Kuwait J. Sci. **41** (2) pp. 151-174, 2014

Impact of learning rate and momentum factor in the performance of back-propagation neural network to identify internal dynamics of chaotic motion

S. KARMAKAR*, G. SHRIVASTAVA** AND M. K. KOWAR*

**Bhilai Institute of Technology (BIT), Bhilai House, Durg, 491001, Chhattisgarh (INDIA).*
***Dr. C.V. Raman University, Bhilaspur, Chhattisgarh (INDIA).*
e-mail: dr.karmakars@gmail.com

ABSTRACT

The utilization of back-propagation neural network in identification of internal dynamics of chaotic motion is found appropriate. However, during its training through Rumelhart algorithm, it is found that, a high learning rate (α) leads to rapid learning but the weights may oscillate, while a lower value of 'α' leads to slower learning process in weight updating formula $\Delta v_{jk} = \alpha \delta_j x_i$ Momentum factor (μ) is to accelerate the convergence of error during the training in the equation $w_{jk}(t+1) = w_{jk}(t) + \alpha \delta_k z_j + \mu\{w_{jk}(t) - w_{jk}(t-1)\}$ and $v_{jk}(t+1) = v_{jk}(t) + \alpha \delta_k z_{+} \mu\{v_{jk}(t) - v_{jk}(t-1)\}$ while transfer function sigmoid $f(x) = \frac{1}{1+e^{-\delta x+n}}$. It is the most complicated and experimental task to identify optimum value of 'α' and 'μ' during the training. To identify optimum value of 'α' and 'μ', firstly the network is trained with 10^3 epochs under different values of 'α' in the close interval $0 < \alpha < 1$ and $\mu = 1$. At $\alpha = 0.3$ the convergence of initial weights and minimization of error (i.e., mean square error) process is found appropriate. Afterwards to find optimum value of μ, the network was trained again with $\alpha = 0.3$ (fixed) and with different values of μ in the close interval $0 < \mu < 1$ for 10^3 epochs. It was observed that the convergence of initial weights and minimization of error was appropriate with $\alpha =$ 0.3 and $\mu = 0.9$. On this optimum value of α and μ the network was trained successfully from local minima of error = 1.67029292416874E-03 at 10^3 epochs to global minima of error = 4.99180426869658E-04 at 15×10^5 epochs. At the global minima, the network has exhibited excellent performance in identification of internal dynamics of chaotic motion and in prediction of future values by past recorded data series. These essentials are presented through this research paper.

Keywords: Back-propagation; learning rate; momentum factor; neural network.

INTRODUCTION

Chaos theory and identification of internal dynamics for prediction of future values is a subject matter of study in mathematics, with applications in several disciplines including, physics, engineering, medical science, meteorology and hydrology (climate forecasting). Chaos theory studies the behavior of dynamical systems those are highly sensitive to initial conditions, an effect which is popularly referred to as the butterfly effect. Small differences in initial conditions yield widely diverging outcomes for such dynamical systems, rendering long-term prediction impossible in general. This happens even though these systems are deterministic, meaning that their future behavior is fully determined by their initial conditions, with no random elements involved. In other words, the deterministic nature of such systems does not make them predictable. This behavior is known as deterministic chaos, or simply chaos. Lorenz (1996, 1972) has described Chaos, as that when the present determines the future, but approximate present does not approximately determines the future. Chaotic behavior is being observed in many natural systems, such as weather. In common usage, "chaos" means, "a state of disorder". However, in chaos theory, the term is defined more precisely.

Although there is no universally accepted mathematical definition of chaos, a commonly used definition says that for a dynamical system to be classified as chaotic, it must have the following properties:

(1) Sensitive to initial conditions.

(2) Random motion.

(3) High internal dynamics.

(4) Difficult to forecast.

Sensitivity to initial conditions is popularly known as the "butterfly effect" (Lorenz, 1972). For example, the flapping wings of a butterfly represent a small change in the initial condition of the system, which causes a chain of events leading to large-scale weather phenomena. A consequence of sensitivity to initial conditions is that, if we start with only a finite amount of information about the system, then beyond a certain time the system will no longer be predictable. This is most familiar in the case of weather, which is generally predictable only about a week ahead. The most widespread techniques used for prediction are the numerical and statistical methods. But it is quite difficult to forecast such chaotic behavior. Researches in the field of predicting chaotic data time series are being conducted for a long time, but successes are rarely visible. For example, Basu & Andharia (1992) found that the rainfall data time series shows a chaotic behavior with its predictors not only to be chaotic in nature but also suffer from epochal changes. They presented an alternative approach based on the theory of chaos, which treated the time series of monsoon rainfall as deterministic but possibly chaotic. They used past '*n*' years rainfall data as predictors making the forecast possible for "$n+1$" months in advance. Some significant contributions also found for the same by statistical approach. However, Guhathakurta (1998, 2000, 2006); Guhathakurta *et al.* (1999); Rajeevan (2001); Rajeevan *et al.* (2004); Thapliyal & Kulshrestha (1992); Thapliyal (1997); Thapliyal & Rajeevan (2003); Krishnamurthy & Kinter (2003); Krishnamurthy & Kirtman (2003) and Sahai *et al.* (2002) have found that statistical models have inherent limitations such as the models are not useful to study the highly nonlinear relationships between dependent (i.e., target) and independent (i.e., predictors) parameters, even if one considers models like power regression. It is concluded that the identification of internal dynamics of rainfall for long period (chaos) is approximately difficult.

From Rumelhart *et al.* (1986), the Artificial Neural Networks (ANNs) have been proved to be a powerful soft computing technique for prediction of highly complex and nonlinear systems like chaos. ANNs belong to the black box time series models and offer a relatively flexible and quick means of modeling. These models can treat the non-linearity of system to some extent due to their parallel architecture. Kowar *et al.* (2013) have found that the successful applications of ANN models may be in the simulation of chaotic series with high degree of accuracy. A broad literature review from 1986 to 2012 has been carried out.

It has been found that the two main architecture of ANN are back-propagation neural network (BPN) and radial-basis function (RBF) network commonly used by the researchers, especially for this problem. However, BPN is used frequently in various applications worldwide. For illustration, the long-range monsoon rainfall forecasting over a smaller geographical region is a very challenging task for the scientists around the globe. According to Basu & Andharia (1992); Mohammed (2010); Patil & Ghatol (2010) it is mainly because of the chaotic behavior of rainfall data time series and due to the same reason, researches in these fields are being conducted for a long time, but successes of these models are rarely visible. Many researchers have introduced number of models for chaotic series forecasting. No multiple models have forecasted the same situation in exactly same way with same results. At the same time, no single model is reliable for chaos forecasting. Climate and rainfall are highly non-linear phenomena in nature.

