iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 **Issue:** VII **Month of publication:** July 2019

DOI: http://doi.org/10.22214/ijraset.2019.7048

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue VII, July 2019- Available at www.ijraset.com

A New Approach for Rainfall Prediction using Artificial Neural Network

Deeksha Tripathi[1], Rajendra Kumar Gupta[2]
[1, 2]*Department of Computer Science & Information Technology, MITS, Gwalior (M.P.) 474005, India*

Abstract: Rainfall is considered as one of the major components of the weather forecasting. In the current world climate change, the accuracy of rainfall forecasting model is very important factor. Rainfall affects the drought and flood situation. India is an agricultural country. Rainfall are also affects the area of agriculture. We have considered the monthly rainfall data of East Madhya Pradesh, India from 1901-2017. Afterwards, in this paper, to evaluate an actual prediction of rainfall forecasting, were used an Artificial Neural Network (ANN) by the Cascade-forward Back propagation Neural Network (CFBPNN) technique. In this study, to trained the training data of the rainfall information using 2 hidden layers of CFBPNN technique, with three different epochs: [2-50-10-1] with epoch fix to 500, [2-50-20-1] with epochs fix to 1000 and 1500. To measure the performance of the developed architecture, the mean square error (MSE) algorithm is employed using CFBPNN. The experimental results showed that [2-50-10-1] architecture with epoch fix to 500 and learning rate 0.1, produced a good performance result with the value of MSE was 0.0063408. Eventually, CFBPNN algorithm has provided a best accuracy model to predict monthly rainfall in East Madhya Pradesh, India.

Keywords: ANN, CFBPNN, Monthly Rainfall, MSE

I. INTRODUCTION

In the earth's environment, Rainfall is one of the key parameter. The essential requirement of rainfall forecast data is to endorsement water resources management specifically which is concerned to be change the global climate in different tropical regions [1]. In the areas of climate and weather forecasting, Rainfall Prediction has a broader domain. On the attention of the scientific community, industries, governments, and risk management entities, were describe the rainfall prediction, which affects many human activities like construction power generation, forestry and tourism, agricultural production, among others. Different types of forecasting the rainfall techniques are provided in India, because India is an agricultural country and the rainfall and humidity is the main factor of agriculture [2]. In India, Rainfall Prediction are depends on the mainly two perspectives: Dynamical method and Empirical method. Using the physical models, Dynamical methods are used to rainfall prediction which is based on systems of equations that predict the evaluation of the Global Climate System (GCS), were response to initial atmospheric condition. The empirical approaches used for rainfall prediction are Artificial Neural Network (ANN), Linear Regression, Decision Tree Algorithm, Fuzzy Logic, and group method of data handling. Empirical method is works on analysis of historical data and it is related to a variety of atmospheric parameter [3].

Mostly, farmers are depending on the monsoon for fertilizing their crop, it is requirement for predicting the rainfall to be having advanced knowledge of actual rainfall. In India, most of the states are facing the problem of the flood whereas the some of the states are suffered from the situation of drought. To evaluating the flood and drought situation in advance, a precise and effective rainfall forecast model must be required. This type of model of rainfall forecasting will assist to better handle the worst circumstances caused by flooding or drought. This model also provides sufficient time to anticipate rainfall to make appropriate provisions to save life, transport, procure and supply food and medications [4].

Data mining relates to information extracted or "mined" from big quantities of information. We understand that gold mining from rocks or sand is not called rock or sand mining, but gold mining. Data mining should therefore have been more properly described as "Knowledge extraction from Data", which is sadly rather lengthy. "Knowledge Mining" is a shorter word from big quantities of information may not indicate the emphasis on mining. Nevertheless, mining is a word that characterizes a method that discovers from a lot of raw material a tiny number of valuable nuggets. The data mining uses complex data structure, Artificial Intelligence and Algorithms. It uses complete to learn from prior understanding and to recognize hidden patterns of information and to deliver realistic outcomes along with rationalization. Knowledge Discovery in the KDD database is synonymous with data mining [5, 6].

A. Types of Forecasting

The four categories are described the rainfall forecasts [7, 8]:

1) *Now Casting:* In now casting, datails about the current weather and rainfall forecasts up to a few hours further are given.
2) *Short Range Forecasts:* In short range forecasts, rainfall and weather forecasts (mainly rainfall) are predicted in each successive 24 hrs. Intervals may be predicted up to three days.
3) *Medium Range Forecasts:* In Medium range forecasts, Average rainfall conditions and also the weather on a daily basis is prescribed and the more and more lesser details and accuracy than that for short range forecasts. Medium range forecasts are predicts the rainfall from four to ten days.
4) *Long range / Extended Range Forecasts:* In the long range forecasts, there is no rigid definition which had the range from a monthly to a seasonal forecasts. Large range forecasts are predicts the rainfall more than 10 days to a season.

II. LITERATURE REVIEW

Several studies are being conducted in India to determine temperature and rainfall changes and their connection with climate change. However, researchers used distinct lengths of information and now studies have been reported using information over a decade. All such studies showed a country-scale warming trend. It was better to predict rainfall estimates using long-term information sequence. Various papers described to predicting and analysis of monthly rainfall was suggesting different algorithms and methods. These are described below:

Archana Nair et al. [9] applied a nonlinear technique i.e., Artificial Neural Network. Authors has been developed Global Climate Models (GCMs). In this study, GCM are considered from the National Centre for Environmental Prediction (NCEP) and the International Research Institute (IRI). The monthly and seasonal rainfall information had been predicted over the Indian domain of different tropical region. Nowadays, the scientist of the meteorological community are being faced many challenges in the rainfall prediction concerned. In this study, two types of dataset are required: GCM predicted hindcast dataset, these type of dataset are collected from observational dataset and International Research Institute (IRI), these type of dataset are collected from Indian Meteorological Department (IMD). This analysis is developed by using double cross-validation and simple-randomization technique on dataset. The performance of coupled and uncoupled are enhanced the prediction of rainfall of the individual months using the ANN technique.

Seyed Amir Shamsnia et al. [10] developed a system for weather parameters including precipitation, monthly temperature and relative humidity using random mechanism (i.e., ARIMA). The authors have been collected 20 years long dataset which is statistics of relative humidity and monthly average temperature in Iran at Abadeh station which is used by Interactive Time Series Modelling (ITSM) analysis software. They designed ARIMA (0 0 1) (1 1 1)12 for precipitation, ARIMA(2 1 0)(2 1 0)12 for monthly average temperature and ARIMA (2 1 1) (1 1 0)12 for relative humidity.

Md. Mahsin et al. [11] has become a major tool to predicting the monthly rainfall information using Time Series Analysis and Forecasting in different application for meteorological phenomena. Authors had introduced a model by Box and Jenkins methodology and Autoregressive Integrated Moving Average (ARIMA). In this research, they have 30 years monthly rainfall information from 1981-2010 at Dhaka Station, Bangladesh. They designed the ARIMA(0 0 1)(0 1 1) model for forecasting the Monthly Rainfall.

A. Abraham et al. [12] introduced a challenging task which is Long Term rainfall prediction where scientific communities are facing the major environment problem of global warming. In this research, authors had been used Multivariate Adaptive Regression Splines (MARS) and some soft computing techniques: Artificial Neural Networks (ANNs), Scaled Conjugate Gradient Algorithm (SCGA), Adaptive Basis Function Neural Network (ABFNN), General Regression Neural Network (GRNN) and Neuro-Fuzzy Systems.

N. Prasad et al. [13] proposed Decision Tree algorithm using SLIQ (Supervised Learning in Quest). This study generated some classification rule for rainfall prediction and the average efficiency has been found to be 72.3%.

R. Adhikari et al.[14] suggested the NAR-based Feed-Forward Neural Network model for time series forecasting was also an significant predictive result.

Genetic Algorithms [15] and Artificial Neural Networks [16, 17, 18, 19], Multi-Layer Perceptron (MLP), Functional Link Artificial Neural Network (FLANN) and Legendre Polynomial Equation (LPE) [20], Multiple Linear Regression (MLR) methods [21] have been implemented to predict rainfall.

