



International Technology Management Review
Volume 2 Number 1 (October) 2009
<<http://www.academicglobalpublications.com/itmr/>>

Forecasting Bangladeshi monsoon rainfall using neural network and genetic algorithm approaches

Dr. Shipra Banik

Assistant Professor,
School of Engineering and Computer Science,
Independent University, Bangladesh, Dhaka, Bangladesh.
Email: banik@secs.iub.edu.bd

Dr. Mohammed Anwer

Professor,
School of Engineering and Computer Science,
Independent University,
Bangladesh, Dhaka, Bangladesh.
Email: manwer@secs.iub.edu.bd

A.F.M. Khodadad Khan

Professor,
School of Engineering and Computer Science,
Independent University, Bangladesh, Dhaka, Bangladesh.
Email: khoda@secs.iub.edu.bd

Rifat Ara Rouf,

Senior Lecturer,
School of Engineering and Computer Science,
Independent University, Bangladesh, Dhaka, Bangladesh.
Email: rifatar@secs.iub.edu.bd

Farah Habib Chanchary,

Senior Lecturer,
School of Engineering and Computer Science,
Independent University, Bangladesh, Dhaka, Bangladesh.
Email: farah@secs.iub.edu.bd

Abstract

Title: Forecasting Bangladeshi monsoon rainfall using neural network and genetic algorithm approaches

Keywords: Atmospheric dynamics, forecasting, artificial intelligence, neural network, fuzzy inference system, genetic algorithm.

Category of paper: Research paper

Purpose of the research/paper: True information about rainfall is crucial for human activities such as the use and the management of water resources, hydroelectric power projects, warning to impend droughts or floods, urban areas sewer systems and many others. This paper investigates the development of an efficient model to forecast monthly monsoon rainfall for a number of stations, namely Barishal, Chittagong, Dhaka, Khulna, Rajshahi and Sylhet. It is believed that rainfall forecasting is difficult and also a challenging task for anyone because rainfall data are multi-dimensional and nonlinear. Therefore, to model rainfall data, the AI forecasting models, namely artificial neural network (ANN), adaptive neuro fuzzy inference system (ANFIS) and genetic algorithm (GA) have been used. Results found by the AI models are also compared to the linear regression model to show advantages of selecting these models.

Methodology: Quantitative.

Findings: Results suggest that the ANFIS model and the GA model can be used to forecast monthly monsoon rainfall more accurately than the ANN model and the statistical model.

Value of the paper: The findings will be of interest to individuals e.g. applied researchers, academics, planners, policy makers and practitioners for formulating wise policies related to rainfall in the context of the country, Bangladesh.

Number of pages: 18

Number of tables/figures: 8

Section headings: **Abstract, Introduction, Model and data set, Techniques used to predict rainfall, Design of considered techniques, Results and discussion, Conclusion**

© 2009 Academic Global Publications P/L. This work is copyright. You may download and print only one paper copy from this electronic file for your personal use only, from which you may not make any further paper copies.



International Technology Management Review
Volume 2 Number 1 (October) 2009
<<http://www.academicglobalpublications.com/itmr/>>

Forecasting Bangladeshi monsoon rainfall using neural network and genetic algorithm approaches

Dr. Shipra Banik, Dr. Mohammed Anwer, A.F.M. Khodadad Khan, Rifat Ara Rouf, and Farah Habib Chanchary

Introduction

The important hydrological event rainfall is the quantity of water falling in drops from vapor condensed in the atmosphere. When water droplets in clouds become too heavy to stay in the air, they fall out towards the ground. This paper proposes an appropriate model to predict monsoon rainfall for different stations of Bangladesh. Making reliable prediction about rainfall is very important in many areas of human activities, indicated as follows: Rainfall mainly controls our water supplies, which are the basis for crops production. It is especially important for rainfed (or dry) land agriculture. For example, crops plant use a huge amount of water and rains, to a major extent, provide this water through the soil for the development and growth of plants. Since Bangladesh is an agro-based country, accurate prediction of rainfall will be useful for proper plannings of cultivation. That's why those who are involved in agriculture, they will be interested to know whether the next days (or months) will be rainy/non-rainy. Although water is vital to life, yet water can be extremely destructive. Thus, rainfall forecasting can warn of happening flood or drought so that peoples can save their lives and properties. Rainfall forecasting is also important for engineering applications, mainly for the design of hydroelectric power projects, because this system requires prior information about average rainfall, maximum/minimum rainfall, maximum intensity, duration etc. for a year/each month. In urban areas, rainfall also has a strong influence on traffic control, operations of sewer systems and others. Thus, we believe that precise rainfall prediction is important for practitioners who are interested to make wise policies related to this event.

Literature review

True quantitative rainfall forecasting is generally difficult and also a challenging task for anyone because of our complex atmospheric processes. Thus, rainfall is treated as one of the most complex and difficult events among other hydrological events. Forecasting techniques, namely the statistical methods (ARIMA, regression model, hidden Markov model, exponential smoothing etc.) and the AI methods (neural networks (artificial neural network (ANN), adaptive network based fuzzy inference system (ANFIS) etc.), fuzzy inference system (FIS), genetic algorithm (GA) and others) are proposed in literature. Note that statistical techniques have been developed for many years but have been proven

International Technology Management Review, October 2009, v2, n1.

to be incapable to handle the non-linear series (Kamruzzaman & Sarker 2003; Kihoro et al. 2000). Hence, the AI methods have been proposed and engaged successfully in series forecasting, where only goal is to minimize predictive error for the considered model. Thus, these methods have been applied in solving numerous forecasting problems (e.g. Jang 1993; Charhate & Deo 2007; Banik et al. 2007; Kumer & Debroy 2007 and many others). We believe rainfall data are multi-dimensional, non-linear and dynamic, therefore to search for an appropriate model, the powerful techniques ANN, ANFIS and GA have been chosen. Results obtained by these models are also compared to the linear multiple regression (MR) model to show selected AI models performances.