Through literature review, it is also found that architectures of ANN such as BPN and RBF are best established to forecast chaotic behavior and are efficient enough to forecast rainfall as well as other weather parameter (chaos) prediction phenomenon over the smaller geographical region (Chih-Hong *et al.*, 2011). In support of the same experiment of BPN system in deterministic forecast, we have gone through the literatures. Guhathakurta (2006) has successfully applied BPN in long-range forecast of monsoon rainfall over very smaller Indian region "Kerala". In this forecast, past recorded rainfall data time series is used to forecast the future value. In many other cases BPN is found to be fit for prediction of other climate activities. Enireddy *et al.* (2010) used the BPN model for predicting the rainfall data time series. 99.8% and 94.3% accuracy were obtained by them during the training and testing period respectively. From these results they concluded that rainfall can be predicted in future using the same method. Sawaitul *et al.* (2012) and Kowar *et al.* (2013) also performed experiments on forecasting future weather to arrive at the conclusion that BPN algorithm can also be applied on the weather forecasting data. Thus it is concluded that the ANNs are capable of modeling in identification of internal dynamics of chaotic motion. The ANN signal processing approach for chaos is capable of yielding good results and can be considered as an alternative to traditional approaches. Present survey of literatures in the proposed field of research exposed that deterministic forecasting method is one of the useful techniques for predicting chaotic motion especially when the identification of physically connected predictors is difficult. A finite-dimensional dynamical system is a system, whose state at any instant can be completely characterized by a set of scalar observations x_1, x_2,..., x_n. This set is of course fixed and must always characterize the system throughout its evolution. The evolutionary history of the system is then given by time series $x_1(t)$, $x_2(t)$..., $x_n(t)$; these functions of time trace out a trajectory in n-dimensional phase space. Guhathakurta (1998, 2000, 2006) discussed a dynamical system is deterministic if its evolution is completely determined by its current state and past history. It is found that BPN is sufficiently suitable for identification of internal dynamics of chaotic series by past history of series. However, selection of its parameters likes:

(1) Number of input vectors (n).

(2) Number of hidden layers (m).

(3) Number of neurons in hidden layers (p).

(4) Number of output neurons (y).

(5) Weights and biases.

(6) Learning rate (α),

(7) Momentum factors (μ).

seems crucial during design time. Especially for chaos prediction, no authors have provided optimum value of these parameters. Within these parameters it is found that the impact of 'α' and 'μ' for the performance of BPN system is extremely crucial. It is found that, weight changes in BPN system involve a combination of current gradient and the previous gradient. This approach is beneficial when some training data are very different from a majority of the data. Sivanandam *et al.* (2006) and Kumar (2007) pointed out that a small 'α' is used to avoid major trouble of the direction of learning, when very unusual pair of training patterns is presented in chaos. High learning rate 'α' leads to rapid learning but the weights may oscillate, while a lower learning rate leads to slower learning in weight updating formula $\Delta v_{jk} = \alpha\delta_j x_i$. On the other hand, if 'μ' is added to the weight update formula, then the convergence becomes faster. The weights from one or more previous training patterns must be saved in order to use momentum. For the BPN with 'μ' the new weights for training step $t+2$ is based on t and $t+1$. It is found that 'μ' allows the net to perform large weight adjustments as long as the correction proceeds in the same general direction for several patterns. Thus using 'μ' the network does not proceed in the direction of gradient, but travels in the direction of the combination of the current gradient and previous direction for which the weight correction is made. Sivanandam *et al.* (2006) have explained that the main purpose of the 'μ' is to accelerate the convergence of error propagation algorithm. This method makes the current weight adjustment with a fraction of the recent weight adjustment. The weight updating formulas (Equation 1 and 2) for BPN with momentum are:

$$w_{jk}(t+1) = w_{jk}(t) + \alpha\delta_k z_j + \mu\{w_{jk}(t) - w_{jk}(t-1)\} \tag{1}$$

$$v_{jk}(t+1) = v_{jk}(t) + \alpha\delta_k z_j + \mu\{v_{jk}(t) - v_{jk}(t-1)\} \tag{2}$$

where, $0 < \alpha < 1, 0 < \mu < 1$

Thus identification of an appropriate value of 'α' and 'μ', for the most favorable performance of BPN is a challenge for the scientists. And it is most constructive during the modeling of chaos forecasting. In this study, the BPN is used as deterministic forecast. 4 separate experiments have been prepared with different values of 'α' and 'μ' to recognize the impact of 'α' and 'μ' during its training and testing period.

DATA DESCRIPTION AND PREPROCESSING

Sixty two years (1951 - 2012) total monsoon rainfall data time series of Ambikapur region (total geographical area is 15733 km^2) in India, which represented chaotic motion is considered for the study. Since BPN system with its transfer function 'sigmoid' $f(x) = \dfrac{1}{1 + e^{-\delta x + n}}$ is limited to the close intervals 0 and 1 therefore data time series is normalized by using following Equation 3 and used as input to BPN system. Equation 4 is used to de-normalize authentic representation of output (results) in this paper. Data for first 57 years (1951 - 2007) are used for training the BPN and tested for the years 2008 to 2012.

$$r_i = \frac{(x_i + \min(x_i)}{(x_i + \max(x_i)} \tag{3}$$

$$x_i = \frac{\{\min(x_i) - r_i.\max(x_i)\}}{(r_i - 1)} \tag{4}$$

ABOUT BPN MODEL

BPN in deterministic forecast is illustrated in Figure 1, wherein 11 input vectors (x_1, $x_{2,}$..., x_{11}) in input layer are used to input past eleven years' data time series, 3 neurons in hidden layer (z_1...z_3) and one neuron (y_k) in output unit are used to observe 12$^{\mathrm{th}}$ year prediction value. Karmakar *et al.* (2009, 2012) and Kowar *et al.* (2013) have found that the mean absolute deviation (MAD) is inversely proportional to number of input vector 'n' and $11 \leq n < 15$ is found appropriate. Therefore $n = 11$ has been chosen. $11 \times 3 = 33$ hidden layer weights, 03 output layer weights, 03 hidden layer biases, and 01 output layer bias is used in the system to be trained. And these weights $v_{ij}s$, $w_{ij}s$, v_0, and w_0 (total 40) are trained during the training period. Phillip (2003) observed that one hidden layer is sufficient for all types of chaos, while use of two hidden layers rarely improves the model and it may introduce a greater risk of converging to a local minima. One of the key causes is that it increases unknown variables (weights and biases) in the network to be trained. Karmakar *et al.* (2012) and Kowar *et al.* (2013) identified that the 3 neurons in hidden layer and 11 input vectors provided satisfactory performance of BPN in deterministic forecast. And further increment of neurons in hidden layer is increases MAD between actual and predicted values. The neurons output is obtained as $f\ (x_j)$ known as transfer function is typically the sigmoid axon given in the following Equation 5. The output $f\ (x_j)$ is depicted in Figure 2.

$$f(x) = \frac{1}{(1 + e^{-\delta x + n})} \tag{5}$$

Where δ determines the slope and η is the threshold. In the proposed model $\delta = 1, \eta = 0$ have been considered with the output of the neuron in close interval [0, 1] as shown in Figure 2. The BPN in deterministic forecast is trained with 57 years (1951-2007) training dataset. In every epoch (i.e., parallel iteration process) during the training process (Rumelhart *et al.*, 1986) algorithm is used to minimize the error i.e., mean square error (MSE). The training started with initial set of weights and biases between 0 and 1. During the experiments, it was carefully observed that how MSE got optimized regularly after each epoch.