III.MODEL DEVELOPMENT

Figure 1 shows the significant steps engaged in the information flow sequence, as well as the creation of ANN prediction models, and the results discovered at different steps. It should be well known that a subset of the ten steps shown by Jakeman et al. (2006)[22] depicted here the measures of model growth covering the general scientific process, including a review of the hypothesis, the collection of suitable observations and information, and 16 hypothesis formulations. The first stage involved in the model design process described here is to choose from the accessible information and suitable model output(s) from a set of prospective model input factors (i.e. the variable(s) to be forecast). Even if ANNs are data-driven models, it depends on the model to choose which input variables should be included as part of the process of model development. Based on information accessibility or a priori knowledge, this choice can be enforced. The results create the "Selected (Unprocessed) Data" This should be well known that the complete amount of nodes in the output layer was also discovered once the inputs of the models were chosen. Next, unprocessed data consisting of measured values of the model output variable(s) and potential model input variable(s) must be processed (e.g. scaled, lagged) in order to make the data in an appropriate form for the subsequent steps of the model development process. Once the prospective model's processed database inputs and outputs ("Selected Data (Processed)") have been accumulated, the real model can be constructed. All models of ANN forecast are depicted as follows:

$$Y = f(X,W) + \varepsilon \qquad (1)$$

Where, Y = Model vector outputs,

X = Model vector inputs,

W = Model vector parameters (connection weights),

f(•) = Functional relationship between model outputs, inputs and parameters,

ε = Model errors vector

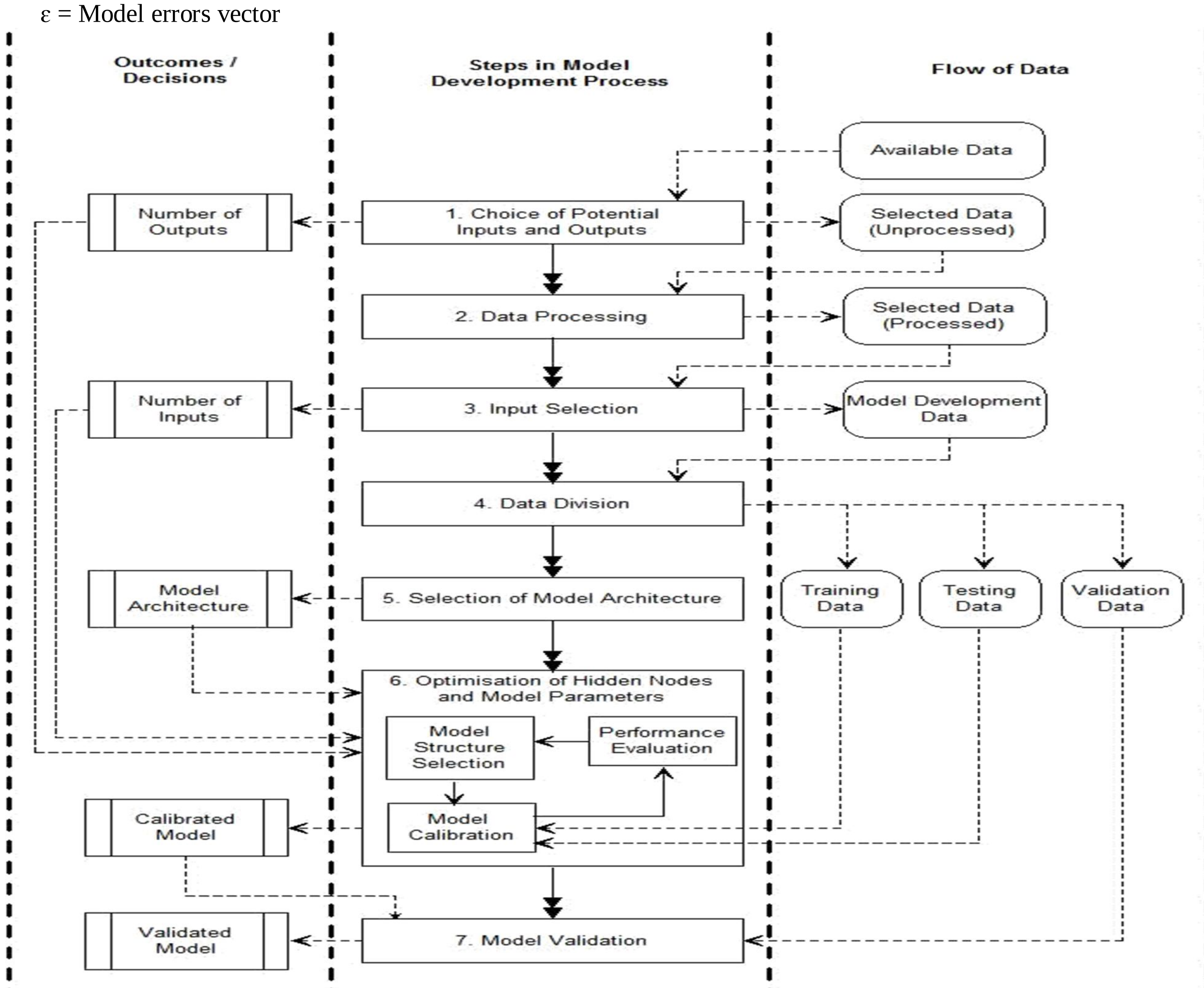


Figure 1: Different steps in ANN Model Development

Accordingly, the form of the functional relationship (f(•)) with the model inputs vector (X) to develop an ANN model. It is governed by the architecture of the neural network (e.g. multilayer perceptron) and geometry (e.g. number of nodes and hidden layers, type of transfer function) and the vector (W) model parameters. The link and bias weights have been included, need to be defined. For each set of model inputs, it had to be repeated in Steps 5 and 6 of **Figure 1**. The number of model outputs and the amount of nodes in the ANN model input layer when the vector is chosen for the model outputs. Model Development Data is subsequently split into two sub-sets: calibration subset and validation sub-set. Using the calibration information, the unknown model parameters (connection weights) are estimated and the calibrated model assessment is validated using the validation information on an autonomous information set.

IV. RESEARCH METHOD AND PROCESS

This research was proposed to use the CFBPNN algorithm to predict rainfall information by studying and analyzing previous information patterns in order to achieve more precise predictive outcomes with minimal error. CFBPNN is also briefly defined.

A. Data Normalization and Separation

The range of the data are high i.e., 0 to 713. So data are transforms into the short range in the data normalization step. ANN is based on the real world data. Normalization are very useful for the neural network because real data are obtained from experiments and analysis so most times are distant from each other [23]. The effect is great because the common activation functions are used for predictive accuracy. Updating weights are less distant from each other then minimize the MSE value. Some inputs to ANN might be not defined in the naturally manner range of values. In this study, the data are normalized in the range of 0.1 to 0.8 using Eq. 2.

$$\hat{x}_i = \frac{0.8*(x_i - x_{min})}{x_{max} - x_{min}} + 0.1 \quad (2)$$

Where $\hat{x}_i$ is the normalized value, x_i is the observation value, x_{max} represents the highest value of information collected and x_{min} represents the lowest value of information collected. After the data have been normalized, normalized data are separated into three groups, from 1901 to 1977 rainfall data will be used as the training data (65%), from 1978 to 2010 rainfall data will be used as the testing data (27%), and from 2011 to 2017 rainfall data will be used as the validation data (8%).

B. Cascade-Forward Back Propagation Neural Network

Cascade-forward back propagation neural network (CFBPNN) is similar to Feedforward neural network in all its operations with slight difference in its architecture. In Feedforward neural network the neurons of one layer can participate in computation and weight updating only for the next layer whereas in Cascade-forward backpropagation neural network the neurons of one layer involve in computation and weight updating of all the layers ahead [24]. It is a type of direct relationship created between input and output while indirect relationship is established in FFNN association between input and output. **Figure 2** shows CFBPNN. CFBPNN architecture has been intended to determine the quantity of information input, hidden and output layers and parameters.

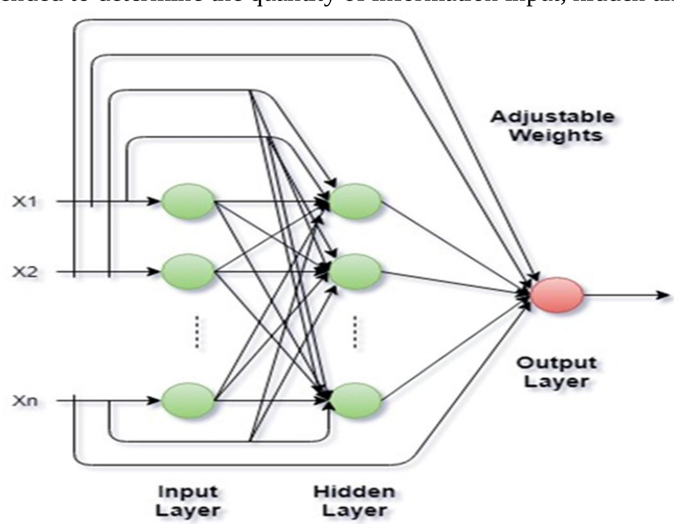


Figure 2: Cascade-forward back propagation neural network

C. Testing And Prediction

The purpose of the test is to determine the amount of precision of CFBPNN in future predicting monthly rainfall information using statistical method.