Empirical works (e.g. Whigham & Crapper 2001; Navone & Ceccatto 1994; Sen & Oztopal 2001; Lee et al. 2006; Mondol & Shahid 2004; Islam et al. 2002; Wahid & Islam 1999 and many others) have been carried out to forecast rainfall in context of various countries. Although studies have been conducted to predict rainfall for Bangladesh (e.g. Mondol & Shahid 2004; Islam et al. 2002; Wahid & Islam 1999 and others), to our knowledge, no comparative work is available to predict rainfall under the AI models for various Bangladeshi stations. We have considered this issue in this paper, which is planned as follows: Rainfall model and data sets description are explained in section 2. Considered techniques to predict rainfall are reviewed in section 3. Computational settings of the selected techniques are given in section 4. The relative performances of the considered methods are discussed in section 4. We conclude the paper with some future research plans in final section.

Model and data set

Based on available information, this section describes the rainfall model and also describes the data set used to conduct this research.

Rainfall model

The following functional relation is considered to predict Bangladeshi monsoon rainfall:

$$R_t = f(\text{Intercept}, T_t, RH_t, CL_t), \quad t = 1, 2, \dots, n \quad (1)$$

where R_t = rainfall (mm), T_t = temperature ($^{\circ}\text{C}$), RH_t = relative humidity (%) and CL_t = cloud coverage (octas). Our target is to estimate the model (1) parameters using the techniques of ANN, ANFIS, GA and MR.

Data set

For different stations of Bangladesh (see Table 1), monthly R_t , T_t , RH_t and CL_t for monsoon periods (April to October) are used, which are extracted from <http://www.barc.gov.bd/database>.

To understand behaviors of the considered variables, summary statistics are reported in Table 1. It can be seen that the highest mean rainfall is observed for the Sylhet region (547.55 mm) and the least mean amount for the Rajshahi region (210.16 mm). Standard deviation measures, e.g., for the Sylhet region, show that all times rainfall was not observed 547.55 mm (expected range: 256.13 to 838.97). Skewed measures indicate that rainfall patterns do not follow the normal distribution. For example, skewness 0.38 mm for the Sylhet region indicates that most of months rainfall amounts are observed below 547.55 mm.

From the Table 1, it is also observed that mean of T_t and mean of RH_t are almost same for all regions. But records of CL_t show that like rainfall, Sylhet region experiences the highest mean amount (5.88 octas), can be a cause of occurring highest rain. Skewed measures (like R_t) show that T_t , RH_t and CL_t variables do not

Table 1: Numerical summary statistics for R_t, T_t, RH_t and CL_t

<i>Barishal Station (Time periods: Jan-1953:Dec-2002)</i>				
Statistical Measures→	Mean	SD	Skewness	Correlation Coefficient
R_t	285.43	176.38	0.79	-
T_t	28.42	0.72	0.09	-0.27
RH_t	85.06	4.47	-0.89	0.61
CL_t	5.22	1.32	-0.50	0.64
<i>Chittagong Station (Time periods: Jan-1949:Dec-2002)</i>				
Statistical Measures→	Mean	SD	Skewness	Correlation Coefficient
R_t	387.81	285.10	1.06	-
T_t	28.04	0.67	-0.12	-0.18
RH_t	82.70	3.70	-0.28	0.67
CL_t	5.22	1.33	-0.52	0.70
<i>Dhaka Station (Time periods: Jan-1953:Dec-2002)</i>				
Statistical Measures→	Mean	SD	Skewness	Correlation Coefficient
R_t	274.24	154.35	0.73	-
T_t	28.56	0.86	-0.17	-0.15
RH_t	80.78	5.58	-1.07	0.59
CL_t	5.21	1.26	-0.57	0.60
<i>Khulna Station (Time periods: Jan-1953:Dec-2002)</i>				
Statistical Measures→	Mean	SD	Skewness	Correlation Coefficient
R_t	234.09	150.81	0.91	-
T_t	29.09	0.89	-1.65	-0.17
RH_t	83.11	5.37	-1.06	0.57
CL_t	4.61	1.41	-0.37	0.66
<i>Rajshahi Station (Time periods: Jan-1964:Dec-2002)</i>				
Statistical Measures→	Mean	SD	Skewness	Correlation Coefficient
R_t	210.16	147.09	0.82	-
T_t	28.96	0.99	-0.37	-0.04
RH_t	79.22	9.61	-1.40	0.64
CL_t	4.54	1.52	-0.42	0.71
<i>Sylhet Station (Time periods: Jan-1953:Dec-2002)</i>				
Statistical Measures→	Mean	SD	Skewness	Correlation Coefficient
R_t	547.55	291.42	0.38	-
T_t	27.27	1.10	-0.60	0.10
RH_t	83.68	4.81	-1.11	0.62
CL_t	5.88	1.24	-0.76	0.72

Source: Developed for this research

follow normal distribution. Correlation coefficient (CC) results indicate that there is a negative relationship exists between T_t and R_t and a positive relationship between RH_t and CL_t . We believe these findings will be useful for individuals who are interested about the events R_t, T_t, RH_t and CL_t .

Techniques used to predict rainfall

Artificial neural network

Culloch & Pitts (1943) introduced this system. It consists the following processing functions:

- Receiving inputs
- Assigning appropriate weight coefficients of inputs
- Calculating weighted sum of inputs
- Comparing this sum with some threshold and finally
- Determining an appropriate output value.

