July 2012

Architecture of Artificial Neural Network in Identification of Internal Dynamics and Prediction of Dynamic System Rainfall Data Time Series

Sanjeev Karmakar

Bhilai Institute of Technology, Durg, Bhilai House, 491001, Durg, Chhattisgarh, India,
karmakar_sanjeev@gmail.com

Manoj Kumar Kowar

Bhilai Institute of Technology, Durg, Bhilai House, 491001, Durg, Chhattisgarh, India, mkkowar@gmail.com

Pulak Guhathakurta

India Meteorological Department, Pune, Sivaji Nagar, Pune, Mharastra, India,
pguhathakurta@rediffmail.com

Follow this and additional works at: <https://www.interscience.in/ijcsi>

Part of the [Computer Engineering Commons](#), [Information Security Commons](#), and the [Systems and Communications Commons](#)

Recommended Citation

Karmakar, Sanjeev; Kowar, Manoj Kumar; and Guhathakurta, Pulak (2012) "Architecture of Artificial Neural Network in Identification of Internal Dynamics and Prediction of Dynamic System Rainfall Data Time Series," *International Journal of Computer Science and Informatics*: Vol. 2 : Iss. 1 , Article 4.

DOI: 10.47893/IJCSI.2012.1060

Available at: <https://www.interscience.in/ijcsi/vol2/iss1/4>

This Article is brought to you for free and open access by the Interscience Journals at Interscience Research Network. It has been accepted for inclusion in International Journal of Computer Science and Informatics by an authorized editor of Interscience Research Network. For more information, please contact sritampatnaik@gmail.com.

Architecture of Artificial Neural Network in Identification of Internal Dynamics and Prediction of Dynamic System Rainfall Data Time Series

Bennykutty Varghese¹, Gyanesh Shrivastava², Sanjeev Karmakar³,
Manoj Kumar Kowar⁴ & Pulak Guhathakurta⁵

^{1,2,3&4}Bhilai Institute of Technology, Durg, Bhilai House, 491001, Durg, Chhattisgarh, India

⁵India Meteorological Department, Pune, Sivaji Nagar, Pune, Maharashtra, India

E-mail : karmakar_sanjeev@gmail.com³, mkkowar@gmail.com⁴, pguhathakurta@rediffmail.com⁵

Abstract - The objective of this study is to expand and evaluate the back-propagation artificial neural network (BPANN) and to apply in the identification of internal dynamics of very high dynamic system such long-range total rainfall data time series. This objective is considered via comprehensive review of literature (1978-2011). It is found that, detail of discussion concerning the architecture of ANN for the same is rarely visible in the literature; however various applications of ANN are available. The detail architecture of BPANN with its parameters, i.e., learning rate, number of hidden layers, number of neurons in hidden layers, number of input vectors in input layer, initial and optimized weights etc., designed learning algorithm, observations of local and global minima, and results have been discussed. It is observed that obtaining global minima is almost complicated and always a temporal nervousness. However, achievement of global minima for the period of the training has been discussed. It is found that, the application of the BPANN on identification for internal dynamics and prediction for the long-range total annual rainfall has produced good results. The results are explained through the strong association between rainfall predictors i.e., climate parameter (independent parameter) and total annual rainfall (dependent parameter) are presented in this paper as well.

Keywords - Meteorology, Rainfall, Spatial, Interpolation, Neural Network, Back-Propagation.

I. INTRODUCTION

In much science and engineering practice today, there is an increasing demand for techniques which are capable of interpolating irregularly scattered data distributed in space. These techniques have many applications including rainfall estimation. Mathematically, the general model for spatial interpolation of values μ in a surface 'R' can be expressed as:

$$\varphi = f(x_1, x_2, \dots, v_1, v_2, \dots, v_n)$$

Where, $(x_1, x_2, \dots)$ is a inputted independent parameters and $v_1, v_2, \dots, v_n$ are additional variables such as weights to be optimized. In this section, contributions from the year 1978 to 2011 have been reviewed. It is found that, there are several interpolation methods for solving the above problem have been used. Techniques such as geostatistical co-kriging (Journel and Huijbregts, 1978) and ANN (Hornik et al., 1987) are common [1, 2]. Goovaerts, 1997, Demyanov, 1998, S. van der Heijden has suggested four methods for interpolation including ANN in their work [4]. Huang et al. (1998), demonstrated the use of dynamic fuzzy-reasoning-based function estimator (DFFE) to interpolate rainfall data in a case study in Switzerland [5]. The functional

