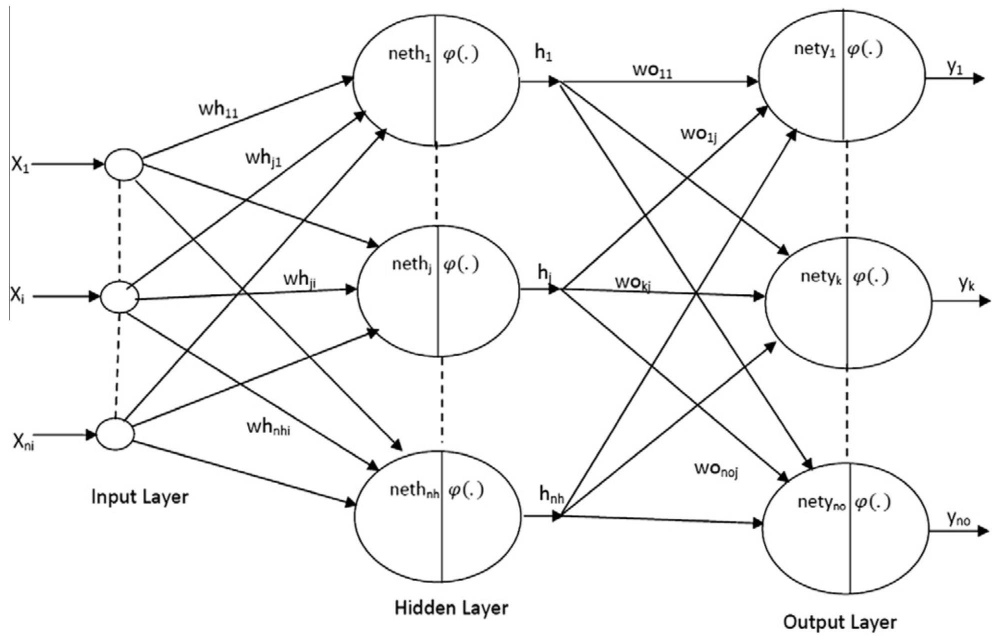

Fig. 1. Architecture of single hidden layer network using GMN model.

and the degree of the above polynomial in n-variables is equal to max $(w_{1i} + w_{2i} + \cdots + w_{ni})$, similarly, the degree of a curve is equal to the degree of the polynomial that defines the curve. Thus the geometric mean operator $f(x,w)$, generates a polynomial of degree n for an n-dimensional input x using the $2n$-dimensional parameters w. For a fixed parameter w, the operator f then represents a function $f : \Re^{2n} \times \Re^n \rightarrow \Re$.

The aggregation function of the new neuron model which gives the weighted geometric mean of the N input signals of the neuron is defined as

$$nety = \exp\left(\sum_{i=1}^{N} w_i \cdot \log(x_i) + b\right) \tag{3}$$

where w_i is adaptive parameter corresponding to the inputx_i. b is the bias of the neuron. *nety* is the net output passing trough the activation function.

On applying an activation function φ, the output y will be given as

$$y = \varphi(nety) \tag{4}$$

It is observed that *tanh* activation function works well for geometric mean neuron model.

3. Approximation capability of geometric mean based neuron model

Classification and prediction problem belong to a broader category of function approximation problem. A common frame work of the approximation is the existence of a relationship between several input variables and one output. This unknown relationship is built up by an approximator whose structure must be chosen such that it represents the best possible relationship between the inputs and output. The most common type of approximator is linear approximator that has the advantage of being simple and cheap in terms of its computational ability. But in real life applications the true relationship between the input and output is non-linear. So it will be advisable to use non-linear approximator for function approximation. The GMN model which is based on weighted geometric mean as the aggregation function can handle non-linear relationship more efficiently Testing the approximation capability of geometric mean based neuron model is done by using the Theorem 1 given in Chen, Chen, and Liu (1995) and Lemma 1 of Huang and Babri (1998). "tanh" is used as activation function in the GMN model which is defined as follows:

$f : \Re \rightarrow \Re$ is called a tan hyperbolic function if the limits, $\lim_{x\rightarrow-\infty} f(x) = -1$, $\lim_{x\rightarrow 0} = 0$, and $\lim_{x\rightarrow\infty} f(x) = 1$, is true.

Following lemma is used in Theorem 1 for proving the approximation capability.

Lemma 1. *If f is bounded tan hyperbolic function and y(x) is a continuous function in* $\Re$ *for which* $\lim_{x\rightarrow-\infty} y(x) = A$, *and* $\lim_{x\rightarrow\infty} y(x) = B$, *where A and B are constant then for any* $\varepsilon > 0$, *there exist* β_i, w_i, b_i, n *such that*

$$\left|\sum_{i=1}^{n} \beta_i f(\exp(w_i \cdot \log(x_i) + b)) - y(x)\right| < \varepsilon \text{ holds for all, } x \in \Re^+$$

Theorem 1. *Given bounded function g(x) in* $\Re$ *and there exists limits* $\lim_{x\rightarrow-\infty} g(x)$ *and* $\lim_{x\rightarrow\infty} g(x)$ *and* $\lim_{x\rightarrow-\infty} g(x) \neq \lim_{x\rightarrow\infty} g(x)$, *then for any arbitrary mapping y(x) in* $C(\Re)$ *for every* $\varepsilon > 0$ *there exist* β_i, w_i, b_i, *n such that*

$$\left|\sum_{i=1}^{n} \beta_i g(\exp(w_i \cdot \log(x_i) + b)) - y(x)\right| < \varepsilon \textit{ holds for all } x \in \Re^+$$

The above theorem can be proved as following

Proof. We can prove the theorem in two steps:

(i) $\lim_{x\rightarrow 0} g(x) = 0$ and $\lim_{x\rightarrow-\infty} g(x) = A$, where A can be any arbitrary nonzero real value.
(ii) $\lim_{x\rightarrow 0} g(x) = B$ and $\lim_{x\rightarrow\infty} g(x) = A$, where A and B are arbitrary unequal real values.

(i) Lets consider $\lim_{x\rightarrow 0} g(x) = 0$ and $\lim_{x\rightarrow\infty} g(x) = A$, where A are arbitrary nonzero real value. Let $g_1(x) = g(x)/A$. Then $\lim_{x\rightarrow 0} g_1(x) = 0$ and $\lim_{x\rightarrow\infty} g_1(x) = 1$. According to Lemma 1, for every $\varepsilon > 0$, there exist $n \in N$ and β'_i, w_i, b_i for $i = 1,2,\ldots,n$