Fig-1 presents a basic structure of ANN, which has 1 input layer, two hidden layers (with sufficient no. of neurons) and 1 output layer. Each neuron receives an input $P_{Q \times 1}$ (T_t, RH_t and CL_t) and produces the net output as

$$\hat{R}_t = WP + b$$

where $W_{N \times Q}$ are the weights, $b_{N \times 1}$ is the bias, N are the number of neurons in the hidden layers and Q are the number of elements in P . Passing net input through an activation function produces the output of the neurons. Usually, the sigmoid function $[y=f(x)=1/(1+e^{-x})]$ is used as the activation function. The properties of this function are to mimic the nerve cell, which either fires or does not fire. Other used activation functions are hard limiter, pureline, transig, logsigmoid and others.

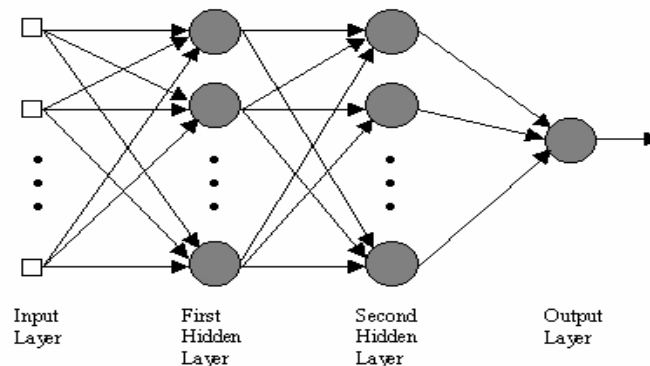


Fig-1: An ANN system

Source: Developed for this research

Networks are trained so that a particular input leads to a specific target output. The training algorithm is the standard BP, which uses the GD technique to minimize error. During training, each desired \hat{R}_t is compared with the actual R_t and calculates error at the output layer. The backward pass is the error BP and adjustments of weights. Thus, the network is adjusted based on a comparison of the output and

the target until the network output matches the target. When the training process is completed, then the network with adjusted estimated parameters ($W_{N \times Q}$ and $b_{N \times 1}$) is used to test a set of data, which is different from the training set of data.

Adaptive network based fuzzy inference system

This architecture is proposed by Jang (1993) and is developed based on the theory of fuzzy set and fuzzy logic. It is a combination of two intelligence systems, namely ANN system and FIS system in such a way that the ANN learning algorithm is used to determine the parameters of the FIS. ANN is a non-linear statistical data-modeling tool, which can capture and model any input-output relationship (or can learn to detect complex patterns in data). FIS (involves membership function (mf), fuzzy logic operator and if-then-rules) is the process of formulating the mapping from a given input to an output using fuzzy logic. This mapping provides a basis from which decisions can be made or patterns can be discerned. A typical ANFIS architecture is given in Fig-2, which shows that it has 5 layers (1 input layer, 3 hidden layers that represents mf and fuzzy rules and 1 output layer) and it uses Sugeno-fuzzy inference model to be the learning algorithm. As an example, fuzzy if-then-rules for first-order Sugeno-fuzzy model can be expressed as:

Rule 1: If x (input 1) is A_1 and y (input 2) is B_1 , then f_1 (output) = $p_1x+q_1y+r_1$

Rule 2: If x (input 1) is A_2 and y (input 2) is B_2 , then f_2 (output) = $p_2x+q_2y+r_2$

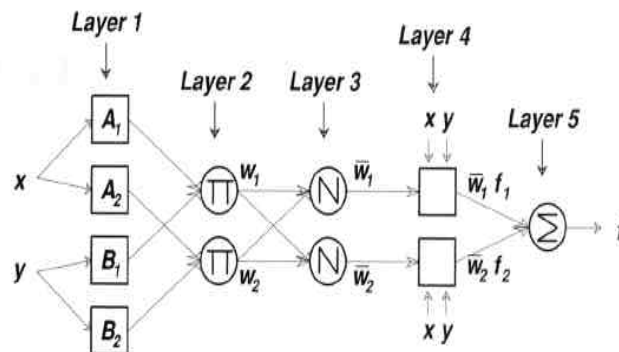


Fig-2: AN ANFIS system

Source: Developed for this research

In the above figure, the circular nodes represent fixed nodes whereas the square nodes are nodes that have parameters to be learnt. Each layer in this figure is associated with a particular step in the FIS. The following concepts are the process, where the input vector is fed through the ANFIS network layer by layer:

Layer 1: Fuzzy layer (generates mf grades): Input x to A_1 and A_2 and input y to B_1 and B_2 respectively, where A_1, A_2, B_1 and B_2 [fuzzy sets] are the linguistic expressions, which are used to distinguish the mf represented by the premise parameters (PP). The relationship between the input-output mf:

$$O_{A_i}^1 = \mu_{A_i}(x); O_{B_i}^1 = \mu_{B_i}(y), i=1,2$$

where $\mu_{A_i}(x)$ and $\mu_{B_i}(y)$ denote mf. Any continuous and piecewise differentiable functions such as generalized bell-shape mf, Gaussain-shaped mf, triangular-shaped mf, trapezoidal-shaped mf can be used in this layer.

Layer 2: Production layer (generate the firing strengths): It is marked as Π and output is defined as

$$O_i^2 = W_i = \mu_{A_i}(x) \times \mu_{B_i}(y)$$

where W_i , $i = 1, 2$ are weight functions of the next layer. Here the t-norm “product” is used as fuzzy operator.