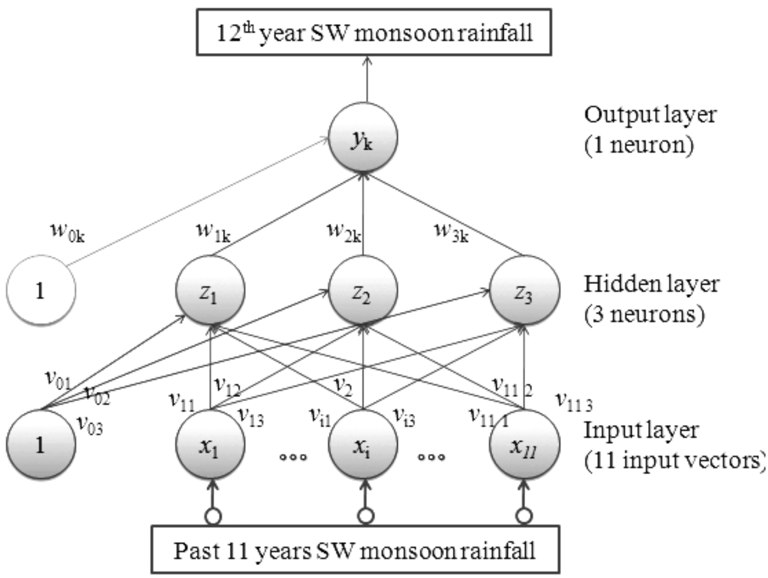

Fig. 1. The BPN model

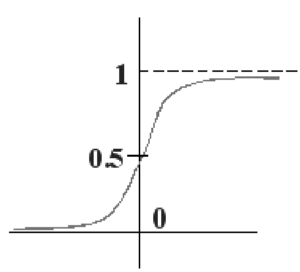

Fig. 2. Output of Sigmoid Axon

The model acceptance criteria is measured by two statistical identifiers namely: standard deviation (SD) and mean absolute deviation (MAD) given in Equation 6 and 7 respectively with a hypothesis (H). Where 'H' is defined as *MAD must incredibly less than or at least half of the SD*. If H is true, then model can be accepted otherwise not. The performance criterion is measured by correlation coefficient (CC) between actual and model predicted values. To accept and check performance, the model is analyzed during the training period (1951-2007), and testing period (2008-2012).

$$MAD = \left| \frac{1}{n} \sum_{i=1}^{n} (x_i - p_i) \right| \quad (6)$$

$$SD = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - m)^2} \quad (7)$$

Where, p_i represents predicted values and m is mean.

IDENTIFICATION OF LEARNING RATE (α) AND MOMENTUM FACTOR (μ)

To observe the impacts of changes in the value of 'α' and 'μ' in the BPN model to identify the internal dynamics of chaotic motion, Four experiments were performed with different values 'α' and 'μ' as follows:

1. Experiment 1 ($0 < \alpha < 1$, $\mu = 1$ and 10^3 epochs).
2. Experiment 2 ($\alpha = 0.3$, $0 < \mu < 1$ and 10^3 epochs).
3. Experiment 3 ($\alpha = 0.3$, $\mu = 0.2$, and $15x10^5$ epochs).
4. Experiment 4 ($\alpha = 0.3$, $\mu = 0.9$, and $15x10^5$ epochs).

Experiment 1 ($0 < \alpha < 1$, $= 1$, and 10^3 epochs)

Trainable weights of the model are initialized by the random values between 0 and 1.Emphasize is given on the impact of 'α' by considering different values of 'α', ranging from 0.1 to 0.9 in the model during the training period. For each value of 'α' the model is trained with 10^3 epochs repeatedly for 10 times. Finally, their average MSE is analyzed as depicted in Table 1. From the data obtained from such experiment, convergence of the network has been analyzed. It is found to be lowest for $\alpha = 0.1$ but it is already proved that the lower 'α' leads to slower learning process, thus 0.1 cannot be considered as an appropriate value of 'α', because the theory of Rumelhart *et al.* (1986) does not support this value practically and which also may cause slower learning as well as adverse effect to the results discussed by Sivanandam *et al.* (2006). Figure 3 demonstrated the graphical representation of the same result given in Table 1. Although in the experiment convergence was found slower (i.e., MSE $= 0.0013768651795$) at $\alpha = 0.3$ as compare to at $\alpha = 0.1$ and 0.2.

Table 1. Average MSE at various value of α between 0.1 to 0.9

α	MSE
0.1	0.00137449301736871
0.2	0.00137601436701650
0.3	0.00137686517950020
0.4	0.00137689263460138
0.5	0.00137606629962170
0.6	0.00137606629962170
0.7	0.00137606629962170
0.8	0.00137606629962170
0.9	0.00137606629962170

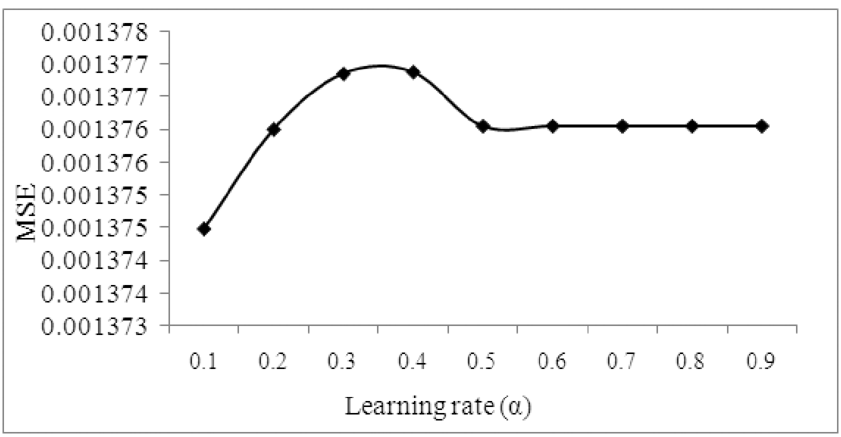

Fig. 3. Minimizing MSE at $0 < \alpha < 1$ and through 10^3 epochs

However, it has been observed that convergence at point 0.3, 0.6, 0.7, 0.8, 0.9 are almost same, but according to the Reumelhart *et al.* (1986) high 'α' leads to rapid learning, but the weights may oscillate. On the basis of all these facts, α = 0.3 is considered as optimum for further experiments.