parameters are also optimized by genetic algorithms (GA). Ricardo and Palutikof, 1999, have successfully applied ANN technique for simulation of daily temperature for climate change scenarios over Portugal [6]. Guhathakurta, 1998, has applied hybrid ANN in prediction of Indian monsoon rainfall and has found very significant results almost more than 85% correctness [7]. Guhathakurta, 1999, applied this ANN method effectively in prediction of surface Ozone which is very dynamic and unpredictable in nature for very smaller geographical earth region. He has also contributed extremely significant finding in the year of 2000 and 2006 with the similar ANN methods especially for climate variable prediction [7-13]. Rajeevan et al. (2000) have given a power regression based long-rang monsoon rainfall forecasting system for peninsular Indian region [14]. However, Guhathakurta (2006) and Karmakar et al. (2008) have found that, the ANN method is better evaluated over the statistical technique for the same [15-21]. Snell, 2000 has found a new method for the spatial interpolation of daily maximum surface air temperatures. This new method uses ANN to generate temperature estimates at eleven locations given information from a lattice of surrounding locations. The out-of-sample performance

of the ANNs is evaluated relative to a variety of benchmark methods (spatial average, nearest neighbor, and inverse distance methods). He has found that, the ANN approach is superior both in terms of predictive accuracy and model encompassing [22]. Antonić et al. (2001) have described Spatio-temporal interpolation of climatic variables over large region of complex terrain using neural networks [23]. Presented interpolation models provide reliable, both spatial and temporal estimations of climatic variables, especially useful for dendroecological analysis. Rigol et al. (2001) have described spatial interpolation of daily minimum air temperature using a feed-forward back-propagation neural network. Simple network configurations were trained to predict minimum temperature using as inputs: (1) date and terrain variables; (2) temperature observations at a number of neighbouring locations; (3) date, terrain variables and neighbouring temperature observations. This is the first time that trend and spatial association are explicitly considered together when interpolating using an ANN [24]. Koike et al. (2001) suggested an interpolation method based on a multilayer neural network (MNN), has been examined and tested for the data of irregular sample locations. He found that, the main advantage of MNN is in that it can deal with geographical data with nonlinear behavior [25]. Bryan and Adams (2002) have successfully applied in interpolation of Annual Mean Precipitation and Temperature Surfaces for China [26, 27]. Li (2002) have suggested ANN models could be used to accurately estimate these weather variables. In this study, ANN-based methods were developed to estimate daily maximum and minimum air temperature and total solar radiation for locations in Georgia. Observed weather data from 1996 to 1998 were used for model development, and data from 1999 to 2000 were used for final ANN model evaluation [28]. Iseri et al. (2002) have found that, the ANN in Medium Term Forecasting of Rainfall over Indian region is sufficiently suitable [29]. Silva 2003 has successfully applied ANN technique for special interpolation of temperature variable carried out in the Austrian Central Institute of Meteorology and Geodynamics – ZAMG, on the scope of the COST Action 719. This work focuses on the capabilities of Neural Networks used in the spatial interpolation of some climate variables showing the advantages and disadvantages [30]. The application of an ANN on the Austria mean air temperature distribution for August has produced good results, explained by the strong altitude dependency of this parameter. Timonin and Savelieva, 2005, have described another application for spatial prediction of radioactivity using GRNN [31]. Guhathakurta 2006, Karmakar et al. (2008, 2009) have found that, the Rumelhart, 1986 learning algorithm based Neural Network model can be also appropriate for long-range monsoon rainfall

prediction over very smaller geographical region. Attorre et al. (2007) have compared three methods that have been proved to be useful at regional scale: 1 - a local interpolation method based on de-trended inverse distance weighting (D-IDW), 2 - universal kriging (i.e. simple kriging with trend function defined on the basis of a set of covariates) which is optimal (i.e. BLUP, best linear unbiased predictor) if spatial association is present, 3 - multilayer neural networks trained with back-propagation (representing a complex nonlinear fitting) and found ANN interpolator has proven to be more efficient [32]. Chattopadhyay, and Chattopadhyay, 2008, have found that, the ANN for Indian Rainfall prediction using ANN is suitable and especially he suggested architecture of hidden layer of the network. Ultimately it has been established that the eleven-hidden-nodes three-layered neural network has more efficacy than asymptotic regression in the present forecasting task [33]. Hung, 2009, presented ANN technique to improve rainfall forecast performance over Bangkok Thailand [34]. Mendes and Marengo, 2010 have developed and tested a novel type of statistical downscaling technique based on the ANN, applied of the climate change. The ANN used here are based on a feed forward configuration of the multilayer perception that has been used by a growing number of authors [35]. Sivapragasam et al. (2010) have found that, the interpolation of Hydrological variable such as rainfall, ground water level, etc for Tamil Nadu state India using ANN is useful. However, 18 stations data samples have been used to develop ANN. It is observed that, by more stations, this methodology can produce better result [36]. Ghazanfari et al. (2011), have suggested a new model PERSIANN (Precipitation estimation from remotely sensed information using artificial neural network) model works based on the ANN system which uses multivariate nonlinear input-output relationship functions to fit local cloud top temperature to pixel rain rates. In this study, PERSIANN model and two interpolation methods (Kriging & IDW) are employed to estimate precipitation cover for North-Khorasan between the years 2006 until 2008. He has found better correlation between PERSIANN output and station data than the other two interpolation methods. While correlation coefficient for Kendal's test is 0.805 between model and Bojnord Station data, this coefficient is 0.488 and 0.565 for Kriging method [37].

