Jurnal
Teknologi

Full Paper

RAINFALL RUNOFF MODELING BY MULTILAYER PERCEPTRON NEURAL NETWORK FOR LUI RIVER CATCHMENT

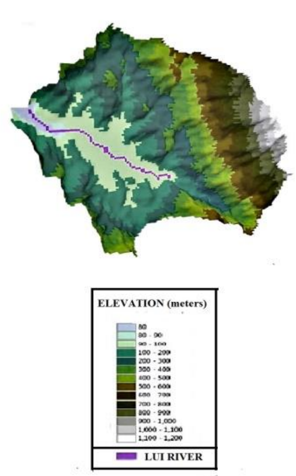
Nadeem Nawaz[a,c]*, Sobri Harun[a], Rawshan Othman[b], Arien Heryansyah[a]

[a]Department of Hydraulics and Hydrology, Faculty of Civil Engineering, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia
[b]Petroleum Department, Koya Technical Institute, Erbil Polytechnic University, 44001 Erbil, Kurdistan Regional Government, Iraq
[c]Faculty of Water Resources Management, Lasbela University of Agriculture, Water and Marine Sciences, 90150 Uthal, Balochistan, Pakistan

Article history
Received
25 May 2015
Received in revised form
20 November 2015
Accepted
22 December 2015

*Corresponding author
nn.engr96@yahoo.com

Graphical abstract



Abstract

Reliable modeling for the rainfall-runoff processes embedded with high complexity and non-linearity can overcome the problems associated with managing a watershed. Physically based rainfall-runoff models need many realistic physical components and parameters which are sometime missing and hard to be estimated. In last decades the artificial intelligence (AI) has gained much popularity for calibrating the nonlinear relationships of rainfall–runoff processes. The AI models have the ability to provide direct relationship of the input to the desired output without considering any internal processes. This study presents an application of Multilayer Perceptron neural network (MLPNN) for the continuous and event based rainfall-runoff modeling to evaluate its performance for a tropical catchment of Lui River in Malaysia. Five years (1999-2013) daily and hourly rainfall and runoff data was used in this study. Rainfall-runoff processes were also simulated with a traditionally used statistical modeling technique known as auto-regressive moving average with exogenous inputs (ARMAX). The study has found that MLPNN model can be used as reliable rainfall-runoff modeling tool in tropical catchments.

Keywords: MLPNN, ARMAX, rainfall-runoff modeling, Lui catchment

1.0 INTRODUCTION

The hydrologists are always dealing with the problem of determining the non-linear relationship between the rainfall and runoff processes. A good understanding of rainfall-runoff relationship is needed for hydrologic design and management. This relationship depends on many factors such as land use, soil moisture, evapotranspiration, infiltration, distribution, duration of rainfall and so on. The need for reliable modeling of the rainfall–runoff transformation process has been one of the major hydrological research activities for decades [1]. However, considering the high stochastic nature of the rainfall–runoff transformation process, many models are still being developed to simulate such a complex process that include physically based models, statistical models and data driven models. The data driven models are able to simulate direct relation between input and output without understanding physical behavior. In the recent past, the use of data driven models, e.g., Artificial Neural Network (ANN), Neuro-Fuzzy Systems (NFS) and Genetic Programming (GP) in water resources engineering has become viable [2-6]. Artificial neural network (ANN) is one the most popular data driven model and have many applications in modeling rainfall-runoff process [7-10]. The main objective of this study was to develop continuous and event based rainfall-runoff model based on MLPNN for Lui River catchment.

ANN is a data processing system consisting of a large number of simple, highly interconnected processing systems consisting of a large number of simple, highly interconnected processing elements (artificial neurons). The basic structure of network basically consists of three layers as can be seen in Figure 1, which include: (1) Input layer, where data is introduced to network; (2) Hidden layer or layers, where the data is processed; (3) Output layer, where the results are produced. The main control parameters of neural network model are interneuron connection and strength also known as weights and biases. There can be several hidden layers, with layer having one or more nodes.

