Modelling daily flows of River Benue using Artificial Neural Network approach

Otache, Y. M., Mohammed, A. S., Ahaneku, I. E. and Egharevba, N. A.
Department of Agricultural & Bioresources Engineering,
Federal University of Technology, Minna, Nigeria
E-mail: martynso_pm@yahoo.co.uk

Abstract

The importance of understanding the dynamics and forecasting of streamflow processes of a particular river finds relevance in the fact that the physical mechanisms governing flow dynamics act on a wide range. In view of this, this study presents a simple basis for and application of Artificial Neural Network (ANN) methodology as an alternative modelling tool for predicting flow. To this end, the main focus is the development of ANN model for short term streamflow forecasting of the Benue River using univariate time series; inter alia, evaluate its performance of extreme events. It is evident from the modelling framework that the application of the knowledge of evolution of a dynamical system in a multi-dimensional state space is a robust approach for determining the size of an ANN model input. The ANN model forecast performance showed that reliable short term forecasts, 5 day - ahead can be made for the daily streamflow series based on CE and R^2 performance indexes. However, on the general question of the suitability of ANN model application for streamflow forecasting as applied in this study (i.e., daily streamflow), though the neural network could simulate the different attributes of the flow hydrograph, its relative forecast performance of high flows is robustly better than the case of low flows; it grossly under predicted and over predicted same depending on the particular network input data pre-processing schema. The forecast performance results also indicated that, for feed-forward MLP networks, with a tan-sigmoid transfer function, standardising the data by subtracting the mean and dividing by the standard deviation is better than rescaling the data to a small interval of particular range. Considering the findings, to appropriately capture the dynamics of the flow regime, it is necessary to include exogenous variables of the runoff generating process in the network input data base.

Keywords: *System-theoretic, nonlinear dynamics, phase-space reconstruction, neural networks, modelling*

INTRODUCTION

The most important process in the hydrologic cycle is the section where rainfall occurs and results in flow; this flow is critical to many activities such as assessing how much water may be extracted from a river for water supply or irrigation (Delleur *et al.*, 1976). Because the accuracy of flow estimation is very important, models that deal with meteorological, hydrologic, and geological variables should be improved so that controlling water and operating water structures effectively will be possible (Ozgur, 2004). There are many mathematical models to predict future flow such as those given by Hurst (1951), Matalas (1967), Box and Jenkins (1976), and Delleur *et al.* (1976). The mathematical models applied for real-time hydrological forecasting, broadly, are of two types: black-box or system-theoretic and conceptual models (Singh, 1988). In a conceptual type model, the internal descriptions of the various sub-processes are modelled attempting to represent, in a simplified way, the known physical processes. Black-box or system-theoretic (data-driven) models are stochastically-based and empirical (Elena and Armando, 2002). They are based primarily on observations and seek to characterise system response from those data.

The modelling technique that adheres most closely to the black-box principle is the use of artificial neural networks (ANN). Inspired by the biological nervous system, neural network technology is being used to solve a wide variety of complex scientific, engineering, and business problems. When using artificial neural networks for forecasting, the modelling principle employed is similar to that used in traditional statistical approaches. In the hydrological context, as in many other fields, artificial neural networks are increasingly used as black-box, simplified models (Bishop, 1994). The advantage of Neural networks and the reasons why they fall firmly into black-box category are that, like their biological counterparts, a neural network can learn, and therefore can be trained to find solutions, recognise patterns, classify data, and forecast future events. Unlike analytical approaches commonly used such as the unit hydrograph method or time series analysis, neural networks require no explicit model or limiting assumptions of normality or linearity. Artificial neural networks are now widely accepted as a potential useful way of modelling complex non-linear and dynamic systems for which there are large amounts of sometimes noisy data. They are particularly useful in situations where the underlying physical process relationships are not fully understood or where the nature of the event being modelled may display chaotic properties.

Unlike mathematical models that require precise knowledge of all the contributing variables, a trained artificial neural network can estimate process behaviour even with incomplete information. It is a proven fact that neural networks have a strong generalisation ability, which means that once they have been properly trained, they are able to provide accurate results even for cases they have never seen before (Hecht-Nielsen, 1991; Haykin, 1994; Ozgur Kisi, 2004). The generalisation ability of the neural network really underscores the very basic need for its application to real world situation. Many of the available techniques for time series analysis assume linear relationships among variables; but in the real world, however, temporal variations in data do not exhibit simple regularities and are difficult to analyse and predict accurately. It seems necessary that nonlinear models such as artificial neural networks (ANNs), which are suited to complex nonlinear systems be used for the analysis of real-world temporal data; especially, the inherently nonlinear relationships between input and output variables complicates attempts to forecast streamflow events.