By review of very significant contributions of all the authors from 1978 to 2011, it is concluded that, ANN method is sufficiently suitable for identifying internal dynamics of rainfall variables, as well as significant to develop an association between dependent and independent variable in prediction as well as interpolation. However, it is also found that, the selection of appropriate architecture of ANN is

exceptionally tricky in addition to its training. Selection of its parameters, i.e., number of input vectors, hidden layers, and hidden neurons is again very dynamic depending on data series. Training of ANN is also again enormous challenging task. It is found that, the selection of parameters and training process may cause of temporal nervousness as well. To train the network, how many epochs are required, how to obtain global minima during error minimization process (i.e., training) is not often clearly available in the literature although very wide range of applications of ANN has been found. No author has found who have clearly described these aspects in their contributions. The development of BPANN with these essential aspects, application in identification of internal dynamics of high dynamic system such as long range total annual rainfall by developing a relationship between rainfall and its predictors and its performances are clearly described in the subsequent sections.

II. DATA DESCRIPTION AND PREPROCESSING

A data time series (1992-2009) of total annual rainfall (TAR) of location Baster of Chhattisgarh (Geological situation Latitude 17046' N to 2405'N Longitude 80015' E to 84020' E) is considered in this study. The eight climate parameters data time series (1992-2009) those are physically connected with the TAR of this location are identified and used to input in the ANN as independently. These parameters are total evaporation, maximum temperature, minimum temperature, sunshine, vapor pressure, wind, humidity and total rainfall of previous year as shown in Table 1. These parameters are shown good Correlation Coefficient (CC) with TAR. The all data time series dependent as well as independent was then split into two sets. About thirteen years (1993-2005) of data are used for training. Other four years (2006-2009) are used for observation (testing). According to Demyanov, 1998, Bryan and Adams, 2001, 2002, important aspect is to normalize the data because network training algorithms are limited to the intervals 0 to 1 and de-normalize it after the testing phase [4, 26]. To normalize the data time series the equation $R_i = (x_i + \min(x_i)) / (x_i + \max(x_i))$ is used and other hand the equation the equation $x_i = (\min(x_i) - R_i \max(x_i)) / (R_i - 1)$ is used to de-normalize.

Table 1. Input Parameters

No.	Parameters (Prev. Year)		CC
1	Total Evaporation	x_1	0.39
2	Total Maximum Temperature	x_2	0.46
3	Total Min Temperature	x_3	0.36
4	Total Sunshine	x_4	0.39
5	Total Vapor Pressure	x_5	0.42

6	Total Wind	x_6	0.33
7	Total Humidity	x_7	0.44
8	Total Rainfall	x_8	0.34

III. ARCHITECTURE OF ANN

Multilayer Back Propagation Artificial Neural Network (BPANN) is shown in Figure 1. The modified architecture of this network as shown in Figure 2 is used in this study. Where, it has input layer (at the bottom), one hidden layer (at the middle) and output layer (on the top). It has n input vectors at the input layer ($x_1 \dots x_i \dots x_n$), p neurons in hidden layer ($z_1 \dots z_j \dots z_p$) and one neuron (y_k) in output unit to observe desired meteorological data as target variable. Bias on hidden unit j i.e. V_{oj} and bias on output unit k i.e. w_{ok} with $n \times p + p$ trainable weights are used in the network. Output target value for the network is actual data to be predicted. The neurons output can be obtained as $f(x_i)$. Where f is a transfer function (axon), typically the sigmoid (logistic or tangent hyperbolic) function.