V. EXPERIMENTAL

A. Determining Samples Of Training And Test

Using the CFBPNN for large-scale information, rainfall information gathered from the Indian Meteorological Department (IMD) to obtain precise forecast outcomes and evaluation in the coming year [25], have been classified into three components, namely the training, testing and validation data. Using this experiment, 117 years rainfall information from 1901-2017 (1404 samples data series) which had been belongs from the East Madhya Pradesh, India, were used Table 1 and Figure 3. In the data normalization process, data are transform into the small range which is 0.1 to 0.8. After that normalized data are divided into three parts which is training data: 924 samples (65%) or 1901-1977, testing data: 396 samples (27%) or 1978-2010, and validating data: 84 samples (8%) or 2011-2018. According to the ANN model development, data had been governed, and data are used for the neural network with a predefined function $p = [p(t-2), p(t-1)]$ were used to evaluated the input layers and $p(t)$ were used to evaluated the output layer, Table 2. Then, the neural network architecture would be comprised with two hidden layer. Tansig, Logsig and Purelin activation function were used in the model from input to hidden layer that were also used for the hidden layer to the output layer. [26]. Levenberg – Marquardt algorithm is mainly built to lesser the sum-of-square error function which is defined by Eq. 3.

$$E = \frac{1}{2}\sum_{p=1}^{n}\sum_{k=1}^{m}\left(y_{pk} - \hat{y}_{pk}\right)^2 \quad (3)$$

Where, n is the number of observations, m is the total units of output, y is the target output and $\hat{y}$ is the output which are predicted by proposed model. Furthermore, MSE was used to evaluate the degree of accuracy of forecasting using CFBPNN. MSE is used as the performance criteria for deciding when the stop the training process. The error of training and validation set was monitored by the MSE error function. The MSE error function is defines by Eq. 4.

$$MSE = \frac{1}{N}\sum_{P=1}^{p}\sum_{i=1}^{N}(t_{pi} - y_{pi})^2 \quad (4)$$

Where, t_{pi} = Predicted value for data point i, y_{pi} = Actual value for the data point i and N = Total number of data points.

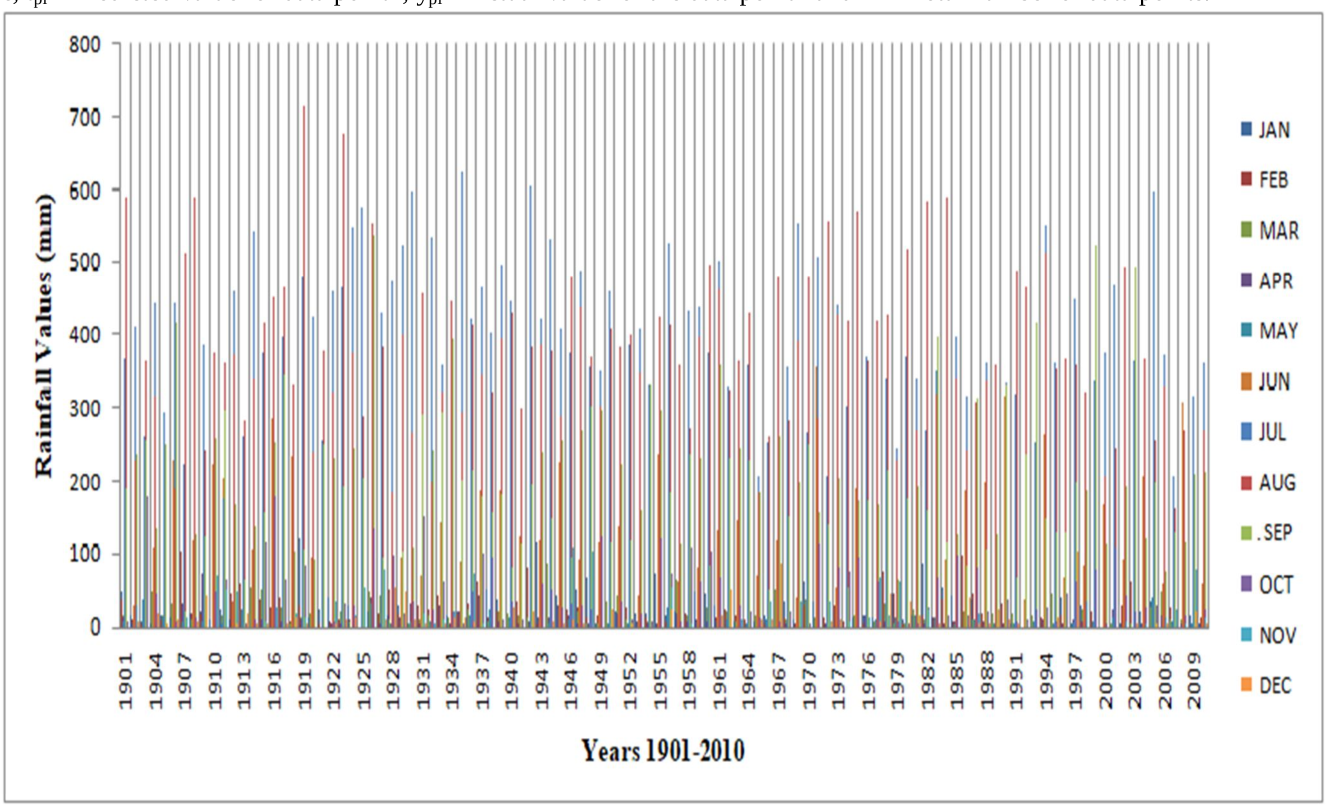


Figure 3: Plot of real rainfall data of East Madhya Pradesh from 1901-2010

TABLE 1
RAINFALL DATA 1901-2010

Years	1901	1902	1903	……	……	…..	2008	2009	2010
Jan	48.5	14.9	5.6	…..	…..	…..	1.4	14.8	2.9
Feb	38.1	8.9	2.9	…..	…..	…..	4.1	1	10.4
Mar	15.7	0	0.3	…..	…..	…..	8.9	1.8	0.1
Apr	10.7	3.6	0.9	…..	…..	…..	2.5	2.7	0.7
May	6.2	2.7	37.5	…..	…..	…..	5.3	3.7	1.7
Jun	61	28	67.5	…..	…..	…..	307.6	35.3	58.4
Jul	367.5	411.9	261.4	…..	…..	…..	268.2	317.6	363.3
Aug	589.2	227	366.7	…..	…..	…..	270.2	191.3	271.9
Sep	189.9	236.6	257.4	…..	…..	…..	116.1	208.7	213.4
Oct	5.9	17	177.9	…..	…..	…..	15.5	68.9	23.6
Nov	0	27.6	0	…..	…..	…..	3	79.2	10.9
Dec	0	6.1	0	…..	…..	…..	0.4	19	3.3

TABLE 2
RAINFALL SAMPLES OF INPUT AND OUTPUT NEURON FOR NEURAL NETWORK

	Group 1	Input neurons P = [p (t-2),p(t-1)]		Target neurons		Group 2	Input neurons P = [p (t-2),p(t-1)]		Target neurons
		t-2	t-1	t			t-2	t-1	t
Training Group	1	0.112	0.104	0.154	Testing Group	925	0.175	0.105	0.125
	2	0.104	0.154	0.143		926	0.105	0.125	0.183
	…	…	…	…		…	…	…	…
	…	…	…	...		…	…	…	…
	923	0.288	0.169	0.175		1319	0.339	0.126	0.112
	924	0.169	0.175	0.105		1320	0.126	0.112	0.104

B. *Flow of CFBPNN for Prediction*

The three main phases are consisted by the CFBPNN technique [27]:

1) Feedforward propagation phase of the input pattern
2) Counting and calculates the error
3) Weight updation

After the Feedforward phase, the errors (the difference from the output to the target) are calculated with the running process, this phase is also known as the error counting phase. After the error counting phase, weights are recalculated to update the weights. Phase of update weight is carried out up to no iteration and error finding process pause, according to the specified termination condition. In this Study, we discuss about the conjugate gradient optimization method for the adjustments of weight of the CFBPNN model.