It must be pointed out that the use of neural networks does not preclude the need for knowledge or prior information about the systems of interest. However, they merely reduce the model's reliance on this prior information whilst totally removing the need for the model builder to be able to correctly specify the precise functional form of the relationship that the model seeks to represent. Against the backdrop of the fact that there is no reported work on modelling and forecasting done on this river in available literature , the exclusive objective here is to present a simple basis for and application of artificial neural network (ANN) methodology as an alternative modelling tool for predicting flow data. To this end, the main focuses are the development of Artificial Neural Network (ANN) model for short term streamflow forecasting, in this case, a univariate time series (daily flow series), and to determine which characteristics of the model have the greatest impact on model performance.

MATERIALS AND METHODS

Hydrology of the Study River

In this study, historical time series for gauging stations at the base of the Benue River (i.e., Lower Benue River Basin) at Makurdi (7°44′ N, 8°32′ E) was used. A total of 26 years (1974–2000) water stage and daily discharge data were collected. An existing rating curve was used to convert the respective stage data to their corresponding discharge values. The Benue River is the major tributary of the Niger River. It is approximately 1 400 km long and almost navigable during the rainy season (between July and October). Its headwaters rise in the Adamawa Plateau of the Northern Cameroon, flows into Nigeria south of the Mandara Mountains through the east-central part of Nigeria. There is only one high-water season because of its southerly location; this normally occurs from May to October, while on the other hand, the low-water period is from December to June. Figure 1 explains the hydrological flow regime of the Benue River in line with the general climatic pattern. There are definite wet and dry seasons which give rise to changes in river flow and salinity regimes. The flood of the Benue River (upper, middle, and downstream) lasts from July to October, and sometimes up to early November.

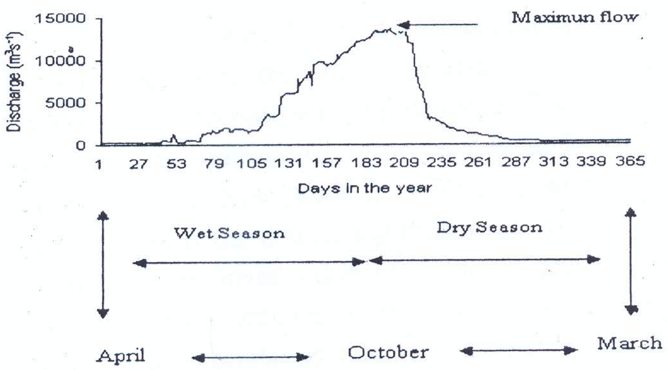

Fig. 1: General hydrological year flow regime

MODELLING FRAMEWORK

Network Design and Topology

When neural networks are used to build a function that estimates the process behaviour, the central issue is the determination of appropriate network structure. One of the ways to address this problem involve carrying out initial investigation of statistics like autocorrelation and cross-correlation which may explain the variance by multi-linear regression, and the Akaike criterion for ARMA model of corresponding order. However, cross-correlation approach can only be of useful application where there is a number of different input variables (i.e., multi-variate time series), and may not provide any meaningful relevance in the case of univariate time series. For a univariate time series, phase-space reconstruction using embedding dimension, which is based on dynamical systems theory is of better use. This approach was adopted for the determination of the appropriate number of input neurons for this study.

To describe the temporal evolution of a dynamical system in a multi-dimensional state space with a scalar time series, the time delay coordinate method (Takens, 1981; Packard *et al.*, 1980) can be used to reconstruct the state space. The method requires that the state vector X_i in a new space, the embedding space be formed from time delayed values of the scalar measurements $\{Y_i\}$ as $X_i = [Y_i, Y_{i-\tau}, \ldots, Y_{i-(m-1)\tau}]$, where Y_i is the observed value of the time series at time i, m is the embedding dimension, and τ is the delay time. To do this, It is expected that an optimum value of τ should give the best separation of neighbouring trajectories within the minimum embedding phase-space. Because of the strong annual seasonality, the autocorrelation function value of τ may become increasing large, and too, choosing τ value = 1 may result in the phase-space being redundant and consequently, lead to loss of valuable information (Wang, 2006). Besides, there could be intermittency problem in the data. Thus, to circumvent this problem, for the fear of the data being intermittent, τ was set to 78 based on the analysis of the autocorrelation function of the daily streamflow series (i.e., the point where the autocorrelation function plot first crosses the zero line) as shown in Fig. 2.