Layer 3: Normalized layer (normalize the firing strength): It is marked as N and is used to normalize W_i (i.e. calculates the ratio of firing strength of the rules with the total firing strengths). It is defined as:

$$O_i^3 = \bar{W}_i = W_i / \sum_{i=1}^2 W_i, i = 1, 2.$$

Layer 4: Defuzzy layer (calculates output rules): Here an adaptive node \bar{W}_i are outputs and $\{p_i, q_i, r_i\}$ are the parameter sets [known as consequent parameters (CP)] in this layer. The relationship between input and output is:

$$O_i^4 = \bar{W}_i f_i = \bar{W}_i (p_i x + q_i y + r_i), i = 1, 2.$$

Layer 5: Total output layer (sum all inputs from the layer 4): Its node is marked as Σ , computes the overall output. It can be expressed as:

$$O_i^5 = \text{Output} = \sum_{i=1}^2 \bar{W}_i f_i = \bar{W}_i (p_i x + q_i y + r_i)$$

Next step is to familiar how the ANFIS learns PP and CP for mf and rules. From the Fig-2, it is clear that the final layer output can be expressed as a linear combination of CP. In symbols, we get

$$f = (\bar{W}_1 x) p_1 + (\bar{W}_1 y) q_1 + (\bar{W}_1) r_1 + (\bar{W}_2 x) p_2 + (\bar{W}_2 y) q_2 + (\bar{W}_2) r_2.$$

Thus, we have a set of total ANFIS parameters S , which is calculated as $S = S1 \cup S2$, where $S1$ = set of PP (non-linear) and $S2$ = set of CP (linear). The learning algorithm of ANFIS is a hybrid algorithm, which combines the GD method and the least square estimation (LSE) for an effective search of PP and CP. That means ANFIS uses a two pass of learning algorithm to reduce the error: (i) Forward pass (FP) and (ii) backward pass (BP). The hidden layer is computed by the GD method of the feedback structure and the final output is estimated by the LSE. Activities in each pass can be summarized in the following way:

	FP	BP
S1	Fixed	GD
S2	LSE	Fixed
Signals	Node outputs	Errors signals

The process is as follows:

FP: Present the input vector. Calculate the node outputs layer by layer (go forward until the layer 4). Identify S2 parameters using the LSE (here S1 is fixed) and compute error measure for each training pairs.

BP: Error signals propagate backward and the PP is updated by the GD method (details, Jang (1993) see pp.4-5).

The output equation from the layer 5 can be rearranged as more usable form:

$$Y = XW$$

where $X = [\bar{W}_1x, \bar{W}_1y, \bar{W}_1, \bar{W}_2x, \bar{W}_2y, \bar{W}_2]$ and $W = [p_1, q_1, r_1, p_2, q_2, r_2]^T$. When the input-output training pattern exists, the vector W can be solved using the ANFIS learning algorithm.

Genetic algorithm

This technique is proposed by Holland (1975). Later Koza (1992) used this approach (called his method genetic programming (GP)) to evolve programs to perform certain tasks (details, see Koza (1992). By definition, GA is a technique based on the 'Darwin's Principle of Natural Selection' and is used to solve optimisation problems. The basic idea is to select the best, discard the rest and differs from other systems in the following way. GA does not require any assumptions such as linearity, stationarity, homogeneity, chaotic or others. Thus, to handle the complex multi-dimensional behaviours of a system, this approach has been used efficiently in literature (e.g. Lee et al. 2006; Holland 1975; Koza 1992; Wei et al. 2004 and others). Fig-3 flowchart shows the basic steps involved in a GA process (see Table 2, for the standard GA options). Each steps brief explanation of Fig-3 and Table 2 are as follows:

Step-1: Create an initial population consisting of random chromosomes. To understand the GA process, e.g., for (1), consider the following random population of 6 chromosomes with 4 parameters each:

(0.13,0.01,0.84,0.68), (0.20,0.74,0.52,0.37), (0.19,0.44,0.20,0.83), (0.60,0.93,0.67,0.50),
(0.27,0.46,0.83,0.70), (00.19,0.41,0.01,0.42).

Population size is chosen usually from 100-500. Larger population may produce more robust solution.

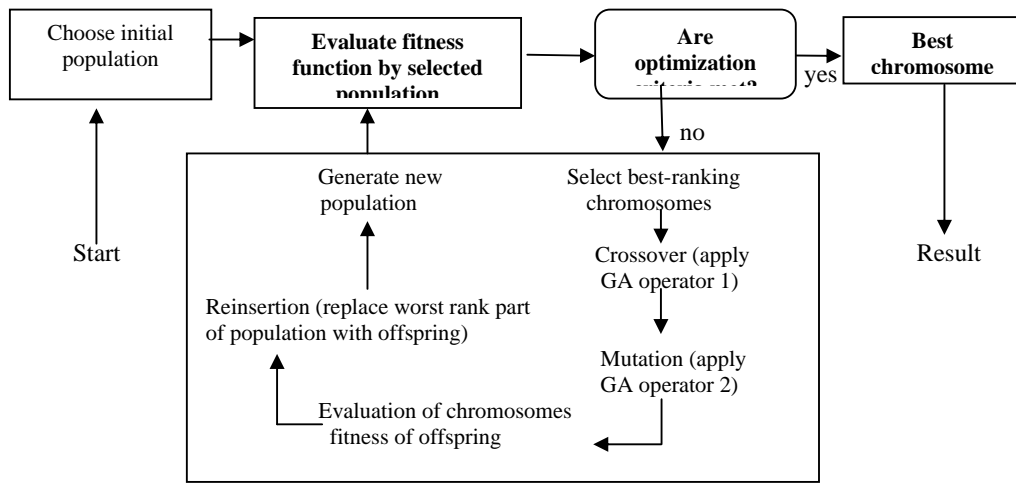


Fig-3: A GA structure

Source: Developed for this research

Step-2: Fitness scaling is used to provide a measure of how selected chromosomes perform in the problem domain. The model (1) fitness is evaluated through a criterion like RMSE (CC, MAE can be also used). For the model (1), we get RMSE:

26.19, 32.09, 53.75, 20.18, 18.67 and 66.64.