Sigmoid function $f(x) = \frac{1}{1 + e^{-\delta x + \eta}}$, where δ

determines the slope and η is the threshold. In the proposed model $\delta = 1$, $\eta = 0$ is considered such that $\forall \pm n \in I^+$, the output of the neuron will be in close interval $[0, 1]$ as shown in Fig. 3.

Fig.1 : Architecture of general multilayer BPANN

Fig. 2 : Architecture of BPANN

The network performance is determined through comparison between mean absolute deviation (MAD) (% of mean) and standard deviation (SD) (% of mean) as shown in Equation 1 and 2 respectively. Where x_i is random variable with mean μ , and p_i is predicted value.

Fig. 3. Output of Sigmoid Axon

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

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

IV. DESIGN OF TRAINING ALGORITHM

Rumelhart et al. (1986) first introduced BPANN based on gradient descent method (Kumar, 2007) [38]. Being a gradient descent method, it minimizes the mean square error (MSE) of the output computed by the network during the feed-forward and back-propagation process. In this study, BPANN is trained by supervised learning method. Supervised training is the process of providing the network with a series of sample inputs and comparing the output with the expected response. The following algorithm 'a' has been designed specifically for this application to train BPANN. The aim of this training was to train the BPANN in order to achieve a relationship between the inputted independent parameters $x_1, x_2, \dots, x_8$ (see Table 1) and the dependent parameter i.e., TAR.

Algorithm a: Designed and implemented training Algorithm of BPANN through Java for this application where-

Input training vector x : $(x_1 \dots x_i \dots x_n)$

Output target vector t : $(t_1 \dots t_k \dots t_m)$

δ_k : Error at output unit y_k .

α : Learning rate.

V_{oj} : Bias on hidden unit j.

Z_j : Hidden unit j.

w_{ok} : Bias on output unit k.

y_k : Output unit k.

Set initial weights w_k such that $(0 < w_k < 1)$.

Consider training period: m;

Set number of input vector ρ such that $\rho < m$.

do{

for($i = 1; i \leq (m - \rho); i++$)

{

for($j = 1; j \leq (\rho + j); j++$)

{

input vector x_j ;

}

/* ...Feed-Forward and calculate target/output variable $y...$ */

- Each input unit receives the input signal x_i and transmits this signals to all units in the layer above i.e., hidden units.

- Each hidden unit ($z_j, j = 1 \dots p$) sums its

$$\text{weighted input signals } z_{-inj} = v_{oj} + \sum_{i=1}^n x_i v_{ij} \quad (1)$$

- applying activation function $Z_j = f(z_{inj})$ and sends this signal to all units in the layer above i.e., output units.

- Each output unit ($y_k, k = 1 \dots m$) sum its weighted input signals

$$y_{-ink} = w_{ok} + \sum_{j=1}^p z_j w_{jk} \quad (2)$$

- Applying activation function $y_k = f(y_{-ink})$
- Calculate error factor $\delta = \text{actual}(z) - \text{target}(y)$;
- Initialize predicted value $P_{\rho+i} = \text{target}(y)$;

/* ...Back-Propagation and calculate new weights w_k */

- Each output unit ($y_k, k = 1 \dots m$) receives a target pattern corresponding to an input pattern error information term is calculated as

$$\delta_k = (t_k - y_k) f'(y_{-ink})$$

- Each hidden unit ($z_j, j = 1 \dots n$) sums its delta inputs from units in the layer above

$$\delta_{-inj} = \sum_{k=1}^m \delta_j w_{jk}$$

- The error information term is calculated as

$$\delta_j = \delta_{-inj} f'(z_{-inj})$$

/*...Updating of Weights and Biases...*/

- Each output unit ($y_k, k = 1 \dots m$) updates its bias and weights ($j = 0 \dots p$) The weights correction term is given by $\Delta W_{jk} = \alpha \delta_k z_j$

- And the bias correction term is given by $\Delta W_{ok} = \alpha \delta_k$

- Therefore, $W_{jk}(new) = W_{jk}(old) + \Delta W_{jk}$, $W_{ok}(new) = W_{ok}(old) + \Delta W_{ok}$

- Each hidden unit ($z_j, j = 1 \dots p$) updates its bias and weights ($i = 0 \dots n$) the weights correction term $\Delta V_{ij} = \alpha \delta_j x_i$

- The bias correction term $\Delta V_{oj} = \alpha \delta_j$

- therefore, $V_{ij} = V_{ij}(old) + \Delta V_{ij}$, $V_{oj}(new) = V_{oj}(old) + \Delta V_{oj}$