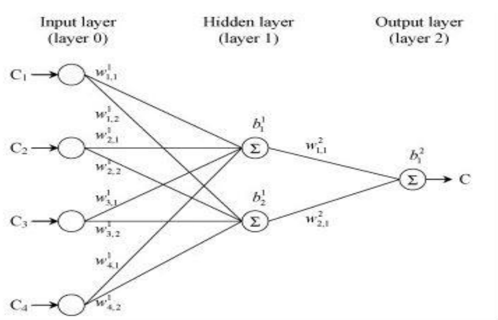


Figure 1 Basic structure of ANN

ANN is characterized by its architecture which represents the connection between nodes, its method of determining the connection weights, and the activation function [11]. ANN can be categorized based on the direction of information flow and processing. In feed-forward network the information passes from the input layer and ending at the final output layer. The nodes in one layer are connected to those in the next, but not to those in same layer. This, the output of a node in a layer is only dependent on the inputs it receives from previous layers and cross ponding weights. On the other hand in a recurrent ANN, information flows through the nodes in both directions [12].

Multilayer Perception (MLP) is a supervised and feed forward neural network with one or more layers of nodes between input and output nodes. It is a most commonly used neural computing technique. Each node is the basic element of a neural network called neuron. The decisions that affect the performance of the MLP models during training include the number of input nodes, the number of hidden nodes, learning rate, momentum constant and the transfer function. The accuracy of the model depends on the selection of input nodes derived from the characteristics of data series. In rainfall-runoff modeling the input nodes consist of rainfall series and the desired output is runoff. The reason for selecting ANFIS to infill missing rainfall data was due to its capability of simulating complex input and output relationship. It uses a combination of the least-squares method and the back propagation gradient descent method for training FIS membership function parameters for a given training data set.

2.0 METHODOLOGY

2.1 Study Area and Data Used

Lui River catchment is located at the Hulu Langat district, Selengor state, Malaysia (with an area of 68.1 km^2) as can be seen in Figure 2. Lui catchment has land surface elevations ranging from 80 to 1,200 meter above sea level.

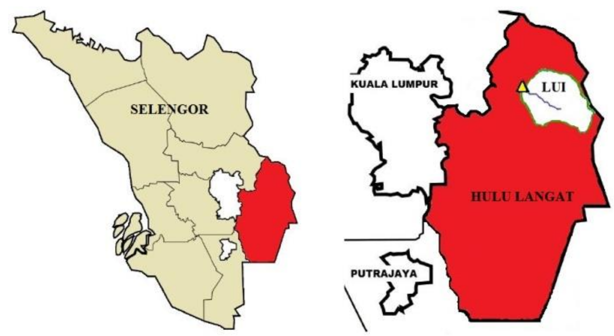


Figure 2 Location map of Lui River catchment

Approximately 87% of the area is mountainous, and valleys cover 13% of the catchment area. Heavy rainfall events are recorded in Malaysia because of its presence in tropical zone. Malaysia receives approximately 2400mm rainfall per annum [13]. The northeast monsoon contributes heavy rainfall events in the eastern part of Peninsular Malaysia that occur during November-February and the western part of Peninsular Malaysia receives southwest monsoon during May-August. Peninsular Malaysia receives the most rainy days in both monsoons. Lui River catchment is in the state that also experiences inter-monsoon period during March-April and September-October [14]. The daily and hourly data of the five rainfall stations and one runoff station was arranged from department of irrigation and drainage (DID), Malaysia. Five years (1999-2013) daily and hourly rainfall and runoff data was provided by the department of irrigation and drainage (DID), Malaysia. The daily data was used for continuous rainfall-runoff modeling and the hourly data was used for the event based modeling. All the rainfall and runoff data were normalized before analysis. Normalization concentrates the dispersed data into a defined interval and for this study the interval was kept from 0.1 to 0.9. The normalization method used in this study follows [15] which can be given by:

$$x_n = F_{min} + \left[\frac{x_i - x_{min}}{x_{max} - x_{min}}\right] \times (F_{max} - F_{min}) \quad (1)$$

where FMIN and FMAX are the required minimum and maximum of the new domain (e.g. 0.1-0.9), x_n is the standardized data, x_{min} and x_{max} are the minimum and maximum observed data, respectively; and x_i is the observed data.

2.2 Input Data Selection and MLPNN Model Development

As the catchment have five rainfall stations the sensitivity analysis were performed to select different input combinations. For developing the MLPNN model the number of input nodes were selected carefully as accuracy of the model is dependent on these nodes. The data was divided into three subsets: (1) training dataset used to train the model; (2) validation dataset used to validate the model; and (3) testing data set used to check the model performance. The training of MLPNN was accomplished by backpropagation algorithm. In general the process or procedure followed for the backpropagation algorithm can be summarized in 9 steps.

i. Obtain a set of training patterns.
ii. Setup neural network model (no. of input neurons, hidden neurons and output neurons).
iii. Set model parameters (Learning rate, momentum rate).
iv. Initialize all connections, weights and biases to random values.
v. Set minimum error.
vi. Start training by applying input and desired outputs and propagate through the layers then calculate total error.
vii. Backpropagate error through output and hidden layer and adapt weights.
viii. Backpropagate error through hidden and input layer and adapt weights.
ix. Check if error < minimum error. If not repeat steps 6-9. If yes stop training.