Experiment 2 (α = 0.3, 0 < μ < 1 and 10^3 epochs)

To identify the impact of 'μ' on BPN model, the model is trained with optimum value of 'α' i.e., 0.3 and different values of between 0.1 to 0.9 was considered. For each value of 'μ' the BPN is trained 10 times for 10^3 epochs. Finally their average MSEs are found as given in Table 2 and Figure 4.

Table 2. Average MSE
($\alpha = 0.3$ and $\mu = 0.1$ to 0.9 with 10^3 epochs)

α	μ	**MSE**
0.3	0.1	0.001658868009638590
0.3	0.2	0.001659529384976310
0.3	0.3	0.001660240177338890
0.3	0.4	0.001659545857317930
0.3	0.5	0.001659545857317930
0.3	0.6	0.001659545857317930
0.3	0.7	0.001659681426853800
0.3	0.8	0.001659711835229300
0.3	0.9	0.001659606166807380

Results of above experiment illustrated that the minimum MSE is found at $\mu = 0.1$. And increased at $\mu = 0.2$ to 0.3. At $\mu = 0.3$ it is decreased. And after that however, the MSE remains almost constant for $\mu = 0.4$ to 0.6 with slight variation between $\mu = 0.7$ to 0.9. As in the theory it is clearly mentioned that, in case of higher value of 'μ' the weights may oscillate. Therefore, value of $\mu = 0.2$ is considered as an optimum. Because the convergence at $\mu = 0.2$ to 0.6 except for $\mu = 0.3$ MSE almost remains the same. The adverse effects on results due to weight oscillation can be avoided by fixing the value of μ at 0.2, this being in the lower side of the range of value of μ. From the above two experiments, it has been finalized that the values of α and will be 0.3 and 0.2 respectively for further experiments.

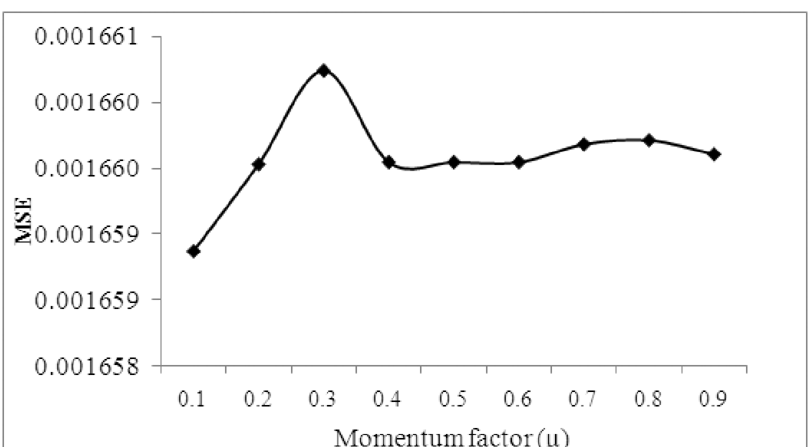

Fig. 4. Minimizing MSE at $\alpha = 0.3$, $0 < \mu < 1$ through 10^3 epochs

Experiment 3 ($\alpha = 0.3$, $\mu = 0.2$, and 15x10^5 epochs)

With the value of $\alpha = 0.3$, $\mu = 0.2$ the BPN model is trained for 15x10^5 epochs intended for obtaining global minima (maximum trained point). At this point, the model becomes capable to identify the internal dynamics of chaos. The training started with initial set of random weights between 0 and 1 as shown in Table 3. The MSE minimized regularly after each epoch as depicted in Table 4 and Figure 5. After 15x10^5 epochs, the MSE reached a minimum level of 8.47798380689012E-04. The optimized weights at this point are shown in Table 5.

Table 3. Initial random weights

v_i = 1 to 11; j = 1 to 3		
0.7794150114059440	0.3072796463966360	0.5058456063270560
0.8212675452232360	0.5119876265525810	0.6868295073509210
0.3309511542320250	0.8144663572311400	0.5876017212867730
0.1313177943229670	0.1872385144233700	0.6259894967079160
0.6847181916236870	0.3795685768127440	0.3410908579826350
0.4834440946578970	0.3880071043968200	0.0004483461380005
0.2634690403938290	0.3107233643531790	0.8642087578773490
0.8884155154228210	0.2106779813766470	0.2879743576049800
0.8755260109901420	0.1681295037269590	0.4218183755874630
0.7731931805610650	0.1036903262138360	0.8074117898941040
0.8535689115524290	0.7007688283920280	0.8036214113235470
Initial weights v_{0i}; i = 1 to 3		
0.7650007009506220	0.3771285414695740	0.4249131083488460
Initial weights w_i; i = 1 to 3		
0.9578890800476070	0.7296340465545650	0.2057505846023550

Table 4. Minimization of MSE during training process

Epoch Count	MSE
1	1.63038145643859E-03
10^2	1.63022276413987E-03
10^3	1.62896874155549E-03
10^4	1.35184423358056E-03
10^5	1.34879656789549E-03
5x10^5	1.33598879654699E-03
11x10^5	1.32886756688897E-03
12x10^5	1.12765899896565E-03
13x10^5	9.98956773987995E-04
15x10^5	8.47798380689012E-04

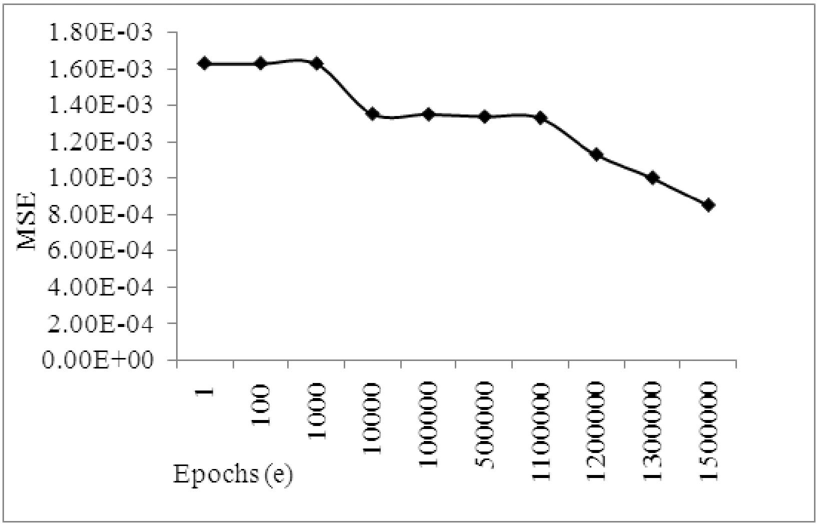

Fig. 5. Minimization of MSE during training process

Table 5. Optimized weights after 15x10^5 epochs

	$v_{=1 \text{ to } 11;\ j=1 \text{ to } 3}$	
9.441118622142240	2.522718281525280	-3.800881407232070
6.808217910988250	1.883782784715540	-3.929845922089530
-6.441564391167690	-0.894517875416847	2.421932330501940
-4.386089144258910	-0.897749338227441	1.553870729111280
1.628959993914080	-0.388992286997265	-1.874375683363850
-6.107070634909740	-1.206877589966790	3.342372200262020
-0.784777109922923	-0.782243359175445	0.538706691691305
-0.402774653460837	-1.514272539028290	-1.615819434660100
-2.469422223603950	-0.460996744249289	1.169718064312870
-2.200750523205190	-0.188971677296908	1.286681721617640
0.672876739092872	0.912859764380702	0.685643438154767
	Updated weights $v_{0i};\ i = 1 \text{ to } 3$	
0.001980641526768	-0.004014601830225	0.002435093590160
	Updated weights w_0	
	0.00354914779937090	