Table 2: GA options

Step	Algorithm Option
1. Creation	Uniform, normal
2. Fitness scaling	Rank, proportional, top (truncation), linear scaling, shift
3. Selection	Roulette, stochastic uniform selection, tournament, uniform
4. Crossover	Binary-valued (single-point, two-point, n-point, uniform), real-valued (intermediate, line)
5. Mutation	Gaussian, uniform
6. Reinsertion	Pure, uniform, elitist, fitness-based

Source: Developed for this research

Using the linear-ranking process (e.g.) (details, see Pohlheim (2005)), Fit(RMSE):

1.2, 0.8, 0.4, 1.6, 2.0, 0.

Step-3: Based on step-2 results, choose parents for the next generation. To understand it, consider the following distribution:

RMSE:	26.19	32.09	53.75	20.18	18.67	66.64
Fit(RMSE):	1.2	0.8	0.4	1.6	2.0	0
pdf of RMSE:	0.2	0.13	0.06	0.26	0.33	0
cdf of RMSE:	0.2	0.33	0.39	0.65	0.98	1.0

It is observed that chromosome 5 is the fittest chromosome, because it occupies the largest interval, whereas chromosome 3 is the second least fit chromosome as the smallest interval. Chromosome 6 is the least fit interval that has a fitness value of 0 and get no chance for reproduction. Using (e.g.) the roulette wheel method (purpose is to eliminate the worst chromosomes and to regenerate better substitutes), selected 4 parents are:

(0.20,0.74,0.52,0.37), (0.13,0.01,0.84,0.68), (0.27,0.46,0.83,0.70), (0.13,0.01,0.84,0.68).

Next step is to produce offspring from selected parents by combining entries of a pair of parents (known as crossover) and also by making random changes to a single parent (known as mutation).

Step-4 (GA operator-1): Basic operator for producing new (improved) chromosomes is known as crossover (a version of artificial mating). It produces offspring that have some parts of both parents genetic material. Offspring are produced using the intermediate crossover method, because this is a method proposed to recombine of parents with real valued chromosomes (details, Pohlheim (2005)). Thus, crossover offspring are:

(0.16,0.16,0.85,0.57), (0.13,0.22,0.76,0.43), (0.13,0.15,0.83,0.69), (0.26,0.45,0.84,0.68).

Step-5 (GA operator-2): Offspring are mutated after being produced crossover offspring and this GA operator increases the chance that the algorithm will generate better fittest RMSE than the step-4. GA creates 3 types of offspring: These are: Elite offspring (number of best RMSE values in the current generation that are guaranteed to survive to the next generation), crossover offspring and mutation offspring. To understand it, consider an example: Suppose population size is 20 and elite count is 2. If crossover fraction is 0.8, then the distribution of offspring: 2 elites, 14 ($18*0.8$) are crossover offspring and the remaining 4 are mutation offspring. Just to know, a crossover fraction of 1 means, all offspring other than elite are crossover offspring, while a crossover fraction of 0 means that all offspring are mutation offspring. How offspring are produced under the mutation process, see Pohlheim (2005). The mutation offspring are found (0.16,0.17,0.85,0.56), (0.13,0.22,0.76,0.43), (0.13,0.14,0.83,0.69), (0.26,0.45,0.84,0.68) respectively.

Step-6: Once offspring have been produced using steps 4-5, offspring fitness (i.e. RMSE values) must be determined (procedure similar to step-2). We get improved RMSE:

23.37, 28.13, 24.11, 18.62.

If less offspring are produced than the original population size, then to maintain size, offspring have to be reinserted into the old population. By this step, it is determined which chromosomes will be replaced by offspring. Using (e.g.) the fitness-based reinsertion method, the following RMSE are found:

20.18, 18.67, 18.62, 23.37, 24.11, 28.13.

If termination criteria are not defined, GA returns to the step-3 and continues up to the step 6. It is satisfied when either maximum number of generations is achieved or when all chromosomes in the population are identical (i.e. converge). The creator sets this number before running GA, which ensures that GA does not continue indefinitely.

Method of multiple linear regression

This method is developed based on the assumption of linearity and its main aim is to minimize error. The model (1) slope parameters are estimated by

$$\hat{\alpha}_i = (X'X)^{-1}(X'R_t), i=1,2,3$$

and intercept parameter is estimated by

$$\hat{\alpha}_0 = \bar{R}_t - \hat{\alpha}_1\bar{T}_t - \hat{\alpha}_2\bar{RH}_t - \hat{\alpha}_3\bar{CL}_t$$

where X includes T_t , RH_t and CL_t .

Design of considered techniques

The first 70% data for each selected stations are used as the training period and the rest 30% as the testing period (see Fig-4). To reduce error, an error and trial approach is used to find the computational settings for each selected AI techniques, which are reported in the Table 3.

Results and discussion

This section presents the results of the considered forecasting models.

The forecasting performances of the considered models are evaluated against two widely used statistical measures, namely, root mean square error (RMSE) and coefficient of determination (R^2), defined as follows:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\text{Actual } R_t - \text{Predicted } R_t)^2}$$

$$R^2 = 1 - [ESS/TSS]$$

where ESS is the sum of squares of differences between actual and predicted R_t values and TSS is the sum of squares of actual R_t . Note that RMSE is a measure of the accuracy of prediction through representing the degree of scatter and a smallest value of RMSE indicate higher accuracy in forecasting. R^2 is a measure of

Table 3: Computational settings for ANN, ANFIS and GA networks

ANN	ANFIS	GA
1) <i>Topology</i> - 3:10:7:1 i.e. 3 input nodes and two hidden layers consisting of 10 and 7 hidden	1) <i>Input</i> - T_t , RH_t and CL_t . 2) 2 mfs (type Gaussian) for each of input variables. Thus, 8 if-then fuzzy rules were learned.	1) <i>Population size</i> - 1500, generated at random from the uniform probability distribution. 2) <i>Fitness function</i> - RMSE of model (1);