The hyperbolic-tangent (tansig) was used as activation function. This model was developed with one hidden layer. After training the validation dataset was passed through the network and errors over the dataset were calculated.

2.3 Model Performances

The performances of DENFIS model in this study were evaluated based on several statistical measures such as coefficient of efficiency (CE), coefficient of determination (R^2), Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Relative Peak Error (RPE).

$$CE = 1 - \frac{\sum_{i=1}^{n}(Q_i - \hat{Q}_i)^2}{\sum_{i=1}^{n}(Q_i - \bar{Q}_i)^2} \quad (2)$$

$$R^2 = \left[\frac{\sum_{i=1}^{n}(Q_i - \bar{Q})(\hat{Q}_i - \tilde{Q})}{\sqrt{\sum_{i=1}^{n}(Q_i - \bar{Q})^2} \times \sqrt{\sum_{i=1}^{n}(\hat{Q}_i - \tilde{Q})^2}}\right]^2 \quad (3)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(Q_i - \hat{Q}_i)^2}{n}} \quad (4)$$

$$MAE = \frac{\sum_{i=1}^{n}|Q_i - \hat{Q}_i|}{n} \quad (5)$$

$$RPE = \frac{|(Q_p) - (\hat{Q}_p)|}{(Q_p)} \quad (6)$$

where $\bar{Q}$ is the average observed discharge and n is the total number of the observations, Q_i is observed flow rate and $\hat{Q}_i$ is the simulated flow rate, Q_p and $\hat{Q}_p$ is the observed peak discharge and simulated peak discharge.

2.4 ARMAX Model

The conventional regression method use linear or piecewise-linear demonstration for forecasting function. In this mechanism, linear combination determines the functional relationship that supplies the requested forecast, which assumes linear relationship without adequate reasons. Furthermore the input-output functional relationship between observed phenomena and its underlying cause are more often not stationary in conventional regression like in the case of Rainfall Runoff process. Hence the conventional regression approach produces averaged results as it does not have enough adaptability to identify inherent spatio-temporal variation. ARMAX linear models with its improved efficiency for time series analysis have been developed by Box and Jenkins [16]. This model is frequently use because of producing acceptable predictions [17]. In this study, the ARMAX model was developed using different combinations of rainfall antecedents (up to present time) as well as stage antecedents (up to t-1) as exogenous inputs to estimate the runoff at present time t. The same training dataset used for MLPNN has been used to develop ARMAX model and the validation process was done using testing data set.

3.0 RESULTS AND DISCUSSION

Five years complete daily and hourly data was available for each of the rainfall stations and runoff station. The daily data was divided into three, two and one year for training, validation and testing respectively. Different input combinations were compared in testing face and on the basis of best input selection the model was developed for the Lui River catchment. Initial analyses showed that out of five rainfall stations the data for three rainfall stations had good correlation with the runoff data. Different combinations of two, three and four stations were tested for the model development. The different combinations were also tested with Q(t-1), Q(t-2) and Q(t-3). It was found that taking Q(t-1) as input the model performance was much better. So it was decided to keep Q(t-1) as a constant input with other combinations. Table 1 shows the different input combinations and their performances obtained in the testing phase.

Table 1 MLPNN performances in testing phase for different input combinations

Input Selection					Performances in Testing Phase				
Input1	**Input2**	**Input3**	**Input4**	**Input5**	**CE**	**R^2**	**RMSE**	**MAE**	**RPE**
R1(t)	R1(t-1)	R4(t-2)	Q(t-1)		0.866	0.878	4.469	2.440	0.149
R2(t-1)	R2(t-2)	R2(t-3)	Q(t-1)		0.836	0.909	7.668	5.028	0.170
R2(t-2)	R2(t-3)	R2(t-4)	Q(t-1)		0.960	0.966	3.665	2.027	0.150
R1(t)	R2(t-1)	R4(t-2)	Q(t-1)		0.761	0.809	5.001	2.380	0.112
R1(t-1)	R1(t-2)	R1(t-3)	Q(t-1)		0.829	0.863	8.017	4.322	0.259
R1(t-2)	R1(t-3)	R1(t-4)	Q(t-1)		0.850	0.885	5.764	3.239	0.168
*R1(t)	R4(t-1)	R5(t-1)	Q(t-1)		0.973	0.974	3.142	1.915	0.101
R1(t-3)	R1(t-4)	R2(t-3)	R2(t-4)	Q(t-1)	0.969	0.969	3.406	2.936	0.117