Suppose that the set of all network weights belongs to a weight vector ω, some lengths s, and the objective function is $e = \frac{1}{2}(x_t - \hat{x}_t)^2$. Where, the target value is x_t and the expected value is $\hat{x}_t$. Defined Q is the positive definite matrix of size s×s where QT = Q. The Conjugate Gradient optimization algorithm stages are defined as follows:

a) *Step 1:* Set k = 0, select the initial parameter Ω(0)
b) *Step 2:* Evaluate the gradient of the initial weight

$$g^{(0)} = \frac{\partial e}{\partial \omega^{(0)}} = \left.\frac{\partial e}{\partial \omega}\right|_{\omega=\omega^{(0)}} = \left[\frac{\partial e}{\partial \omega_1^{(0)}} \cdots \frac{\partial e}{\partial \omega_s^{(0)}}\right]^T$$

If g(0) = 0 then stop, and it obtained the optimal weight = Ω(0). Else, set d(0) = g(0).

c) *Step 3:* Evaluate $\alpha_k = \arg min_{\alpha \geq 0}\, e\left(\omega^{(k)} + \alpha d^{(k)}\right) = -\frac{g^{(k)T} d^{(k)}}{d^{(k)T} Q d^{(k)}}$

d) *Step 4:* Evaluate $\Omega^{(k+1)} = \Omega^{(k)} + \alpha_k d^{(k)}$

e) *Step 5:* Evaluate $g^{(k+1)} = \frac{\partial e}{\partial \omega^{(k+1)}}$, if $g^{(k+1)} = 0$ stop and the optimal weight is $\omega^{(k+1)}$

f) *Step 6:* Evaluate

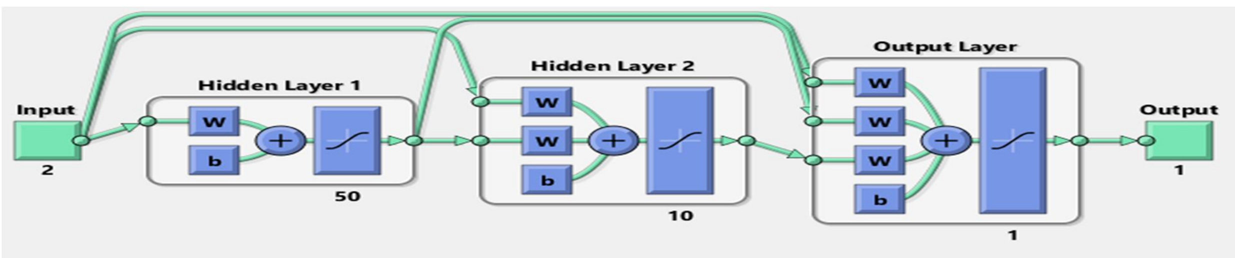
$$\beta_k = \frac{g^{(k+1)T} Q d^{(k)}}{d^{(k)T} Q d^{(k)}}$$

g) *Step 7:* Evaluate $d^{(k+1)} = -g^{(k+1)} + \alpha_k d^{(k)}$

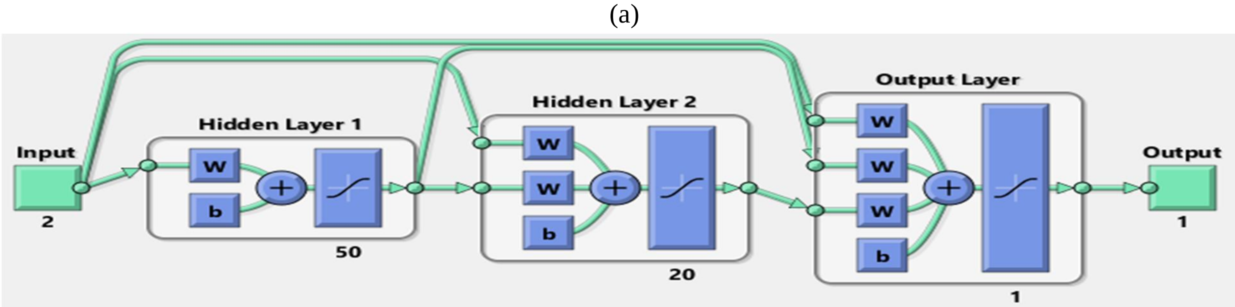
h) *Step 8:* k=k+1; go to Step 3.

The iteration process for weight searching on CFBPNN is usually called the epoch. Suppose that the maximum amount of epochs allowed is K, if the requirement for the termination criterion has not been met until epoch k = K then the iteration process will be stopped.

In this experiment, a predefined function $p = [p(t-2), p(t-1)]$ were used to evaluated the input layers and $p(t)$ were used to evaluated the output layer, were the values for t-2, t-1 and t were taken from **Table 2**. In this model, two hidden layers are implemented, first hidden layer have 50 neurons and second hidden layer have 10 and 20 neurons. **Figure 4** are represents the architecture of CFBPNN.



(a)



(b)

Figure 4: (a) A typical of CFBPNN by using 2-50-10-1 architecture, (b) A typical of CFBPNN by using 2-50-20-1 architecture

VI. RESULTS AND DISCUSSIONS

In this section, the test outcomes of monthly rainfall information are described using the CFBPNN based on two distinct architectures shown in Figure 4. 2-50-10-1 is the first architecture using CFBPNN technique. In this architecture representation, the first number (i.e., 2) represents the number of neurons present in input layer, second number (i.e., 50 and 10) represents the number of neurons present in the hidden layer and at the final position (i.e., 1) represents the number the neurons in the output layer. After that, learning-rate and epoch had been fix to 0.1 and 500, respectively. MSE was used to evaluate the degree of forecast precision that is predefined as shown in Table 3, to compare the expected output with the real output. MSE was 0.0063408 for the first CFBBPNN architecture. Then, 2-50-20-1 is the second architecture using CFBPNN technique. In this architecture representation, the first number (i.e., 2) represents the number of neurons present in input layer, second number (i.e., 50 and 20) represents the number of neurons present in the hidden layer and at the final position (i.e., 1) represents the number the neurons in the output layer. It was used in two distinct epochs fix to be 1000 and 1500 with learning rate fix to be 0.1. It has two different results which are shown in Table 3. It produced MSE were 0.0066908 and 0.0069920 in each. Figure 5 shows that the best proposed network used by the CFBPNN technique: 2-50-10-1 with 500 epochs and 0.1 learning rate. Figure 5(a) shows that the best performance result of training which is obtained MSE was 0.0063408. Figure 5(b) shows that the best regression result which is obtained Regression value

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue VII, July 2019- Available at www.ijraset.com

was 0.97178. It implies that by using the equation Y = 0.98*Target+0.004, the predicting of monthly rainfall information training had the best prediction accuracy. Then, Table 4 defined the error of epoch of testing data for CFBPNN models.

In this experiment, the period of the epoch time was also regarded as the good prediction accuracy. For each architecture, the epoch time training has encountered the highest efficiency, achieving certain times. In the first architecture, the time of epoch has been accomplished in 19 minutes and reached the epoch 500, but the performance of result was best. Meanwhile, in the second architecture, the time of epoch has been accomplished in 15 minutes and reached the epoch 1000. However, in the third architecture, it demonstrated longer epoch time have been accomplished in 55 minutes and reached the epoch 1500. On the other hand, the analysis results of second and third architecture were still not good. This implies that the accuracy of the architecture of the CFBPNN architecture also impacts the efficiency of the period of the iterations.

In this proposed work, the best prediction accuracy performance result of CFBPNN models has been described in which the value of MSE < 0.009. Thus, the result of the second test architecture has gained a good value that was similar value of learning rate and the value of different epochs compares with third architecture. It demonstrates that CFBPNN efficiency is also affected by epochs. Epochs with values 1000 and 1500 therefore have bad values below a predetermined value. This implies that the 2-50-10-1 CFBPNN model with epoch fix to 500 and learning rate fix to 0.1 is an outstanding model for predicting monthly rainfall information in the future, it shown in Figure 6.