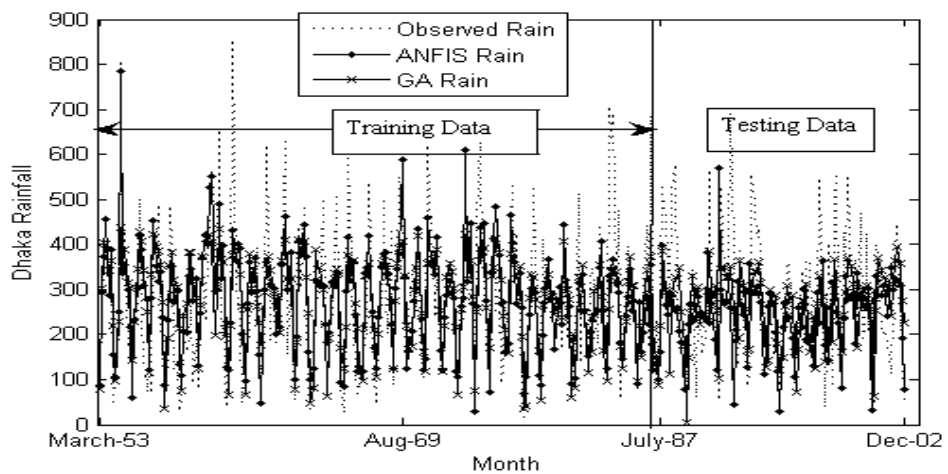
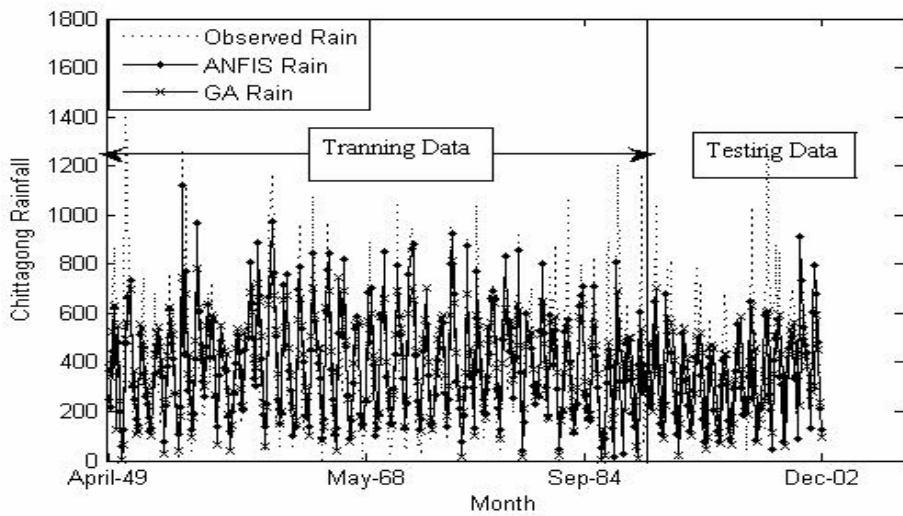
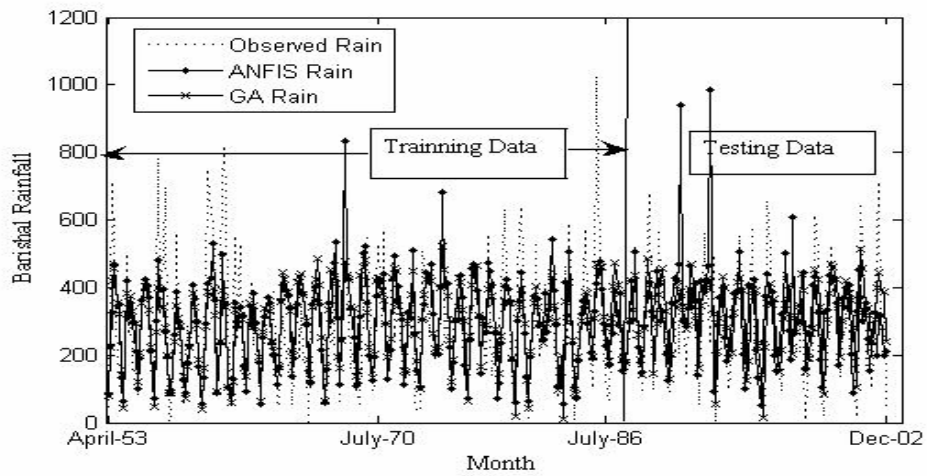
International Technology Management Review, October 2009, v2, n1.

<p>neurons respectively.</p> <p>2) <i>Learning rate</i>- 0.15.</p> <p>3) <i>Momentum parameter</i>-0.90.</p> <p>4) Activation function for the hidden neurons- <i>Purelin</i>.</p>	<p>Total parameters (44)= premise parameters (12) + consequent parameters (32)</p> <p>where premise parameters is calculated by # of inputs x # of mfs x # of parameters of Gaussian distribution and consequent parameters is calculated by # of mfs x # of parameters of Gaussian distribution x # of fuzzy rules.</p>	<p><i>Fitness scaling</i>- Linear ranking process.</p> <p>3) <i>Selection</i>-Roulette wheel.</p> <p>4) <i>Crossover</i>- fraction: 0.90; Intermediate method.</p> <p>5) <i>Mutation</i>- fraction: 0.05; Uniform method.</p> <p>6) <i>Reinsertion</i> –The fitness-based reinsertion method.</p> <p>7) <i>Termination Criteria</i>- Maximum # of generations, assumed 300.</p>
--	--	---

Source: Developed for this research

the accuracy of prediction of the trained network models. Higher R^2 value indicates better prediction. All computational works were carried out using the programming code of MATLAB (version 7.0). Table 4 summarizes the performances of different considered forecasting models achieved for the one-day ahead prediction R_t values using error measures RMSE and R^2 .