Experiment 4 ($\alpha = 0.3$, $\mu = 0.9$, and $15x10^5$ epochs)

To evaluate and review the impact of variations in α and against the results of last experiment, the same experimental setup with same data set but with different values of α and μ has been repeated. Here, may cause weight changes to be in a direction that would increase the error. Thus the value of $\mu = 0.9$ is considered as appropriate value for training the model which will accelerate the convergence but avoid the increase in error. The training started with initial set of weights between 0 and 1 as shown in Table 6, i.e., after $15x10^5$ epochs the MSE is minimized up to 4.99180426869658E-04 marked as M_G (Global minima) and the optimized weights are shown in Table 7. The training started with initial set of weights between 0 and 1 at point '*P*' where MSE = 1.63289262934093E-01. After $15x10^5$ epochs the MSE reached its lowest point 4.99180426869658E-04 marked as M_G, the global minima or maximum trained network point as shown in Table 8 and Figure 6. In the previous literatures various authors have clearly mentioned that attaining such point is almost difficult or temporal nervousness. Interestingly, such point has been achieved in the present study. In this experiment MSE is more minimized than that obtained during experiment 3.

Table 6. Initial random weights

	$v_{i=1 \text{ to } 11;\ j=1 \text{ to } 3}$	
0.280800521373748	0.168759763240814	0.044127523899078
0.472349166870117	0.809812307357788	0.855300962924957
0.119313240051269	0.312592983245849	0.731210827827453
0.923533260822296	0.312689185142517	0.295242071151733
0.313810527324676	0.941224575042724	0.792520821094512
0.007087528705597	0.538136720657348	0.904589712619781
0.512941122055053	0.947724163532257	0.393840074539184
0.071080148220062	0.571404635906219	0.451620757579803
0.040320515632629	0.674218833446502	0.487735211849212
0.353351771831512	0.232466399669647	0.005873143672943
0.984928369522094	0.470367133617401	0.641462087631225
	Initial weights $v_{0ii} = 1 \text{ to } 3$	
0.5814671516418457	0.5955716967582703	0.21110987663269043
	Initial weights $w_{ii} = 1 \text{ to } 3$	
0.6662726998329163	0.20196348428726196	0.8917340636253357

Table 7. Optimized weights

	$v_{i=1 \text{ to } 11;\ j=1 \text{ to } 3}$	
13.212680441	-3.043977842	-14.397325147
12.839138110	6.880667764	-3.267866395
-18.220808470	-11.208725106	-7.818172624
-1.665566401	3.380170693	9.262771717
6.007453047	2.824425500	-0.613637485
-13.113640093	-5.893302749	2.576067728
-5.373775965	-6.117592000	-3.119470042
-4.065856516	-5.748926243	-5.717660860
1.752119801	2.226165188	4.205130417
-5.860865432	-3.593515579	-2.458649507
9.387518936	14.261855302	20.131087700
	Updated weight $v_{0i;\ i=1 \text{ to } 3}$	
0.0001052014214497	0.0001015295532543	0.0000240460520913
	Updated w_0	
	0.000167955628075041	

Table 8. Optimized MSE

Epoch Count	**MSE**
1	1.63289262934093E-01
10^2	1.67082747834919000E-03
10^3	1.67029292416874000E-03
10^4	1.65516581866368000E-03
10^5	1.33629916609829000E-03
$5x10^5$	9.15076092085467000E-04
$11x10^5$	5.75971925301642000E-04
$12x10^5$	5.63477906270142000E-04
$13x10^5$	5.43576724081598000E-04
$15x10^5$	4.99180426869658E-04

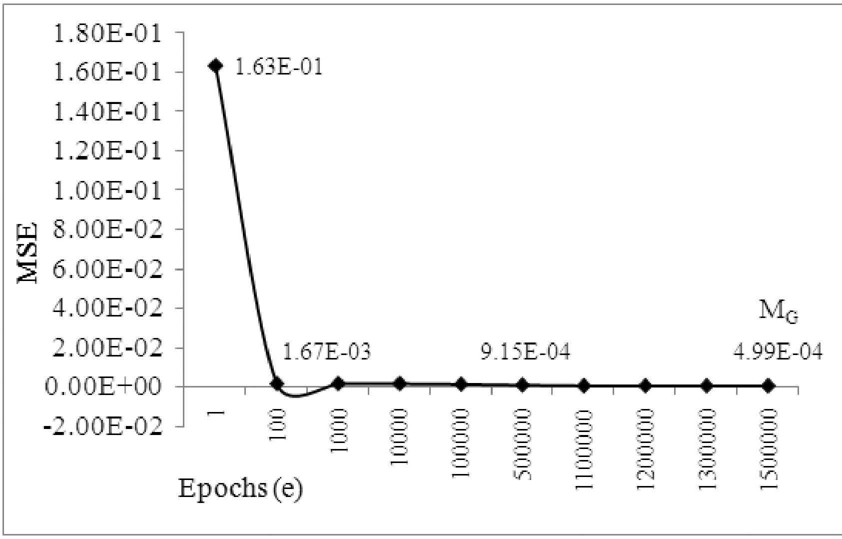

Fig. 6. Minimizing MSE at $\alpha = 0.3$, $\mu = 0.9$, and 15x10^5 epochs

RESULTS AND DISCUSSIONS

From experiments 3 and 4, it was observed that the minimization of MSE is almost equal up to 10^3 epochs. However, in experiment 3 constant trend of MSE was noted even after 10^3 epochs and convergence speed was found to be slow. MSE got further minimized to 8.47798380689012E-04 after 15x 10^5 epochs. In experiment 4, the MSE was found to decrease gradually after 10^3 epochs and convergence speed was noted to high. Finally the MSE got minimized to 4.99180426869658E-04 after 15x 10^5 epochs. It was observed that experimental set up 4 is more efficient in minimizing the MSE, compared to experimental set up 3. In both the experiments the BPN was tested independently through independent data set from 2008 to 2012. The statistical data sets of the training and testing period for both the experiments are provided in Table 9. During the testing period in experiment 3, the MAD (% of mean) is just half of the SD. value of CC = 0.7 indicated that the model was not trained properly. Therefore, poor performance is shown in the testing period. During testing period, it was observed that the MAD (% of mean) is more than the SD indicating that the proposed model cannot be accepted.