Calculate MSE between series X_i and P_i ;

$$MSE = \left| \frac{1}{m} \left(\sum_{i=1}^m X_i - P_i \right) \right|$$

set $i = 1; j = 1$;

} while($MSE < 0.000001$);

V. SELECTION OF BPANN PARAMETERS

A. Initial Weights

It will influence whether the network reaches a global (or only a local) minima of the MSE and if so how rapidly it converges. It is found that, if initial weight is too large the initial input signals to each hidden or output unit will fall in the saturation region where the derivative of sigmoid has very small value ($f'(network) = 0$). If initial weights are too small, the network output to a hidden or output unit will approach zero, which then causes extremely slow learning. To get the best result the initial weights (and biases) are set to random number between 0 and 1. The initial weights (bias) can be done randomly and moreover there is a specific approach. The faster learning of a BPANN can be obtained by using Nguyen-Widrow (NW) initialization (Equation 3). Kumar, 2007 have found this method is designed to improve the learning ability of the hidden units [39].

$$\beta = 0.7(p)^{1/n}$$

where, n = number of input units, p = number of hidden units, β = scale factor.

B. Learning Rate (α)

It is found that, a high learning rate ' α ' leads to rapid learning but the weights may oscillate, while a lower learning rate leads to slower learning. Kumar, 2007, suggested that initially set ' α ' between lower value 0 and 1 and increase in order to improve performance. It is found that, $\alpha = 0.3$ is suitable for almost all of the cases and in this study [39].

C. Number of Hidden Layers

Phillip, 2003 found that, nearly all problems, one hidden layer is sufficient. Using two hidden layers rarely improves the model, and it may introduce a greater risk of converging to a local minima. And also there is no theoretical reason for using more than two hidden layers [39].

D. Number of input Vector (n)

It is observed that, the MAD (% of mean) is inversely proportional to n for all type of input data.

$$MAD \propto \frac{1}{n}$$

In this study, $n = 8$ is used to input eight climate parameter i.e. $x_1, x_2, \dots, x_8$ (see Table 1). However, more parameter may used as input to identify TMR as dependent parameter (karmakar et al., 2008, 2009) [15-21].

E. Neurons in Hidden Layer (p)

It has been found that, if number of neurons (p) increases in the hidden layer, the MAD (% of mean) between actual and predicted value increases (Karmakar et al., 2008, 2009). In other words, the relation between neurons (p) and MAD (% of mean) is shown as

$$MAD \propto p$$

$$MAD = cp$$

Where, c is error constant. It is verified that, if $p = 2$ then it would provide two fundamental benefits (1) MAD between actual and predicted values during training will be least and (2) Time complexity of network training process will be not as much of. Note that, more hidden neurons increases more unknown variables (i.e., trainable weights w_i) is required additional process time to optimized, it can be temporal timidity. Thus, two neurons $p = 2$ have been used in the BPANN. Finally, BPANN have been developed to predict TAR as shown in following Figure 4. Where eight input vectors i.e. $x_1, x_2, \dots, x_8$, in input layer are used to input eight independent variable. Two neurons in hidden layer ($z_1 \dots z_2$) are used.

VI. TRAINING OF BPANN: LOCAL AND GLOBAL MINIMA

The minimizing mean square error (MSE) of the network called parallel process or epoch by using above algorithm during the training period is shown in following Table 2 and Figure 5. The training started with initial set of weights between 0 and 1 at point 'P' where $MSE = 3.874818901696980000E-04$ using algorithm. Minimum $MSE = 2.297276501530190000E-04$ is encountered at point M_L after 10,000 epochs. M_L is called local minima. After this point the MSE is showing a constant trend. Generally, scientists are often stopping training process in this point for their specific applications. This is the only grounds of poor result. The often think that more training will not minimize the MSE or may be difficult or not possible. Table 2 and Figure 5 shown some observations here. However It is cleared that in day 3, after 50,00000 epochs, the MSE becomes $1.98445160845550000E-04$ marked by the point M_G (Fig. 5). This point is called global minima called maximum trained network point. In the literature the various authors are clearly declared that, obtaining this point is almost difficult or temporal timidity. But this observation it is clear that this concept is not true. No one should talk about such type of misconception. After this point MSE is exhibiting an increasing trend and is considered as over training of the network. This experiment is done with one GB RAM with IBM

Pentium machine. Total time required to process is three days. The experimental result is shown in Table 2 and Figure 5. The processes are developed using Java mathematical package.

Fig. 4 : Architecture of Designed BPANN.