*Input Selection with best performance

The combination of (R(t), R4(t-1), R5(t-1) and Q(t-1)) gave better result in testing phase then other input combinations. Comparison of the simulated time series by the MLPNN model and the observed is shown in Figure 3.

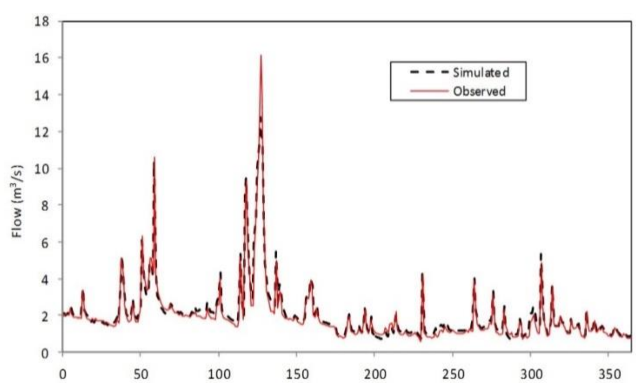


Figure 3 Observed and simulated hydrograph with best input combinations in testing phase

The MLPNN was able to predict both low and high runoff reasonably well. Figure 4 shows the scattered plot obtained from the MLPNN model in testing phase.

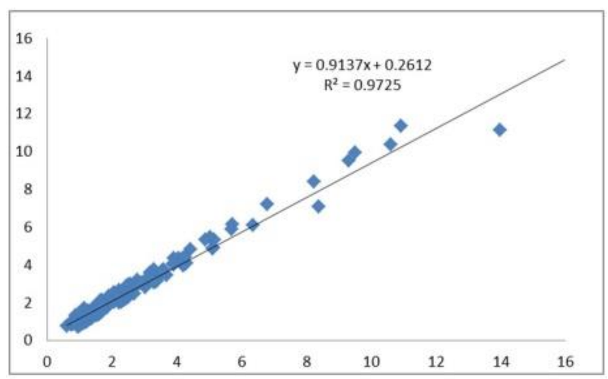


Figure 4 Scattered plot obtained from the MLPNN best performance

For further comparison, the results obtained by MLPNN model were compared with the ones obtained by ARMAX. The ARMAX model was developed with the same input which gave best performances for the MLPNN model. Table 2 shows the comparison of MLPNN and ARMAX model in terms of statistics obtained from coefficient of efficiency (CE), coefficient of determination (R^2), Root Mean Square Error (RMSE), Mean Abso-lute Error (MAE) and Relative Peak Error (RPE).

Table 2 Comparison of MLPNN and ARMAX model performances

Model	**CE**	**R^2**	**RMSE**	**MAE**	**RPE**
MLPNN	0.973	0.974	3.142	1.915	0.101
ARMAX	0.772	0.771	5.117	3.225	0.198

For event based rainfall-runoff modeling twenty extreme events were extracted from the hourly data (1999-2013). The events were randomly chosen for the calibration and testing. Out of twenty events sixteen events were selected for the calibration phase and four events for testing phase. The MLPNN model was developed using same input combination that gave good performance for the continuous modeling. The model was trained with the selected sixteen events. Figure 5 shows the observed and simulated hydrograph obtained from the MLPNN model. The ARMAX model was also developed but that was unable to capture extreme events. (RPE).