TABLE 3
RESULT OF TRAINING FOR CFBPNN MODEL

Model	Architecture	Epochs	MSE training
1	2-50-10-1	500	0.0063408
2	2-50-20-1	1000	0.0066908
3	2-50-20-1	1500	0.0069920

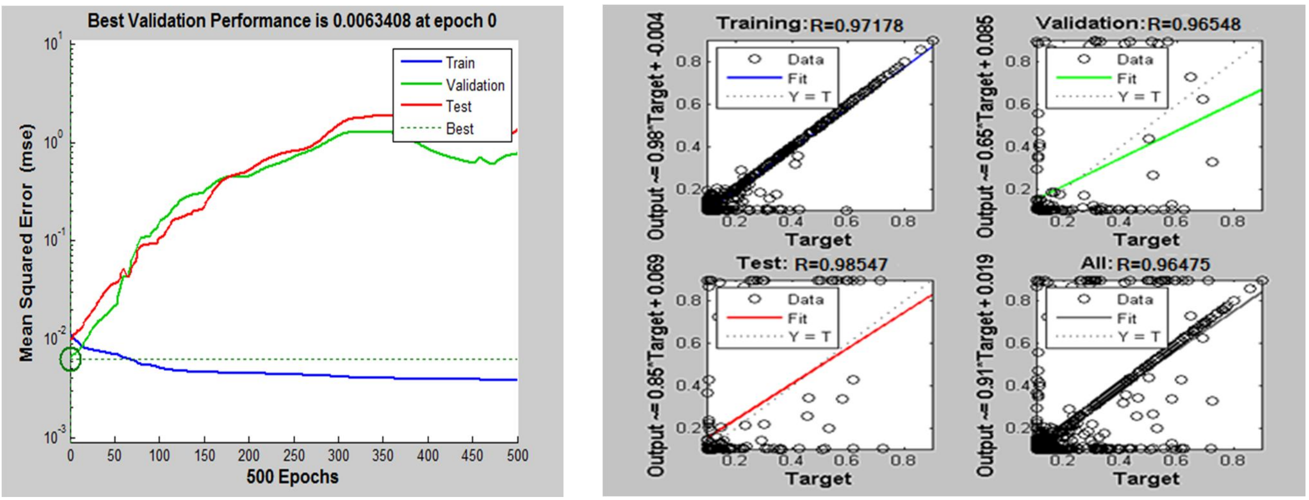
TABLE 4
ERROR OF EPOCH OF TESTING DATA FOR CFBPNN MODEL

	Epoch				Epoch				Epoch				Epoch		
Real	500	1000	1500	Real	500	1000	1500	Real	500	1000	1500	Real	500	1000	1500
0.125	0.016	0.000	0.008	0.102	-0.004	-0.013	-0.003	0.717	0.261	0.263	0.284	0.148	-0.174	-0.131	-0.152
0.183	0.017	0.015	0.014	0.113	-0.012	-0.008	-0.010	0.675	0.155	0.340	0.393	0.106	-0.019	-0.011	0.006
0.134	-0.121	-0.159	-0.167	0.310	0.154	0.156	0.152	0.269	-0.045	0.084	-0.047	0.101	-0.013	-0.014	-0.010
0.106	-0.008	-0.008	0.004	0.456	-0.110	-0.106	-0.106	0.128	-0.139	-0.128	-0.147	0.101	-0.037	-0.034	-0.039
0.100	-0.020	-0.017	-0.015	0.373	-0.062	-0.164	-0.077	0.100	-0.028	-0.009	0.013	0.167	0.025	0.028	0.022
0.335	0.197	0.200	0.195	0.193	0.015	0.011	-0.013	0.100	-0.020	-0.017	-0.018	0.102	-0.183	-0.175	-0.183
0.479	-0.004	-0.001	-0.016	0.116	-0.003	0.002	0.011	0.127	-0.015	-0.012	-0.018	0.101	-0.009	-0.021	-0.015
0.581	0.083	0.042	0.146	0.101	-0.005	-0.021	-0.013	0.110	-0.065	-0.066	-0.069	0.100	-0.041	-0.038	-0.043
0.342	-0.050	-0.017	0.012	0.146	0.017	0.020	0.017	0.153	0.025	0.027	0.029	0.235	0.093	0.096	0.090
0.109	-0.122	-0.121	-0.133	0.142	-0.069	-0.070	-0.074	0.104	-0.107	-0.114	-0.116	0.512	-0.031	-0.036	-0.038
0.116	0.008	0.017	-0.011	0.150	0.005	0.011	0.018	0.102	-0.010	-0.014	-0.009	0.515	-0.071	-0.014	0.011
0.153	0.002	0.002	0.001	0.109	-0.050	-0.048	-0.042	0.191	0.049	0.052	0.047	0.653	0.410	0.303	0.295
0.153	-0.045	-0.054	-0.055	0.100	-0.015	-0.016	-0.011	0.506	0.121	0.122	0.110	0.123	-0.275	-0.266	-0.222
0.151	-0.001	0.007	0.014	0.109	-0.025	-0.022	-0.026	0.499	-0.090	-0.059	-0.046	0.101	0.159	-0.003	-0.175
0.103	-0.046	-0.037	-0.029	0.151	-0.002	0.000	-0.004	0.244	-0.041	-0.103	-0.111	0.103	-0.020	-0.017	-0.018
0.100	-0.012	-0.015	-0.011	0.379	0.172	0.166	0.164	0.106	0.009	-0.034	-0.020	0.122	-0.023	-0.020	-0.026
0.113	-0.028	-0.025	-0.030	0.445	-0.034	-0.039	-0.042	0.102	-0.015	-0.003	0.001	0.100	-0.064	-0.065	-0.067
0.211	0.055	0.057	0.052	0.453	0.061	0.071	0.063	0.114	-0.025	-0.022	-0.027	0.100	-0.024	-0.020	-0.022
0.374	-0.045	-0.082	-0.093	0.191	-0.085	-0.108	-0.137	0.144	-0.012	-0.011	-0.014	0.102	-0.040	-0.037	-0.043
0.354	-0.248	-0.202	-0.252	0.119	0.016	-0.005	-0.002	0.118	-0.065	-0.072	-0.072	0.104	-0.040	-0.037	-0.043
0.172	-0.086	-0.106	-0.162	0.104	-0.003	-0.017	-0.007	0.104	-0.019	-0.017	-0.011	0.333	0.188	0.191	0.186
0.107	-0.012	0.005	0.003	0.108	-0.021	-0.018	-0.020	0.102	-0.028	-0.025	-0.027	0.328	-0.164	-0.159	-0.174
0.169	0.059	0.045	0.053	0.120	-0.027	-0.025	-0.029	0.101	-0.040	-0.037	-0.042	0.514	0.159	0.193	0.179
0.102	-0.170	-0.173	-0.180	0.105	-0.051	-0.052	-0.053	0.175	0.034	0.037	0.031	0.235	-0.356	-0.266	-0.155
0.111	0.001	-0.013	-0.007	0.106	-0.023	-0.021	-0.022	0.451	0.128	0.138	0.128	0.128	0.026	-0.008	0.007
0.104	-0.048	-0.046	-0.050	0.105	-0.039	-0.037	-0.041	0.515	-0.042	0.025	0.010	0.105	-0.010	-0.001	0.014
0.104	-0.032	-0.030	-0.033	0.323	0.182	0.184	0.181	0.245	-0.005	-0.122	-0.131	0.100	-0.023	-0.020	-0.019
0.101	-0.042	-0.040	-0.044	0.508	-0.011	-0.011	-0.020	0.153	0.047	0.015	0.026	0.139	0.001	0.004	-0.001