Table 2. Observation of Reducing MSE during Training Period

Observation	Time	MSE
Day 1	4:00	3.874818901696980000E-04
	5:00	3.859082589904380000E-04
Day 2	10:00	2.321827738964890000E-04
	11:00	2.312343862255940000E-04
	12:00	2.304870132009410000E-04
	1:00	2.297276501530190000E-04
	2:00	2.286104885748140000E-04
	3:30	2.275945237306910000E-04
	4:00	2.271197027979140000E-04
Day 3	10:00	1.98445160845550000E-04
	12:00	1.994576586384550000E-04
	2:00	2.116754378654120000E-04

VII. RESULTS AND DISCUSSIONS

In order to check the performances of BPANN during training period, it is trained with the 13 years (1993-2004) training dataset. The performances is shown in Table 3,5 and Figure 6. The BPANN presents low MAD (i.e., 0.0106) and a high CC (i.e., 0.86) between actual and predicted values as shown in Figure 6. This shows strong association between dependent total rainfall, and independent parameters $x_1, x_2, \dots, x_8$. In order to check the BPANN performance during the

testing/independent period with new data, it was tested with the 04 years (2006-2009) of the test data. The performance is shown in Table 4,5 and Figure 7. The BPANN again presented low MAD and a high CC between actual and predicted values in this period. The result shows strong association between dependent parameter rainfall, and independent variables $x_1, x_2, \dots, x_8$ during the testing period as well. Other than that, MAD at training as well as testing period exceptionally less than the SD as shown in Table 5. Thus, it is concluded that the proposed architecture of BPANN and its optimized weights for the variable total rainfall is very appropriate in internal dynamics of total rainfall variable. However, some limitations of BPANN during the training period have been depicted is presented in the following section.

Fig. 5 : Minimizing MSE: Local Minima (M_L) and Global Minima (M_G)

Fig. 6 : Performance of the BPANN in Training Period (1993-2005)

Fig. 7 : Performance of the BPANN in Independent Period (1996-2009)

Table 3. Performance of the BPANN in Training Period (1993-2005)

Year	Actual Total Annual rainfall (in mm.) Normalized Form	Actual Total Annual rainfall (in mm.) De-normalized Form	Predicted Total Annual Rainfall (in mm.) Normalized Form	Predicted Total Annual Rainfall (in mm.) De-normalized Form
1993	0.7102974	1303.10	0.7002873	1204.40
1994	0.7074762	1274.60	0.7001813	1203.39
1995	0.7235890	1445.20	0.7107810	1308.04
1996	0.7023089	1223.80	0.7007918	1209.22
1997	0.6764550	994.00	0.6910319	1118.84
1998	0.7107379	1307.60	0.7080345	1280.20
1999	0.7189336	1393.90	0.7095379	1295.37
2000	0.7075761	1275.60	0.7075301	1275.14
2001	0.7273262	1487.65	0.7146373	1348.04
2002	0.6810714	1032.30	0.7046470	1246.57
2003	0.7408898	1652.00	0.7196289	1401.45
2004	0.7384596	1621.30	0.7206172	1412.26
2005	0.7260216	1472.70	0.7216772	1423.93

Table 4. Performance of the Model in Independent Period (2006-2009)

Year	Actual Total Annual rainfall (in mm.) Normalized Form	Actual Total Annual rainfall (in mm.) De-normalized Form	Predicted Total Annual Rainfall (in mm.) Normalized Form	Predicted Total Annual Rainfall (in mm.) De-normalized Form
2006	0.7012332	1213.45	0.7158403	1360.74
2007	0.7195046	1400.10	0.7260217	1472.70
2008	0.6946861	1152.00	0.7063202	1263.08
2009	0.7300391	1519.20	0.7275417	1490.13

Table 5. Performance in Tainting and Independent Period

Training Period			Independent /Testing Period		
SD	MAD	CC	SD	MAD	CC
14.48	8.20	0.87	12.79	6.81	0.94

VIII. CONCLUSIONS

This study focuses on the capabilities of ANN in the prediction of high internal dynamics long-range rainfall variables showing the advantages and disadvantages compiled. The advantage of this study is that it has been cleared that ANN is sufficiently suitable for identification of internal dynamics of highly dynamic non linear system and predictions. Disadvantage is that, it required many training effort, possibly over fitting, and a temporal nervousness. It is found that, the back-propagation using gradient descent (algorithm) often converges very slowly or not at all. On large-scale problems its success depends on user-specified learning rate and momentum parameters. There is no automatic way to select these parameters, and if incorrect values are specified the convergence may be exceedingly slow, or it may not converge at all. While back-propagation with gradient descent still is used in the model development, it is no longer considered to be the best or fastest algorithm. Instead of gradient descent, conjugate gradient algorithm can be used as a future work to adjust weight values using the gradient during the back-propagation of errors through the ANN. Compared to gradient descent; the conjugate gradient algorithm takes a more direct path to the optimal set of weight values. Usually, conjugate gradient is significantly faster and more robust than gradient descent. Conjugate gradient also does not require the user to specify learning rate and momentum parameters. In addition, performance of the RBFANN, GRANN can also be compared with the applied BPANN as a future work.