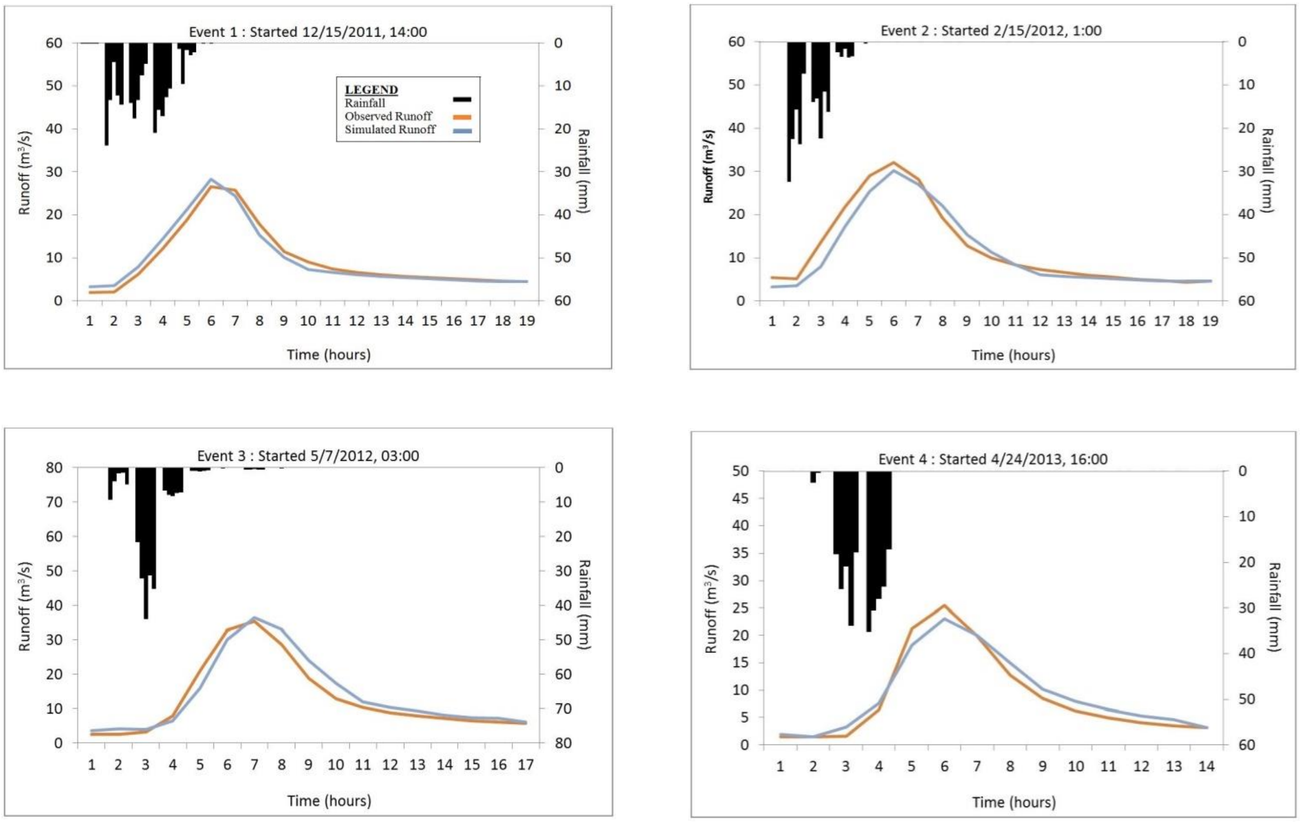


Figure 5 Hyetograph and hydrograph for the four testing events

The present study shows a successive application of MLPNN model for both continuous and event based rainfall-runoff modeling. In literature, a number of studies on the successful applications of MLPNN in rainfall-runoff modeling can be found [18-19]. Earlier the study conducted by [20] reported that MLPNN performance was better then other modeling tools. The results of the present study support the previous findings in literature on the successful application of MLPNN in rainfall-runoff modeling and also confirm its superiority over traditionaly used ARMAX model in simulating both low and high flows.

4.0 CONCLUSION

This study was performed for the continuous and event based modeling of rainfall- runoff processes for the Lui catchment. The MLPNN model was selected to perform this simulation. The results of this study have shown the ability of MLPNN for simulation of the complex relationship between rainfall and runoff processes. The MLPNN model gave good performances based on all the statistical measures used in this study. The traditionally used statistical model ARMAX for solving non-linear time series relationships was also used to model rainfall-runoff process. The statistical model was not able to simulate extreme events. Moreover the MLPNN was able to simulate peak discharges which show the superiority of the model to capture flood events.

Acknowledgement

The authors would like to express their gratitude towards the Department of Irrigation and Drainage for provision of hydrological data for Lui River catchment and copiously thankful to Lasbela University of Agriculture, Water and Marine Sciences for financial support to the scholar (PD-MS/LUAWMS/315).

References

[1] Shoaib, M. Shamseldin, A. Y. and Melville, B. W. 2014. Comparative Study Of Different Wavelet Based Neural Network Models For Rainfall–Runoff Modeling. *Journal of Hydrology*. 515(0): 47-58.