On the other hand in experiment 4, during testing period the MAD (% of mean) was very less as compare to SD with very high (0.88) value of CC. These indicated that the model got trained properly and clearly defined the internal dynamics by creating a relationship between independent variables (x_1, x_2,..., x_{11}) and dependent variable (y_k).

Table 9. Performance of the BPN during Training and Testing Period in experiment 3 and 4

	Training Period			Testing Period		
	SD	MAD	CC	SD	MAD	CC
Experiment #3	7.3	4.18	0.7	10.1	11.4	0.9
Experiment # 4	7.3	2.841	0.88	10.1	7.1	0.7

Experiment 4 is most appropriate since during the testing period of this experiment, MAD (% of mean) was much less than SD as evident from above Table 9. From the present study, it is thus concluded from the results obtained that in identification of the internal dynamics of a chaotic series (rainfall data time series for the present study), the following parameters are most appropriate in designing a BPN Model:

(1) Number of input vector (n) : 11

(2) Number of layer (m_1) : 03

(3) Number of hidden layer (m_2) : 01

(4) Number of hidden neurons (p) : 03

(5) Number of output neuron (y) : 01

(6) Optimum value of 'α' : 0.3

(7) Optimum value of 'μ' : 0.9

(8) Transfer function $f(x)$: Sigmoid axon

(9) Number of epochs (e) : $15\text{x } 10^5$

Performance of the BPN model during training and testing period by the experiment 3 and 4 is shown in Table 10 and 11 respectively. Although experiment 3 wherein $\alpha = 0.3$ and $\mu = 0.2$ cannot be accepted because it is not trained properly and also 'H' is false since observed MAD (% of mean) is found to be on the higher side. Hence it is concluded that the proposed BPN model is improper during training as well as testing period. On the other hand, the proposed BPN used in experiment 4 with $\alpha = 0.3$ and $\mu = 0.9$ is found to be appropriate since the value of MAD (% of mean) is observed low whereas, the value of CC is high. This reiterates that the acceptance of the hypothesis 'H' is true.

Verification for the current year 2012 is also given in the Table 12 for both the experiments. It is found that, experiment 3 is completely impractical. It produced very high deviation between actual and predicted value i.e., 1146.4 mm. However, experiment 4 with $\alpha = 0.3$ and $\mu = 0.9$ predicted accurately with a nominal deviation (32.9 mm.). Thus, it is clear that the impact of 'α' and 'μ' is vital during the design of BPN model especially for the prediction of chaotic motion. Any slight change on the value of 'α' and '' may collapse the BPN model as observed during the experiments performed. The graphical representation of performance of the BPN during training, testing and verification in both the experiments is presented in Figure 7 and 8 respectively. It is seen from Figure 7 that the deviation between actual and predicted value is very high as predicted. However, Figure 8 has properly explained the internal dynamics during the training and testing period.

Table 10. Performance during training period (1962-2007)

		Experiment 3 $\alpha = 0.3$ **and** $\mu = 0.2$		**Experiment 4** $\alpha = 0.3$ **and** $\mu = 0.9$	
Year	**Actual data (in mm.)**	**Predicted data (in mm.)**	**Absolute deviation (in mm.)**	**Predicted data (in mm.)**	**Absolute deviation (in mm.)**
1962	952.3	1204.0	251.7	928.3	24
1963	1089.6	1136.6	47	1105.7	16.1
1964	1523.4	1129.6	393.8	1253.0	270.4
1965	1226.0	1257.1	31.1	1449.5	223.5
1966	815.4	1061.8	246.4	863.7	48.3
1967	1081.4	1070.8	10.6	1127.2	45.8
1968	914.8	1098.0	183.2	1039.6	124.8
1969	1193.3	972.7	220.6	1206.2	12.9
1970	906.0	821.1	84.9	827.1	78.9
1971	1773.5	1254.9	518.6	1751.2	22.3
1972	1188.6	1146.5	42.1	1267.9	79.3
1973	1153.4	942.0	211.4	1213.8	60.4
1974	919.4	1782.1	862.7	988.4	69
1975	1534.6	1386.6	148	1592.5	57.9
1976	1604.3	1151.8	452.5	1416.5	187.8
1977	1674.0	989.6	684.4	1590.4	83.6
1978	1103.4	1461.8	358.4	1353.7	250.3
1979	1020.1	1371.5	351.4	1111.1	91
1980	936.9	1590.3	653.4	951.5	14.6
1981	1106.9	1171.2	64.3	1221.2	114.3
1982	1303.2	1233.5	69.7	1329.4	26.2
1983	1254.8	1313.6	58.8	975.6	279.2
1984	1184.4	1085.3	99.1	1085.0	99.4

Cont. Table 10. Performance during training period (1962-2007)

		Experiment 3 $\alpha = 0.3$ and $\mu = 0.2$		Experiment 4 $\alpha = 0.3$ and $\mu = 0.9$	
Year	Actual data (in mm.)	Predicted data (in mm.)	Absolute deviation (in mm.)	Predicted data (in mm.)	Absolute deviation (in mm.)
1985	1231.7	1443.3	211.6	1018.4	213.3
1986	1257.8	1077.1	180.7	1187.5	70.3
1987	1540.0	1128.3	411.7	1621.6	81.6
1991	1645.6	894.7	750.9	1334.2	311.4
1992	1190.7	950.6	240.1	1401.2	210.5
1993	1236.6	1624.9	388.3	1219.9	16.7
1994	2092.8	1331.2	761.6	1498.5	594.3
1995	1146.7	1226.4	79.7	1095.5	51.2
1996	1619.7	1435.0	184.7	1514.9	104.8
1997	1139.4	1391.9	252.5	1250.8	111.4
1998	1049.3	1168.5	119.2	1048.7	0.6
1999	1229.5	1223.9	5.6	1582.1	352.6
2000	1236.0	1230.9	5.1	1218.0	18
2001	1820.5	1137.9	682.6	1787.9	32.6
2002	1086.0	1319.7	233.7	1078.1	7.9
2003	1240.6	1091.0	149.6	1106.6	134
2004	858.4	1584.7	726.3	803.8	54.6
2005	952.7	1003.2	50.5	872.8	79.9
2006	1066.3	1041.0	25.3	1010.6	55.7
2007	1046.8	968.2	78.6	945.8	101

Table 11. Performance during testing period (2008 - 2011)

		Experiment 3 $\alpha = 0.3$ and $\mu = 0.2$		Experiment 4 $\alpha = 0.3$ and $\mu = 0.9$	
Year	Actual data (in mm.)	Predicted data (in mm.)	Absolute deviation (in mm.)	Predicted data (in mm.)	Absolute deviation (in mm.)
2008	1358.4	1142.7	215.7	1139.2	219.2
*2009	603.2	1211.0	607.8	1271.1	667.9
*2010	649.7	1177.3	527.6	1345.6	695.9
2011	1445.5	1087.4	358.1	1412.0	33.5