REFERENCES

- [1]. Journel, A. G., Huijbregts, C. J., 1978: Mining geostatistics. Academic Press, New York.
- [2]. Hornik, K., Stinchcombe, M., White, H., 1989: Multilayer feedforward networks are universal approximators. *Neural Networks*, 2, 359-366.
- [3]. Goovaerts, P., 1997: Geostatistics for natural resources evaluation, Oxford University Press, New York, Oxford.
- [4]. Demyanov, V., 1998: Neural Network Residual Kriging Application for Climatic Data", *J. Geographic Information and Decision Analysis*, 2 (2), 215-232.
- [5]. Huang, Y., Wong, P. M. Gedeon, T. D., 1998: Prediction of reservoir permeability using genetic algorithms, *AI Applications*, 12(1-3), pp. 67-75.
- [6]. Ricardo, M. T., Palutikof, J.P., 1999: Simulation of Daily Temperatures for Climate Change Scenarios over Portugal: A Neural Network Model Approach, *Climate research (Clim., Res.)*, 13, 45-59.
- [7]. Guhathakurta, P., 1998: A Hybrid Neural Network Model for Long Range Prediction of All India Summer Monsoon Rainfall, *Proc. WMO international workshop on dynamical extended range forecasting*, Toulouse, France, November 17-21, 1997, PWPR No. 11, WMO/TD. 881, 157-161.
- [8]. Guhathakurta, P., 1999: Long Range Forecasting Indian Summer Monsoon Rainfall By Principle Component Neural Network Model, *J. Meteor. & Atmos. Phys.*, 71, 255-266.
- [9]. Guhathakurta, P., 1999: A Short Term Prediction Model for Surface Ozone At Pune: Neural Network Approach, *Vayu mandal*, 29(1-4), 355-358.
- [10]. Guhathakurta, P., 1999 : A Neural Network Model for Short Term Prediction of Surface Ozone at Pune, *Mausam*, 50(1), 91-98.
- [11]. 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*, Springer-Verlag, Austria.
- [12]. Guhathakurta, P., 2000: New Models for Long Range Forecasts of Summer Monsoon Rainfall over North West and Peninsular India, *Meteorology and Atmospheric Physics*, 73 (3), 211-255.
- [13]. Guhathakurta, P., 2006: Long-Range Monsoon Rainfall Prediction of 2005 for the Districts and Sub-Division Kerala With Artificial Neural Network, *Current Science*, 90(6), 773-779.
- [14]. Rajeevan, M., Guhathakurta, P., Thapliyal, V., 2000, "New Models for Long Range Forecasting of Monsoon Rainfall over Northwest and Peninsular India". *Meteorology and Atmospheric Physics*, 73, 211-225.
- [15]. 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(2), 41-45.
- [16]. Karmakar, S., Kowar, M. K., Guhathakurta, P., 2009: Evolving and Evaluation of 3LP FFBP Deterministic ANN Model for District Level Long Range Monsoon Rainfall Prediction", *J. Environmental Science & Engineering*, 51(2), 137-144
- [17]. Karmakar, S., Kowar, M. K., Guhathakurta, P., 2008: Development of Artificial Neural Network Models for Long-Range Meteorological Parameters Pattern Recognition over the Smaller Scale Geographical Region-District", *IEEE Xplore, IEEE Computer Society*, Washington, DC, USA.
- [18]. Karmakar, S., Kowar, M. K., Guhathakurta, P., 2008: Development of an 8-Parameter Probabilistic Artificial Neural Network Model for Long-Range Monsoon Rainfall Pattern Recognition over the Smaller Scale Geographical Region - District, *IEEE Xplore, IEEE Computer Society*, Washington, DC, USA.
- [19]. Karmakar, S., Kowar, M. K., Guhathakurta, P., Development of a 6 Parameter Artificial Neural Network Model for Long-Range July Rainfall Pattern Recognition over the Smaller Scale Geographical Region -District, *CSVTU Research Journal*, pp. 36-40.
- [20]. Karmakar, S., Kowar, M. K., Guhathakurta, P., 2009: Long-Range Monsoon Rainfall Pattern Recognition & Prediction for the Subdivision 'EPMB' Chhattisgarh Using Deterministic