0.101	-0.037	-0.035	-0.040	0.479	-0.108	-0.025	0.080	0.100	-0.016	-0.006	0.023	0.112	-0.086	-0.087	-0.090
0.381	0.237	0.240	0.234	0.221	-0.105	-0.116	-0.119	0.100	-0.012	-0.016	-0.013	0.131	0.009	0.012	0.016
0.517	0.064	0.089	0.092	0.122	0.044	-0.016	0.003	0.104	-0.038	-0.035	-0.041	0.104	-0.061	-0.065	-0.065
0.680	0.165	0.202	0.266	0.100	-0.007	-0.010	-0.001	0.100	-0.048	-0.045	-0.050	0.147	0.026	0.029	0.030
0.296	-0.127	-0.127	-0.071	0.102	-0.021	-0.017	-0.019	0.101	-0.037	-0.035	-0.040	0.284	0.076	0.073	0.070
0.103	-0.206	-0.171	-0.188	0.100	-0.045	-0.042	-0.048	0.108	-0.035	-0.032	-0.038	0.770	0.244	0.246	0.240
0.100	-0.007	-0.023	-0.003	0.100	-0.041	-0.038	-0.043	0.104	-0.047	-0.045	-0.050	0.388	-0.030	-0.081	-0.044
0.137	-0.003	0.000	-0.006	0.121	-0.021	-0.018	-0.024	0.207	0.069	0.071	0.067	0.323	0.154	0.065	0.106
0.115	-0.077	-0.079	-0.082	0.100	-0.066	-0.066	-0.070	0.604	0.160	0.149	0.138	0.131	-0.090	-0.081	-0.128
0.101	-0.024	-0.021	-0.017	0.104	-0.021	-0.017	-0.019	0.503	0.026	-0.013	-0.014	0.100	-0.020	-0.014	-0.005
0.127	-0.002	0.000	-0.002	0.314	0.167	0.170	0.164	0.322	0.035	0.063	0.077	0.105	-0.014	-0.010	-0.010
0.102	-0.071	-0.073	-0.075	0.322	-0.225	-0.227	-0.229	0.169	0.055	0.050	0.015	0.100	-0.048	-0.045	-0.051
0.117	-0.005	-0.002	-0.002	0.503	0.121	0.201	0.156	0.186	0.062	0.073	0.082	0.100	-0.038	-0.035	-0.040
0.271	0.112	0.112	0.109	0.241	-0.340	-0.270	-0.168	0.215	0.059	0.052	0.048	0.156	0.014	0.017	0.011
0.480	-0.093	-0.105	-0.102	0.102	0.004	-0.036	-0.022	0.133	-0.020	-0.080	-0.084	0.116	-0.132	-0.126	-0.132
0.403	-0.130	-0.118	-0.097	0.102	-0.014	-0.005	-0.005	0.106	0.000	-0.002	0.013	0.120	0.004	0.003	0.010
0.317	0.085	0.080	0.075	0.126	-0.016	-0.013	-0.019	0.127	0.006	0.009	0.010	0.163	0.015	0.014	0.015
0.104	-0.072	-0.065	-0.110	0.101	-0.071	-0.073	-0.075	0.106	-0.061	-0.064	-0.065	0.519	0.304	0.289	0.287
0.106	0.000	-0.012	-0.008	0.135	0.013	0.017	0.016	0.103	-0.021	-0.018	-0.018	0.468	-0.150	-0.120	-0.090
0.118	-0.027	-0.025	-0.029	0.102	-0.084	-0.086	-0.089	0.195	0.055	0.057	0.053	0.183	-0.148	-0.132	-0.128
0.196	0.041	0.041	0.039	0.102	-0.016	-0.013	-0.012	0.433	0.036	0.033	0.021	0.115	0.013	0.004	0.007
0.115	-0.221	-0.263	-0.275	0.136	-0.007	-0.004	-0.009	0.463	-0.092	-0.045	-0.078	0.114	0.005	-0.009	0.000
0.111	0.005	-0.011	-0.003	0.454	0.267	0.264	0.262	0.310	0.031	0.012	-0.018	0.103	-0.039	-0.038	-0.039
0.102	-0.037	-0.036	-0.037	0.476	-0.062	-0.044	-0.024	0.123	0.050	0.013	-0.013	0.108	-0.025	-0.023	-0.025
0.124	-0.010	-0.007	-0.011	0.421	0.159	0.089	0.069	0.137	0.019	0.017	0.036	0.130	-0.018	-0.016	-0.020
0.209	0.041	0.040	0.037	0.472	0.207	0.183	0.177	0.100	-0.062	-0.066	-0.063	0.110	-0.059	-0.062	-0.063
0.405	0.049	-0.015	-0.028	0.142	-0.174	-0.167	-0.196	0.102	-0.014	-0.012	-0.011	0.103	-0.023	-0.020	-0.018
0.757	0.187	0.208	0.157	0.100	-0.035	0.001	0.012	0.124	-0.021	-0.018	-0.024	0.107	-0.029	-0.027	-0.031
0.280	-0.483	-0.092	-0.048	0.108	-0.006	-0.005	-0.003	0.100	-0.068	-0.069	-0.072	0.184	0.037	0.039	0.035
0.130	-0.001	0.031	0.021	0.104	-0.048	-0.046	-0.050	0.100	-0.023	-0.019	-0.020	0.333	0.000	-0.008	-0.019
0.125	-0.003	0.010	0.036	0.109	-0.030	-0.027	-0.031	0.105	-0.037	-0.034	-0.040	0.282	-0.293	-0.250	-0.245
0.102	-0.039	-0.037	-0.033	0.118	-0.030	-0.028	-0.032	0.237	0.089	0.092	0.086	0.246	-0.072	0.003	-0.037
0.112	-0.011	-0.008	-0.009	0.104	-0.049	-0.050	-0.051	0.477	-0.056	-0.071	-0.073	0.103	-0.039	-0.062	-0.116
0.114	-0.040	-0.038	-0.042	0.103	-0.027	-0.024	-0.026	0.499	-0.029	-0.031	-0.039	0.126	0.008	0.020	0.023
0.100	-0.045	-0.045	-0.046	0.211	0.069	0.071	0.067	0.686	0.440	0.328	0.314	0.102	-0.067	-0.069	-0.071
0.109	-0.021	-0.018	-0.021	0.458	-0.003	-0.014	-0.023	0.189	-0.341	-0.272	-0.170	0.102	-0.021	-0.018	-0.018
0.112	-0.041	-0.038	-0.043	0.648	0.117	0.135	0.105	0.100	0.007	-0.038	-0.093	0.105	-0.038	-0.035	-0.041
0.210	0.064	0.065	0.063	0.176	-0.217	-0.296	-0.216	0.100	-0.008	-0.029	-0.024	0.110	-0.036	-0.033	-0.038
0.494	0.074	0.040	0.029	0.106	-0.052	-0.060	-0.060	0.100	-0.043	-0.039	-0.046	0.103	-0.046	-0.044	-0.048
0.459	-0.103	-0.071	-0.072	0.103	-0.006	-0.022	-0.014	0.100	-0.043	-0.039	-0.046	0.106	-0.030	-0.027	-0.031
0.545	0.233	0.207	0.206	0.102	-0.038	-0.036	-0.040	0.100	-0.042	-0.039	-0.045	0.445	0.299	0.301	0.297
0.173	-0.065	-0.212	-0.196	0.100	-0.042	-0.039	-0.044	0.100	-0.043	-0.039	-0.046	0.401	-0.086	-0.147	-0.107
0.100	-0.033	-0.035	-0.003	0.100	-0.040	-0.037	-0.043	0.108	-0.034	-0.031	-0.037	0.403	0.145	0.146	0.115
0.105	-0.005	-0.021	-0.015	0.101	-0.042	-0.039	-0.045	0.288	0.136	0.139	0.134	0.230	-0.082	-0.035	-0.095
0.160	0.013	0.015	0.010	0.100	-0.043	-0.040	-0.046	0.524	-0.074	-0.074	-0.066	0.117	-0.023	0.001	0.017
0.142	-0.105	-0.105	-0.111	0.108	-0.035	-0.031	-0.037	0.334	-0.225	-0.127	-0.054	0.103	-0.008	-0.004	0.003
0.100	-0.033	-0.023	-0.014	0.143	-0.008	-0.005	-0.010	0.225	0.073	0.084	0.049	0.100	-0.029	-0.027	-0.028
0.104	-0.010	-0.010	-0.008	0.431	0.240	0.234	0.233	0.102	-0.055	-0.034	-0.046	0.117	-0.023	-0.020	-0.026
0.102	-0.045	-0.042	-0.047	0.623	0.118	0.136	0.150	0.100	-0.010	-0.011	-0.014	0.101	-0.060	-0.059	-0.062
0.203	0.061	0.064	0.059	0.364	-0.086	-0.088	-0.054	0.100	-0.041	-0.038	-0.043	0.102	-0.026	-0.023	-0.026
0.327	-0.108	-0.113	-0.124	0.109	-0.163	-0.135	-0.166	0.103	-0.040	-0.037	-0.043	0.103	-0.041	-0.038	-0.043
0.762	0.202	0.251	0.246	0.100	-0.005	0.005	-0.034	0.101	-0.045	-0.041	-0.047	0.104	-0.040	-0.037	-0.042
0.228	-0.289	-0.201	-0.198	0.100	-0.034	-0.031	-0.036	0.103	-0.038	-0.035	-0.041	0.140	-0.004	-0.002	-0.007
0.117	-0.049	0.167	0.075	0.100	-0.043	-0.039	-0.046	0.126	-0.019	-0.016	-0.021	0.456	0.266	0.262	0.260
0.100	-0.010	-0.008	-0.002	0.114	-0.029	-0.026	-0.032	0.114	-0.056	-0.057	-0.060	0.315	-0.232	-0.204	-0.189
0.102	-0.026	-0.023	-0.025	0.118	-0.040	-0.039	-0.043	0.387	0.254	0.256	0.258	0.334	0.265	0.226	0.196
0.148	0.003	0.007	0.001	0.101	-0.046	-0.047	-0.047	0.626	0.173	0.187	0.197	0.177	-0.206	-0.163	-0.182
0.107	-0.109	-0.109	-0.113	0.106	-0.021	-0.018	-0.021	0.374	-0.085	-0.052	-0.074	0.189	0.066	0.082	0.083
0.100	-0.015	-0.015	-0.011	0.218	0.070	0.073	0.068	0.157	-0.104	-0.080	-0.114	0.121	-0.023	-0.027	-0.028
0.106	-0.030	-0.027	-0.032	0.384	-0.090	-0.112	-0.120	0.223	0.104	0.117	0.108	0.103	-0.004	-0.017	-0.006
0.104	-0.045	-0.043	-0.048	0.423	-0.178	-0.133	-0.189	0.101	-0.164	-0.260	-0.280	0.112	-0.015	-0.012	-0.013
0.206	0.066	0.069	0.064	0.567	0.220	0.234	0.199	0.100	-0.009	-0.012	-0.015	0.100	-0.052	-0.051	-0.054