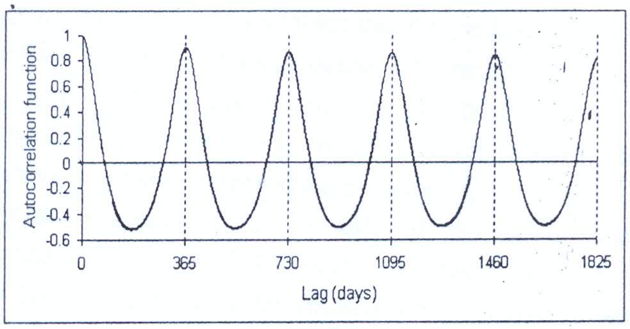

Fig. 2: Correlogram of the raw daily flow series

Because of the problems associated with the mutual information and autocorrelation methods, the method proposed by Kennel *et al.* (1992), called the 'false nearest neighbour' method to determine the minimal sufficient dimension m was used.

For this method, suppose the point
$X_i = [Y_{i-p+1}, \ldots, Y_i]$ has a neighbour $X_j = [Y_{j-p+1}, \ldots, Y_j]$ in a p-dimensional space, then the distance $\lVert X_i - X_j \rVert$ is calculated in order to compute

$$R_i = \frac{|Y_{i+1} - Y_{j+1}|}{\|X_i - X_j\|} \quad (1)$$

If R_i exceeds a given threshold R_τ (a suitable value is $10 \le R_\tau \le 50$), the point X_i is marked as having a false nearest neighbour. As a consequence, the embedding dimension p is high enough if the fraction of points that have false nearest neighbours is actually zero, or sufficiently small, say, smaller than a criterion R_f. For this study, the false neighbour threshold R_τ was set to 10. Based on this, the fraction of false nearest neighbours as a function of the embedding dimension, here, for daily streamflow series was calculated. Figure 3 shows the details of the computed fraction of false nearest neighbours. Thus, if the fraction criterion R_f is set equal to 0.01, the minimal embedding dimension will be 8; this implies that the state of the streamflow process can be determined by eight lagged observed values.

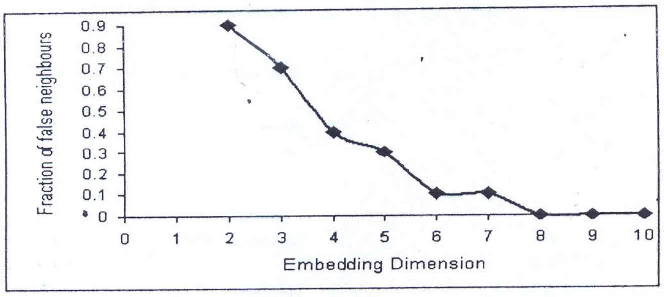

Fig. 3: Fraction of false nearest neighbours as a function of embedding dimension

Following from the analysis, eight lagged values of input variables can be used when fitting the ANN model to the series; specifically, this implies, based on the phase space reconstruction, the discharges are: Q_{t-7}, Q_{t-6}, ...Q_t of day t-7, to day t. The eight lagged input values were used to forecast the discharge from time t+1, i.e., the next day, to t+5; that is, 5-ahead values, using a multiple-output approach rather than a single-output. Figure 4 shows the complete network configuration based on the results of the phase-space reconstruction and multiple-output strategy using feed forward multi-layer perceptron network (MLP); the number of hidden layer neurons was determined to be 7 based on the typical 'trial and error' approach.