- & Probabilistic Neural Network”, IEEE Xplore, IEEE Computer Society, Washington, DC, USA.
- [21]. Karmakar, S., Kowar M. K., Guhathakurta P., 2009, Spatial Interpolation of Rainfall Variables using Artificial Neural Network”, ACM DL, 547-552.
- [22]. Snell, S. E., 2000, Spatial Interpolation of Surface Air Temperatures Using Artificial Neural Networks: Evaluating Their Use for Downscaling GCMs, *Journal of Climate*, 13, 886-895.
- [23]. Antonic, O., Križan, J., Marki, A., Bukovec, D., 2001: Spatio-temporal interpolation of climatic variables over large region of complex terrain using neural networks, *Ecological Modelling*, 138, 255-263.
- [24]. Rigol, J.P., Jarvis, C.H., Stuart, N., 2001: Artificial neural networks as a tool for spatial interpolation, *International Journal of Geographical Information Science*, 15, 323-343.
- [25]. Koike, K., Matsuda, S., Gu, B., 2001: Evaluation of Interpolation Accuracy of Neural Kriging with Application to Temperature-Distribution Analysis. *Mathematical Geology*, 33(4), 421-448.
- [26]. Bryan, B.A., Adams, J.M., 2001: Quantitative and Qualitative Assessment of the Accuracy of Neurointerpolated Annual Mean Precipitation and Temperature Surfaces for China, *Cartography*, 30 (2), Perth, Australia.
- [27]. Bryan, B.A., Adams, J. M., 2002: Three-Dimensional Neurointerpolation of Annual Mean Precipitation and Temperature Surfaces for China, *Geographical Analysis*, 34 (2), 93-111.
- [28]. Li, B., 2002: Spatial Interpolation of Weather Variable using Artificial neural Network, MS Thesis, University of Georgia.
- [29]. Iseri, Y. G. C. Dandy., Maier, R., Kawamura, A., Jinno, K., 2002: Medium Term Forecasting of Rainfall Using Artificial Neural Networks, Part 1 background and methodology, *Journal of Hydrology*, 301 (1-4), 1834-1840.
- [30]. Silva, A. P., :2003, Neural Networks Application to Spatial Interpolation of Climate Variables, Carried Out By, STSM on the Framework of COST 719 ZAMG.
- [31]. Timonin, V., Savelieva, E., 2005: spatial prediction of radioactivity using GRNN”, *Applied GIS*, Monash University Press, 1(2), 1-19.
- [32]. Attorre, F., Alfo, M., Sanctis, M., Francesconi, F., Bruno, F., 2007: Comparison of interpolation methods for mapping climatic and bioclimatic variables at regional scale, *Int. J. Climatol.*, Royal Meteorological Society, Published online in Wiley InterScience (www.interscience. wiley.com) DOI: 10.1002/joc.1495.
- [33]. Chattopadhyay, S., Chattopadhyay, G., 2008: Identification of the best hidden layer size for three layered neural net in predicting monsoon rainfall in India, *J. Hydroinformatics*, 10 (2), 181-188.
- [34]. Hung, N. Q., Babel, M. S., Weesakul, S., Tripathi N. K., 2009: An artificial neural network model for rainfall forecasting in Bangkok, Thailand, *Hydrology and Earth System Sciences*, 13, 1413-1425.
- [35]. Mendes, D., Marengo, J., 2010: South America downscaling: using spatial artificial neural network, *Geophysical Research Abstracts*, 12, EGU2010-9475.
- [36]. Sivapragasam, C., Arun, V.M., Giridhar, D., 2010, A simple approach for improving spatial interpolation of Rainfall using ANN, *Meteoro Atmos Phys*, Springer, 109, 1-7.
- [37]. Ghazanfari, S., Alizadeh, A., Farid, A., Bannayan, M., 2011: Comparison the PERSIANN model with interpolation methods to estimate daily precipitation (A case study: North-Khorasan, Iran), *Geophysical Research Abstracts*, 13, EGU2011-12743.
- [38]. 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, 1, 318-362.
- [39]. Kumar, S., : Neural Network Computer Engineering Serie, The McGraw-Hill, New Delhi, 2007, 104-152.