Table 12. Verification of BPN for 2012

Year	Actual data (in mm.)	Experiment 3 $\alpha = 0.3$ and $\mu = 0.2$		Experiment 4 $\alpha = 0.3$ and $\mu = 0.9$	
		Predicted data (in mm.)	Absolute deviation (in mm.)	Predicted data (in m.)	Absolute deviation (in mm.)
2012	1181.8	2328.2	1146.4	1148.9	32.9

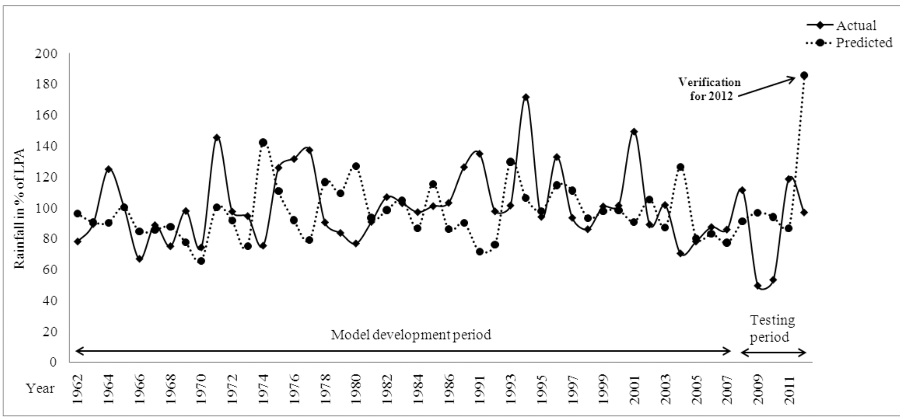

Fig. 7. Performance of the BPN in experiment 3 ($\alpha = 0.3$ and $\mu = 0.2$)

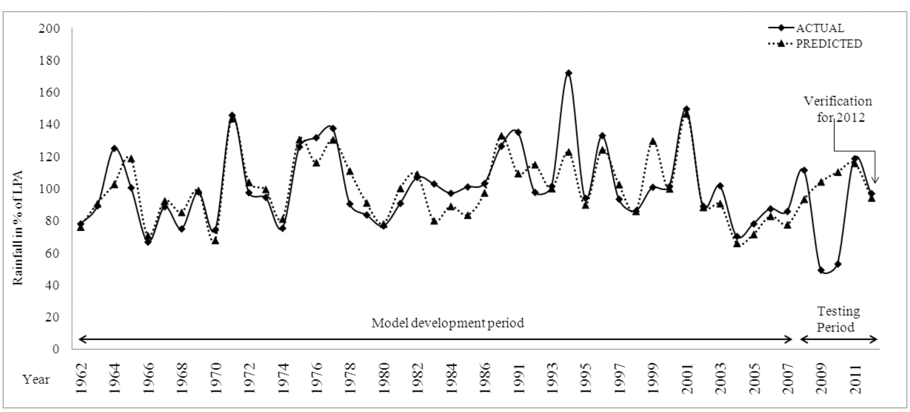

Fig. 8. Performance of the BPN model in experiment 4 ($\alpha = 0.3$ and $\mu = 0.9$)

CONCLUSIONS

Previous researchers in the field of study of the present work during 1986 - 2012 concluded that the identification of internal dynamics of chaotic motion and its prediction for future is extremely difficult. Though BPN has sufficient skills to overcome such shortcomings, but a proper selection of appropriate parameters is of utmost importance and a challenging task. These parameters can be optimized by the theory except 'α' and 'μ'. These two parameters have unusual effects on the performance of BPN model. The impact of variability of these two parameters has identified and observed following three vital facts:

(1) $\alpha = 0.3$ is found optimum. It is neither high nor low ($0.2 < \alpha < 0.9$). The theory states that high α leads to rapid learning, but the weights may oscillate. The lower rate leads to slower learning process.

(2) μ is to accelerate the convergence of error algorithm during the training period. As $\alpha = 0.3$ and $\mu = 0.2$, the BPN has shown high level of convergence of error in limited number of epochs 'e'. However, performance of the BPN is found exceptionally unfortunate with such values of α and μ.

(3). With $\alpha = 0.3$ and $\mu = 0.9$, the BPN is trained properly and also found efficient enough.

Particularly for this problem $\alpha = 0.3$ and $\mu = 0.9$ is found optimum and these values have produced exceptional performance (SD = 7.3; MAD = 2.841; CC = 0.88) with a high level of convergence of error (MSE = 4.99180426869658E-04) during the training process. However, it is noted that their values may diverge for other problems. Thus, identification of impact of 'α' and 'μ' is extremely vital and therefore their optimum values must be chosen carefully through experiments only. Finally, it can be concluded that the BPN model can be applied to forecast chaotic motion through deterministic process. However, required superiority to select its parameters like v_{ij}s,w_{ij}s, v_0, w_0, m_1, m_2, n, p, $f(x)$, e, y_k, α, and μ is vital. In the present study, the optimum value of these parameters, especially for this problem is:

$$m_1 = 3, m_2 = 1, n = 11, p = 3, f(x) = \frac{1}{1 + e^{-+n}}, e = 15 \times 10^5, y_k = 1, \alpha = 0.3 \text{ and } \mu = 0.9.$$

However, it may change with type of data series and chaos present in the series. Thus a careful experimentation to optimize the values of the parameters is highly suggested.

REFERENCES

Basu, S. & Andharia, H. I. 1992. The Chaotic time series of Indian Monsoon rainfall and its prediction, Proceedings of the Indian Academy of Science **101**: 27-34.

Chih-Hong, K., Chun-Fei, H. & Hon-Son, D. 2011. Design of an adaptive self-organizing fuzzy neural network controller for uncertain nonlinear chaotic systems. Neural Computing and Applications **21**: 1243-1253.

Enireddy, V., Varma, K.V.S.R.P., Sankara Rao, P. & Satapati, R. 2010. Prediction of rainfall using backpropagation neural network model. International Journal on Computer Science and Engineering **2**: 1119-1121.

Guhathakurta, P. 1998. A hybrid neural network model for long range prediction of all india summer monsoon rainfall. Proceedings of WMO international workshop on dynamical extended range forecasting, Toulouse, France.

Guhathakurta, P., Rajeevan, M. & Thapliyal, V. 1999. Long Range Forecasting Indian Summer Monsoon Rainfall by Hybrid Principal Component Neural Network Model. Meteorology and Atmospheric Physics **71**: 255-266.

Guhathakurta, P. 2000. New models for long range forecasts of summer monsoon rainfall over north west and peninsular India. Meteorology and Atmospheric Physics **73**: 211-255.

Guhathakurta, P. 2006. Long-Range Monsoon Rainfall Prediction of 2005 for the Districts and Sub-Division Kerala with Artificial Neural Network. Current Science **90:** 773-779.

Karmakar, S., Kowar, M. K. & Guhathakurta, P. 2009. Artificial neural network skeleton in deterministic forecast to recognize pattern of TMRF Ambikapur District, CSVTU Research Journal **2**: 41-45.