In terms of measure of RMSE, our training results show that for all stations, ANFIS performed better (noted smallest RMSE values (e.g. for the Barishal station, we observed RMSE (123.74), for the Chittagong station, RMSE (176.95) etc.)) than other forecasting models, followed by GA, ANN and MR. After the models are built using the training data, R_t is forecasted for each of stations over the testing data. The testing results (see Table 4) when compared to the GA, ANN and MR forecasting models, shows again the daily R_t forecasting ability of ANFIS (except for the Barishal station) is higher than the other models. We noted for the Barishal station, GA model performed (lowest value of RMSE (103.46)) better than the other forecasting models.



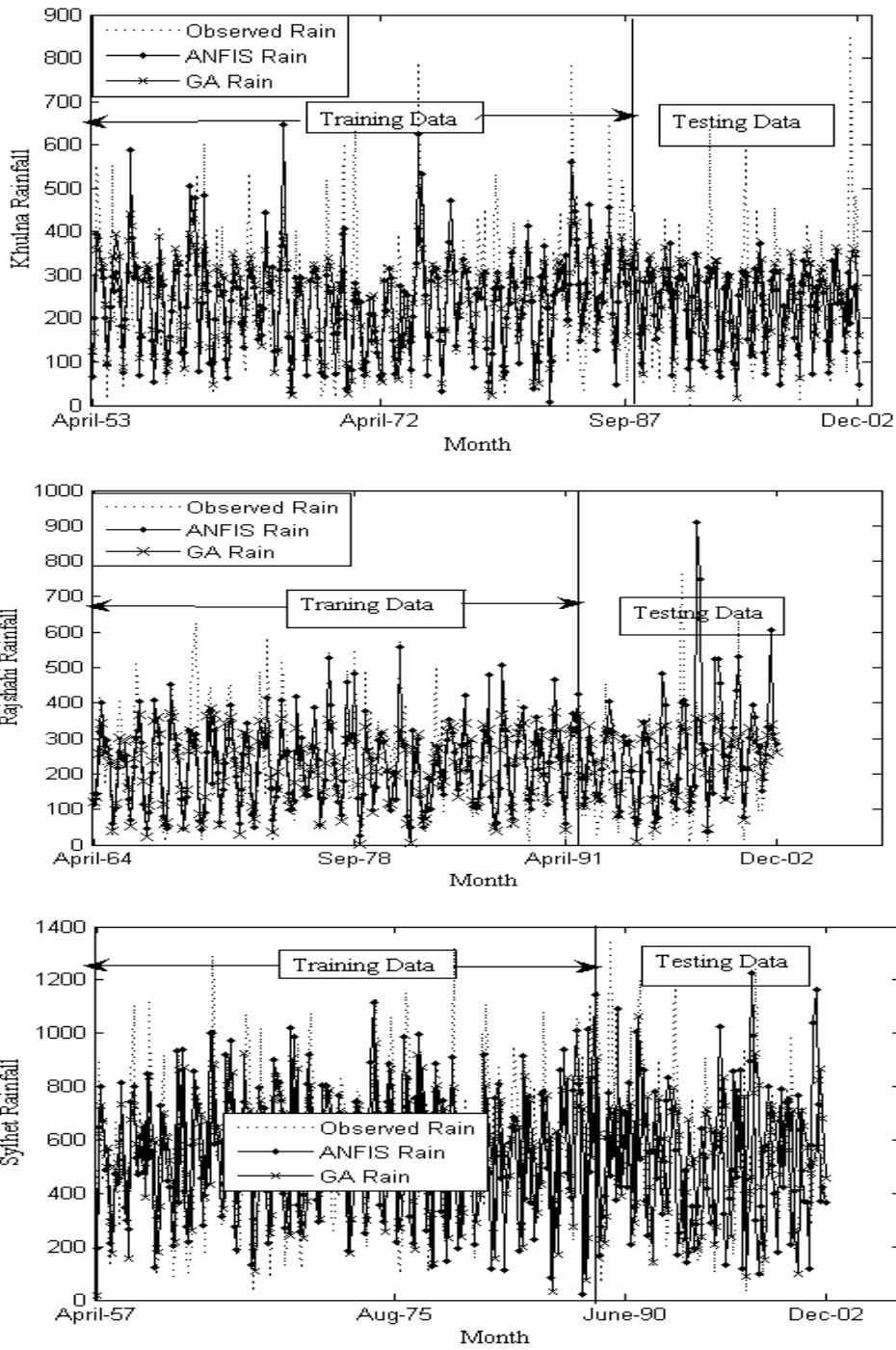


Fig-4: Training and testing results of the considered stations

Source: Developed for this research

Table 4: Performances of different considered models

Station	Measure	ANN		ANFIS		GA		MR	
		Train	Test	Train	Test	Train	Test	Train	Test
Barishal	RMSE	135.70	111.11	123.74	105.07	135.43	103.46	138.55	112.05
	R ²	0.84	0.85	0.87	0.88	0.84	0.88	0.82	0.83
Chittagong	RMSE	193.65	193.04	176.95	169.07	193.44	186.50	196.77	193.05
	R ²	0.82	0.82	0.88	0.89	0.84	0.83	0.80	0.81
Dhaka	RMSE	120.55	111.38	107.70	105.09	120.31	103.69	121.30	111.99
	R ²	0.85	0.86	0.89	0.86	0.86	0.87	0.83	0.84
Khulna	RMSE	112.60	105.41	97.63	94.61	112.38	98.14	115.31	107.42
	R ²	0.84	0.85	0.88	0.85	0.84	0.85	0.82	0.83
Rajshahi	RMSE	98.65	107.63	82.88	95.63	98.63	87.88	99.45	108.64
	R ²	0.85	0.82	0.90	0.86	0.86	0.82	0.81	0.83
Sylhet	RMSE	185.58	170.17	161.62	128.67	185.42	151.41	187.84	172.32
	R ²	0.91	0.92	0.93	0.91	0.91	0.93	0.89	0.90

Source: Developed for this research

The highest R² values (i.e. good match between actual and predicted R_t) indicate that ANFIS again outperforms GA, ANN and MR for all considered stations (see Table 4). For example, we noted for the Barishal station, the accuracy of prediction for the ANFIS model is 87%, for the GA model 84%, for the ANN model 84% and for the MR model 82% respectively.