[2] Akbari, M. Afshar, A. and Sadrabadi, M. R. 2009. Fuzzy Rule Based Models Modification by New Data: Application to Flood Flow Forecasting. *Water Resources Management*. 23(12): 2491-2504.

[3] Anomaa Senaviratne, G. M. M. M. Udawatta, R. P. Anderson, S. H. Baffaut, and C. Thompson, A. 2014. Use of Fuzzy Rainfall–Runoff Predictions For Claypan Watersheds With Conservation Buffers In Northeast Missouri. *Journal of Hydrology*. 517(0): 1008-1018

[4] Babovic, V. and Keijzer, M. 2005. Rainfall-Runoff Modeling Based on Genetic Programming. *Encyclopedia of Hydrological Sciences*.

[5] Asadi, S. Shahrabi, J. Abbaszadeh, P. And Tabanmehr, S. 2013. A New Hybrid Artificial Neural Networks For Rainfall–Runoff Process Modeling. *Neurocomputing*. 121(0): 470-480.

[6] Beriro, D. J. Abrahart, R. J. and Paul, N. C. 2013. Comparison Of Genetic Programming With Neuro-Fuzzy Systems For Predicting Short-Term Water Table Depth Fluctuations by Jalal Shiri & Ozgur Kisi [Computers and Geosciences (2011) 1692–1701]. *Computers & Geosciences*. 56(0): 216-220.

[7] Kisi, O. Shiri, J. and Tombul, M. 2013. Modeling Rainfall-Runoff Process Using Soft Computing Techniques. *Computers & Geosciences*. 51(0): 108-117. Modeling Rainfall-Runoff Process Using Soft Computing Techniques. *Computers & Geosciences*. 51(0): 108-117.

[8] Ghafari, G. and Vafakhah, M. 2012. Rainfall-runoff Modeling Using Artificial Neural Networks And Adaptive Neuro-Fuzzy Inference System Models. *Uncertainty Modeling in Knowledge Engineering and Decision Making*. 7: 951-956.

[9] Tayfur, G. and Singh, V. 2006. ANN and Fuzzy Logic Models for Simulating Event-Based Rainfall-Runoff. *Journal of Hydraulic Engineering*. 132(12): 1321-1330.

[10] Wu, C. L. and Chau, K. W. 2011. Rainfall–Runoff Modeling Using Artificial Neural Network Coupled With Singular Spectrum Analysis. *Journal of Hydrology*. 399(3-4): 394-409.

[11] Fausett, L. V. and Hall, P. 1994. *Fundamentals Of Neural Networks: Architectures, Algorithms, And Applications*. Prentice-Hall Englewood Cliffs.

[12] Govindaraju, R. S. 2000. Artificial Neural Networks In Hydrology. I: Preliminary Concepts. *Journal of Hydrologic Engineering*. 5(2): 115-123.

[13] Che-Ani, A. L. Shaari, N. Sairi, A. Zain, M. F. M. and Tahir, M. M. 2009. Rainwater Harvesting As An Alternative Water Supply In The Future. *European Journal of Scientific Research*. 34(1): 132-140.

[14] Yakubu, M. L. Yusop, Z. and Fulazzaky, M. A. 2014. The Influence of Rain Intensity on Raindrop Diameter and the Kinetics of Tropical Rainfall: A Case study of Skudai, Malaysia. *Hydrological Sciences Journal*. 1-8.

[15] Van Ooyen, A. and Nienhuis, B. 1992. Improving The Convergence Of The Back-Propagation Algorithm. *Neural Networks*. 5(3): 465-471.

[16] Box, G. E. Time Series Analysis: Forecasting and Control.

[17] Bras, R. L. and Rodríguez-Iturbe, I. 1985. *Random Functions And Hydrology*. Courier Corporation.

[18] Drasko, F. 1998. Application Example Of Neural Networks For Time Series Analysis: Rainfall–Runoff Modeling. *Signal Processing*. 64(3): 383-396.

[19] Riad, S. Mania, J. Bouchaou, L. and Najjar, Y. 2004. Rainfall-Runoff Model Usingan Artificial Neural Network Approach. *Mathematical and Computer Modelling*. 40(7-8): 839-846.

[20] Piotrowski, A. P. and Napiorkowski, J. J. 2013. A Comparison Of Methods To Avoid Overfitting In Neural Networks Training In The Case Of Catchment Runoff Modelling. *Journal of Hydrology*. 476(0): 97-111.