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue VII, July 2019- Available at www.ijraset.com

0.547	0.104	0.093	0.082	0.124	-0.179	-0.296	-0.259	0.102	-0.039	-0.036	-0.042	0.101	-0.031	-0.028	-0.032
0.482	0.018	-0.044	-0.030	0.100	-0.098	-0.032	-0.042	0.133	-0.012	-0.008	-0.014	0.102	-0.041	-0.038	-0.044
0.242	-0.105	-0.048	-0.042	0.100	-0.022	-0.019	-0.020	0.101	-0.081	-0.083	-0.085	0.166	0.022	0.025	0.019
0.209	0.121	0.062	0.081	0.111	-0.032	-0.029	-0.035	0.100	-0.018	-0.015	-0.014	0.508	0.232	0.237	0.230
0.100	0.004	-0.032	-0.005	0.112	-0.042	-0.040	-0.045	0.106	-0.036	-0.033	-0.039	0.405	-0.212	-0.179	-0.158
0.100	-0.007	-0.021	-0.022	0.100	-0.044	-0.044	-0.045	0.205	0.056	0.059	0.054	0.339	0.119	0.117	0.130
0.113	-0.030	-0.026	-0.033	0.111	-0.020	-0.017	-0.021	0.195	-0.232	-0.248	-0.259	0.126	-0.073	-0.060	-0.115
0.211	0.054	0.055	0.051	0.101	-0.054	-0.052	-0.056	0.652	0.541	0.532	0.556	0.112	-0.006	0.013	-0.003
0.122	-0.294	-0.333	-0.344	0.397	0.264	0.267	0.263	0.316	-0.273	-0.247	-0.231	0.104	-0.027	-0.024	-0.023



(a) (b)

Figure 5: (a) Performance result of CFBPNN (b) Regression result of CFBBPNN

Table 5 is shown that the predicted results of monthly rainfall of East Madhya Pradesh, India from 2011-2017. It is revealed that heavy rainfall would be occur in July 2011, July 2012, July 2013, July 2014, August 2015, July 2016 and July 2017 respectively with 382.3, 377.9, 344, 297.7, 336, 608.7 and 352.9. The largest monthly rainfall forecasting result would be occur 0.783 in July 2016. Meanwhile, in 2011, 2012, 2013, 2014, 2015, 2016 and 2017, the average rainfall was high with 137.4, 104.7, 117.6, 86.6, 97.1, 135 and 81.7, respectively. Figure 6 plot the result of rainfall predicted values using 2-50-10-1 architecture with 500 epochs and 0.1 learning rate. These test outcomes are consistent with the nature of the monthly rainfall in the East Madhya Pradesh, India. Figure 7 are plots the prediction result with the actual rainfall data of 7 years from 2011-2017.

TABLE 5
MONTHLY RAINFALL PREDICTION VALUES FROM 2011-2017

Years/ Months	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	Min	Max	Mean
2011	4.5	8.9	21.4	18.7	45.5	303	382.3	350.3	336.9	138.1	3.6	36.5	3.6	382.3	137.4
2012	38.3	97.1	12.5	38.3	38.3	38.3	377.9	343.1	219.2	8.9	15.2	29.4	8.9	377.9	104.7
2013	36.5	37.4	104.3	16.9	33.9	29.4	344	328	264.7	107	95.4	14.3	14.3	344	117.6
2014	38.3	79.3	66	16	26.7	35.7	297.7	232.6	105.2	116.8	14.3	9.8	9.8	297.7	86.6
2015	49.9	70.4	17.8	2.7	9.8	110.5	331.5	336	192.5	15.2	16.9	12.5	2.7	336	97.1
2016	37.4	60.6	22.3	55.3	22.3	57.9	608.7	309.3	252.2	155.1	23.2	16	16	608.7	135
2017	38.3	42.8	40.1	33	35.7	50.8	352.9	240.6	8	93.6	23.2	22.3	8	352.9	81.7

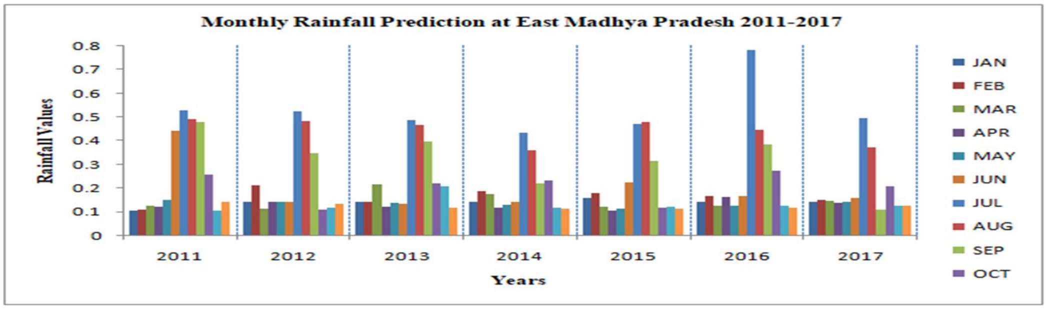
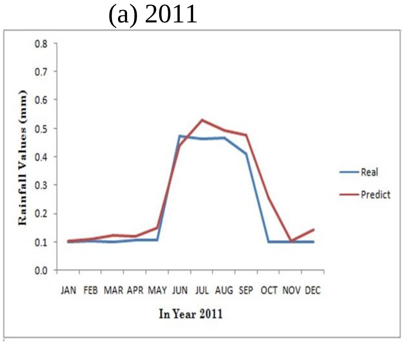
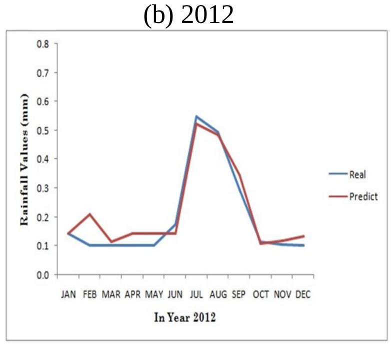


Figure 6: Plot the predicted results with [2-50-10-1] of CFBPNN, epochs 500

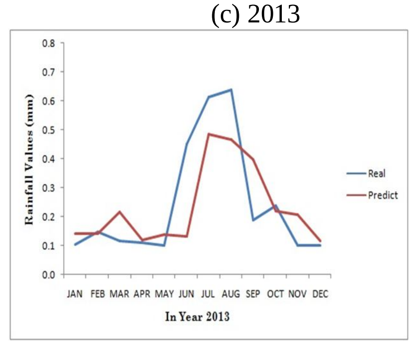
(a) 2011



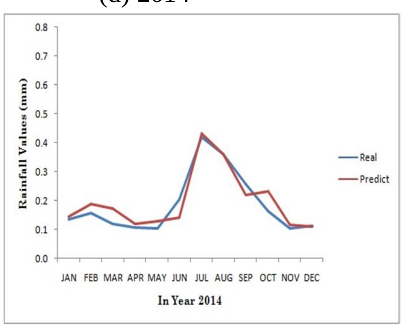
(b) 2012



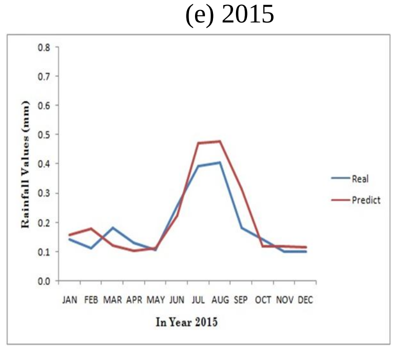
(c) 2013



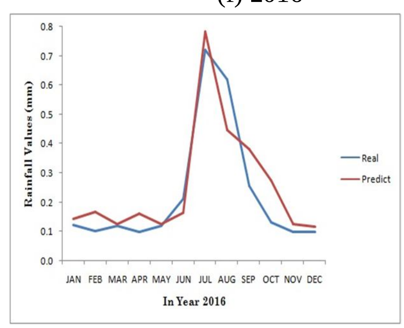
(d) 2014



(e) 2015



(f) 2016



(g) 2017

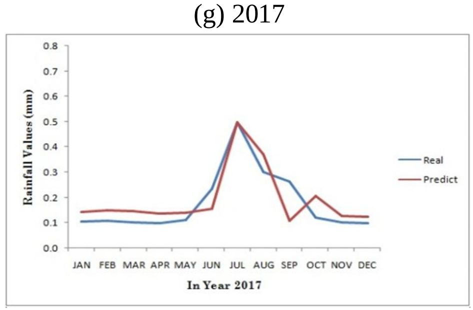


Figure 7: Plot the predicted results of each year's 2011-2017 (a-g) at East Madhya Pradesh