To understand our findings graphically, see Fig-4, where only the ANFIS model and the GA model performances (to avoid clumsiness in figure) are shown with observed R_t. These figures again proved that ANFIS has better forecasting capability of daily R_t as compared to the GA forecasting model.

Thus, our study shows that for Bangladeshi monsoon rainfall modeling, ANFIS is better than other considered forecasting models.

Conclusion

Rainfall forecasting is important for many areas of human activities such as agriculture, water resources, hydroelectric power projects, happening of droughts or floods and others. This paper modelled the complex multi-dimensional behaviors of monthly monsoon rainfall for a number of Bangladeshi stations, namely Barishal station, Chittagong station, Dhaka station, Khulna station, Rajshahi station and Sylhet station using several soft computing techniques. We considered ANN, ANFIS and GA processes and results obtained by these models are also compared to the statistical forecasting method, namely linear multiple regression model to show advantages of proposing the AI forecasting models. Findings suggest that the ANFIS forecasting model and the GA forecasting model can be used to forecast monthly monsoon rainfall more accurately than the ANN model and the statistical model. We believe findings of this paper will be of interest to individuals for formulating wise policies related to rainfall in the context of the country, Bangladesh. Our future research plan is to model the daily monsoon rainfall data, which have significance in the field of agriculture, transportation, sports, tourism activities and others.

Acknowledgments

Authors are grateful to the editor and reviewers for their valuable comments, which improved the presentation of the paper. We are also grateful to the participants of the International Conference on Computer and Information Technology (ICCIT) organized by the Faculty of Electrical & Electronic Engineering, Khulna University of Engineering and Technology, Bangladesh and held on 25-27 December 2008.

References

Banik, S., Chanchary, F.H., Rouf, R.A. & Khan, K. 2007, 'Modeling chaotic behavior of Dhaka stock market index values using the neuro-fuzzy model', Proceedings of 10th ICCIT, pp.80-85.

Charhate, S.B. & Deo, M.C. 2007, 'Storm surge prediction using NN and GP', Paper Presented the 6th Conference on AI application to Environmental Science AMS Annual Meeting, 20-24 January, New Orleans Louisiana, USA.

Culloch, W.S. & Pitts, W. 1943, 'A logical calculus of the ideas immanent in nervous activity', B. Math Biohys, pp.115-133.

Holland, J.N.1975, Adaptation in natural and artificial systems, Ann Arbor, The University of Michigan Press.

Islam, M.N., Islam, S., Hayashi, T., Terao, T. & Hiroshi, U. 2002, 'Application of a method to estimate rainfall in Bangladesh using GMS-5 data', Journal of Natural Disaster Science, pp.83-89.

Jang, J.S.R. 1993, 'ANFIS: Adaptive-network-based fuzzy inference systems', IEEE Transactions on Systems, Man, and Cybernetics, pp.665-685.

Kamruzzaman, J. & Sarker, R.A. 2003, 'Comparing ANN based models with ARIMA for prediction of forex rates', ASOR Bulletin, pp.1-10.

Kihoro, J.M., Otieno, R.O. & Wafula, C. 2000, 'Seasonal time series forecasting: A comparative study of ARIMA and ANN models', African Journal of Science and Technology, pp.41-49.

Koza, J.R. 1992, Genetic programming: On the programming of computers by means of natural selection, Cambridge: MIT Press.

Kumar, A. & Debroy, T. 2007, 'Tailoring fillet weld geometry using a genetic algorithm and a neural network trained with convective heat flow calculations', Welding Journal, pp.26-33.

Lee, Y.H., Park, S.K. & Chang, D.E. 2006, 'Parameter estimation using the genetic algorithm and its impact on quantitative precipitation forecast', Annales Geophysicae, pp.3185-3189.

Mondol, R.U. & Shahid, D.S. 2004, 'A neural network approach for the prediction of monthly rainfall', Jahangirnagar University of Science, pp.87-95.

Navone, H.D. & Ceccatto, H.A. 1994, 'Predicting Indian monsoon rainfall: A neural network approach', Climate Dynamics, pp.305-312

International Technology Management Review, October 2009, v2, n1.

Pohlheim, H. 2005, Documentation for genetic and evolutionary algorithm toolbox for use with MATLAB.

Sen, Z. & Oztopal, A. 2001, 'Genetic algorithms for the classification and prediction of precipitation occurrence', *Hydrological Sciences*, pp.255-267.

Wahid, C.M. & Islam, M.N. 1999, 'Patterns of rainfall in the northern part of Bangladesh', *Journal of Science Research*, pp.115-120.

Wei, Z., Zhi-ming, W.U. & Gen-Ke, Y. 2004, 'Genetic programming based chaotic time series modeling', *Journal of Zhejjang University Science*, pp.1432-1439.

Whigham, P.A. & Crapper, P.F. 2001, 'Modeling rainfall-runoff using genetic programming', *Mathematical and Computer Modeling, Canberra, Australia*, pp.707-721.

© 2009 Academic Global Publications P/L. This work is copyright. You may download and print only one paper copy from this electronic file for your personal use only, from which you may not make any further paper copies.

International Technology Management Review, October 2009, v2, n1.