Karmakar, S., Kowar, M. K. & Guhathakurta, P. 2012. Application of neural network in long range weather forecasting: In the context of smaller geographical region (i.e.Chhattisgarh State, India). Lambert Academic Publishing, Germany. pp. 57-87.

Kowar, M. K., Karmakar, S. & Guhathakurta, P. 2013. Development of backpropagation neural network for prediction of chaotic data time series. Logos Verlag Berlin, Germany.

Krishnamurthy, V. & Kinter, J. L. 2003. The Indian monsoon and its relation to global climate variability. Global Climate - Current Research and Uncertainties in the Climate System (eds Rodo, X. and Comin, F. A.). pp. 186-236.

Krishnamurthy, V. & Kirtman, B. P. 2003. Variability of the Indian Ocean: Relation to Monsoon and ENSO. Quarterly Journal of the Royal Meteorological Society **129:** 1623-1646.

Kumar, S. 2007. Neural network computer engineering series. The McGraw-Hill, New Delhi, pp. 104-152.

Lorenz, E. 1996. The essence of chaos. University of Washington Press.

Lorenz, E. 1972. Predictability: Does the flap of a butterfly's wings in Brazil set off a Tornado in Texas. American Association for the Advancement of Science, Washington D.C.

Mohammed, A. 2010. Chaotic time series prediction using wavelet neural network. Journal of Artificial Intelligence **1**:73-80.

Patil, C. Y. & Ghatol, A. A. 2010. Rainfall forecasting using local parameters over a meteorological station: An Artificial Neural Network Approach. Journal of Engineering Research and Industrial Applications **3**: 341-356.

Phillip, H. S. 2003. DTREG Predictive modeling software, available online on http://www.dtreg.com. pp. 175-181.

Rajeevan, M. 2001. Prediction of Indian summer monsoon: Status, problems and prospects. Current Science **81:** 1451-1457.

Rajeevan, M., Pai, D. S., Dikshit, S. K. & Kelkar, R. R. 2004. IMD's new operational models for long-range forecast of southwest monsoon rainfall over India and Their verification for 2003. Current Science **86**: 422-431.

Rumelhart, D., Hinton, G. E. & Williams, R. J. 1986. Learning internal representation by error propagation. r j parallel distributed processing: Exploration in the Microstructure of Cognition, MIT Press, Cambridge. pp. 318-362

Sahai, A. K., Grimm, A. M., Satyan. V. & Pant, G. B. 2002. Prospects of prediction of indian summer monsoon rainfall using global sst anomalies. IITM Research Report No. RR-093.

Sawaitul, S.D., Wagh, K. P. & Chatur, P. N. 2012. Classification and prediction of future weather by using back propagation algorithm-an approach, International Journal of Emerging Technology and Advanced Engineering **2**: 111-115.

Sivanandam, S. N., Sumathi, S. & Deepa, S. N. 2006. Introduction to neural networks using MATLAB 6.0. The McGraw Hill Companies, Delhi. pp. 189-192

Thapliyal, V. & Kulshrestha, S. M. 1992. Recent models for long range forecasting of southwest monsoon rainfall over India. Mausam **43:** 239-248.

Thapliyal, V. 1997. Preliminary and final long range forecasts for seasonal monsoon rainfall over India. Journal of Arid Environment **36**: 385-403.

Thapliyal, V. & Rajeevan, M. 2003. Updated operational models for long-range forecasts of Indian summer monsoon rainfall. Mausam **54**: 495-504.

Submitted : *02/05/2013*
Revised : *07/01/2014*
Accepted : *12/01/2014*

تأثير نسبة التعلم وعامل الزخم على أداء الشبكات العصبية ذات الانتشار الخلفي في التعرف على الديناميكا الداخلية للحركة الفوضوية

[*]س. كارماكار، [**]ج. شريفاستافا، [***]م. كووار

معهد بيهيلاي للتكنولوجيا - بيت بيهيلاي الهند
جامعة الدكتور س.ف. رامان – بيهيلاسبور – الهند

خلاصة

يعد استغلال الشبكات العصبية ذات الانتشار الخلفي في تحديد الديناميكا الداخلية للحركة الفوضوية مناسب. ولكن اثناء تدريب الشبكة باستخدام لوغارثمروملهارت، وجد ان ارتفاع نسبة التعلم (α) يزيد من سرعة التعلم مع تفاوت الاوزان، بينما انخفاض α يقلل من سرعة التعلم وفق معادلة تحديث الوزن $\Delta V_{jk} = \alpha \delta_j x_i$ الغرض الرئيسي من عامل الزخم (μ) هوتسريع التقارب بين الخطأ أثناء عملية التدريب وفقا للمعادلة $W_{jk}(t+1) = W_{jk}(t) + \alpha \delta_k z_j + \mu\{W_{jk}(t) - W_{jk}(t-1)\}$ والمعادلة $V_{jk}(t+1) = V_{jk}(t) + \alpha \delta_k z_j + \mu\{V_{jk}(t) - V_{jk}(t-1)\}$ عندما يكون تطبيق النقل sigmoid هو $f(x) = \dfrac{1}{1+e^{-\delta x + n}}$ وتبقى عملية تحديد القيم المناسبة لـ "α" و "μ" أثناء عملية التدريب من أصعب المهام التجريبية. ولتحديد أفضل القيم لـ"α" و "μ" تم أولا تدريب الشبكة على 10^3 من العهود تحت قيم مختلفة لـ "α" في الفترة المفتوحة $0 < \alpha < 1$ و$\mu = 1$ عند $\alpha = 0.3$ وجد أن قيم تقارب الأوزان الأولية والتقليل من الخطأ (مربع متوسط الخطأ) تكون مثالية. بعد ذلك لإيجاد القيمة المناسبة لـ "μ"، تم تدريب الشبكة مع إبقاء $\alpha = 0.3$ (ثابت) وتغيير قيم "μ" في الفترة المفتوحة $0 < \mu < 1$ لـ 10^3 من العهود. لوحظ أنالقيمة المثالية هي $\mu = 0.9$. عند هذه القيم المناسبة لـ "α" و "μ" تم تدريب الشبكة بنجاح والتدرج بالقيمة الدنيا المحلية لمربع متوسط الخطأ $1.67029292416874E-03$ عند 10^3 من العهود إلى القيمة الدنيا الشاملة وهي $4.99180426869658E-04$ عند 15×10^5 من العهود.، وقد قدمت الشبكة أداءاً مميزاً عند القيمة الدنيا الشاملة في تحديد الديناميكا الداخلية للحركة الفوضوية وفي

التنبؤ بالقيم المستقبلية من خلال تسجيل سلاسل البيانات الماضية. وقد تم عرض جميعهذه الأساسيات في هذه الورقة البحثية.

الكلمات المفتاحية: الشبكات العصبية ذات الانتشار الخلفي، نسبة التعلم، عامل الزخم.