VII. CONCLUSIONS

In this paper, 117 years rainfall records were analyzed, using data of East Madhya Pradesh, India from Indian Meteorological Department. A CFBPNN technique has been used to developed the network and predict & analysis of monthly rainfall at East Madhya Pradesh, India. After training and testing phase, the three models are developed with distinct epochs i.e., 500, 1000 and 1500, after that the best MSE value was 0.0063408 which is evaluated using 2-50-10-1 architecture and 500 epochs with 0.1 learning rate. With the help of architecture 2-50-10-1, we can find the high accurate result. In comparison with existing methods, this proposed method had been more accurate results. These results predicting the monthly rainfall by this proposed system algorithm. In the result of this experiment, this paper has describes that CFBPNN models can be used as a predictive algorithm that provides a best predictive accuracy. The performance of the predicted outcomes have been demonstrated the suitability of the region of East Madhya Pradesh, India. In future, authors intent to work on several things. So in future we will work on mechanism which would be able to differentiate them. For this purpose, we will include a comparison of few ANN techniques and the optimization method to achieve more accurate prediction results for the monthly forecast of rainfall.

REFERENCES

[1] Available: https://www.nationalgeographic.org/encyclopedia/tropics/
[2] M. Kannan, S. Prabhakaran, P. Ramachandran “Rainfall Forecasting Using Data Mining Technique”, International Journal of Engineering and Technology Vol.2 (6), 2010, pp. 397-401.
[3] Valmik B Nikam and B. B. Meshram, “Modeling Rainfall Prediction using Data Mining Method”, Fifth International Conference on Computational Intelligence, Modeling and Simulation, Issue No: 2166-8531, PP:132-136, 2013.
[4] Mishra, Neelam & Soni, Hemant & Sharma, Sanjiv & K Upadhyay, A., “Development and Analysis of Artificial Neural Network Models for Rainfall Prediction by Using Time-Series Data”, International Journal of Intelligent Systems and Applications, pg. no. 16-23, 2018.
[5] Danish Khan “CAKE – Classification, association and discovery of knowledge an approach to distributed data mining (DDM) using parallel data mining agents (PADMA)”, IEEE / WIC / ACM international conference on web intelligence and intelligent agent technology, 2008.
[6] Yan Chen1, Ming Yang2, Lin Zhang3, “General Data Mining Model System Based on Sample Data Division”, Second International Symposium on Knowledge Acquisition and Modeling, 2009.
[7] Ganesh P. Gaikwad and . V. B. Nikam, “Different Rainfall Prediction Models And General Data Mining Rainfall Prediction Model”, International Journal of Engineering Research & Technology, 2013.
[8] V. B. Nikam and B. B. Meshram, "Modeling Rainfall Prediction Using Data Mining Method: A Bayesian Approach", Fifth International Conference on Computational Intelligence, Modelling and Simulation, Seoul, 2013, pp. 132-136, 2013.
[9] Nair, Archana & Singh, Gurjeet & Mohanty, U C. (2017). Prediction of Monthly Summer Monsoon Rainfall Using Global Climate Models Through Artificial Neural Network Technique. Pure and Applied Geophysics. 175. 10.1007/s00024-017-1652-5.
[10] Seyed, A., Shamsnia, M., Naeem, S. & Ali, L., Modelling weather parameter using stochastic methods (ARIMA Model)(Case Study:Abadeh station, Iran), 2011 International conference on Environment and Industrial innovation IPCBEE, 12, 2011.
[11] Mahsin, M.D., Yesmin, A. & Monira, B., Modeling Rain in Dacca (National Capital) Division of Bangladesh Using Time Series Analysis, Journal of Mathematical Modelling and Application, **1**(5), pp. 67-73, 2012.
[12] Abraham, A., Steinberg, D. & Philip, N.S., Rainfall Forecasting Using Soft Computing Models and Multivariate Adaptive Regression. Splines, IEEE SMC Transactions, Special Issue on Fusion of Soft Computing and Hard Computing in Industrial Applications, 1, pp. 1-6, 2001..
[13] N. Prasad, P. Kumar , M. M. Naidu, “An Approach to Prediction of Precipitation Using Gini Index in SLIQ Decision Tree”, 4th International Conference on Intelligent Systems, Modeling and Simulation, pp. 56-60, 2013.
[14] R. Adhikari, R. K. Agrawal, “Forecasting strong seasonal time series with Artificial Neural Network”, Journal of Scientific and Industrial Research, vol. 71, pp. 657-666, 2012.
[15] S. Singh, J. Gill, “Temporal Weather Prediction using Back Propagation based Genetic Algorithm Technique”, I.J. Intelligent Systems and Applications, vol.6 (12), pp. 55-61,2014.
[16] B. M. Al-Maqaleh, A. A. Al-Mansoub and F. N. Al-Badani, “Forecasting using Artificial Neural Network and Statistics Models”, I.J. Education and Management Engineering, vol.3, pp. 20-32, 2016.
[17] V. Nourani, T. R. Khanghah, and A. H. Baghanam, “Application of Entropy Concept for Input Selection of Wavelet-ANN Based Rainfall-Runoff Modeling”, Journal of Environmental Informatics, vol. 26 (1), pp.52-70, Sep. 2015.
[18] R. C. Deo, M. Sahin, “Application of the Artificial Neural Network model for prediction of monthly Standardized Precipitation and Evapotranspiration Index using hydrometeorological parameters and climate indices in eastern Australia”, Atmospheric Research, vol. 161, pp. 65-81, 2015.
[19] M. Valipour, “Optimization of neural networks for precipitation analysis in a humid region to detect drought and wet year alarms”, Meteorological Applications, vol. 23 (1), pp. 91-100, 2016.
[20] S. K. Nanda, D. P. Tripathy, S. K. Nayak, S. Mohapatra. “Prediction of Rainfall in India Using Artificial Intelligent”, Systems and Applications, vol.12, pp.1-22, 2013.
[21] N. Sethi, K. Garg. “Exploiting Data Mining Technique for Rainfall Prediction”, International Journal of Computer Science and Information Technologies. Vol. 5 (3), pp.3982-3984, 2014.
[22] Jakeman, A. J., Letcher, R. A., Norton, J. P., “Ten iterative steps in development and evaluation of environmental models”, Environmental Modelling & Software 21(5), 602-614, 2006.
[23] Haviluddin and R. Alfred, “Daily Network Traffic Prediction Based on Backpropagation Neural Network”, Australian Journal of Basic and Applied Sciences, pg. no. 164-169, 2014.
[24] Subramanian, Renuga devi & Pachiyappan, Arulmozhivarman & Venkatesh, C & Agarwal, Pranay, “Performance comparison of artificial neural network models for daily rainfall prediction”, International Journal of Automation and Computing, 2016.
[25] Meteorological data source for East Madhya Pradesh, Available: https://data.gov.in/catalog/rainfall-india.
[26] Singh, Sapna, “Backpropagation Learning Algorithm Based on Levenberg Marquardt Algorithm”, Computer Science & Information Technology, Volume 2, pp. 393-398, 2012.
[27] Warsito, Budi & Santoso, Rukun & Suparti, Suparti & Yasin, Hasbi, “Cascade Forward Neural Network for Time Series Prediction”, Journal of Physics: Conference Series. 1025. 012097, 2018.