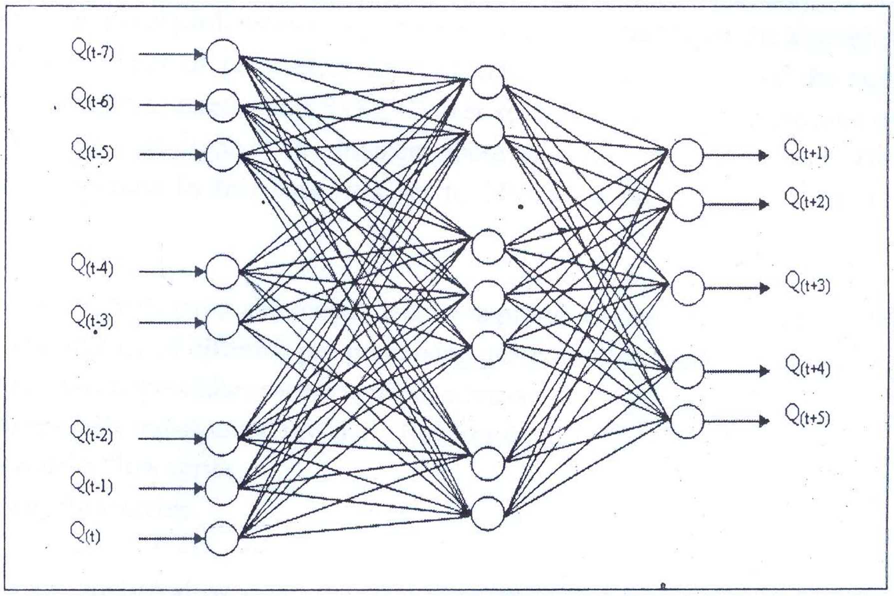

Fig. 4: Schematic of three-layer feed forward ANN architecture used for flow prediction.

2.2.2 Network Training and Data base Management

Training a neural network to predict the properties of the input-target relationship consists essentially of teaching the network the relationship between the input ingredients and its performance. This learning process can take place in a supervised or unsupervised manner. For the purposes of this study, supervised learning was used. The entire time series of length of 9490 daily values was thus partitioned into two-sets of 8670 and 730 data points corresponding to training and validation, respectively. The success of the training procedure depends largely on the power of the optimisation method used to search for the best parameter estimates; training was implemented using the '*trainbr*' function in MATLAB Neural Network Toolbox. The Bayesian regularisation training algorithm (trainbr) was chosen because of its ability to overcome generalisation problems that do result from over-fitting. Also important in neural network training is the transfer function; generally, predictability of future behaviour is a direct consequence of the correct identification of the system transfer function. For the identified network structure in this study, the '*tansigmoid*' and '*purelin*' transfer functions were used in the hidden and output layers, respectively. The '*purelin*' transfer function was considered for the output layer because it allows the network outputs to take on any value, whereas the last layer of a multi-layer network with sigmoid neurons constrains the network outputs to a small range. Instead of making predictions based on an ensemble of neural networks trained for the same task, a single best ANN was used; this was determined after series of attempts using different network structures, based on the mean squared error (MSE) of the network performance during training.

Training of the network was done with some specific parameter considerations. To this end, a decreasing learning rate (0.2 to 0.05) was used to accelerate convergence toward a global minimum; and since without momentum, a network may get stuck in a shallow minimum as it allows the network to ignore small features in the error surface, its value was set to 0.9. A very important issue to consider in the training of an ANN is how to decide when to stop the training because ANNs are prone to either under-fitting or over-fitting if the training is not appropriately stopped. In this regard, when using '*trainbr*', it is important to let the algorithm run until the effective number of parameters has converged; constant values of the sum squared error (SSE) and sum squared weights (SSW) over several iterations are indicative of convergence. So, training was stopped the moment these signals were experienced. The maximum number of iteration in this case was set to 2500 to avoid over-fitting of the network.

Before applying the ANN, both input and output data were pre-processed and normalized. In order to compare the influence of different pre-processing procedures on model performance, three different pre-processing procedures were applied, namely:

- Rescaled logarithmically transformed raw daily flow series
- Standardised raw daily flow series
- Rescaled raw daily flow series

Standardisation was accomplished by using the long term mean and standard deviation for both the training and validation data sets respectively. The '*trainbr*' algorithm generally works best if the network inputs and targets are scaled so that they fall approximately in the range [-1 1]. Thus, rescaling was done to ensure that the data series fall within this bound. Scaling of the original data, say   to the network range was done by

$$X_t = \frac{(U_x - L_x)x_t + (M_x}{M_x - m_x} \qquad (2)$$

where are the original input data, and the input data scaled to the network range, and are respectively the maximum and the minimum of the original input data, while and are the upper and the lower network ranges for the network input respectively. Similarly, the original output, say was scaled to the network range by

$$y_t = \frac{(U_y - L_y)y_t + (M_y L_y -}{M_y - m_y} \qquad (3)$$

where the systems' output was scaled to the network range, and are respectively the maximum and minimum values of the original output data, whereas and are respectively the upper and the lower network ranges for the network output. After scaling the inputs and outputs, the resulting output, say is in the scaled domain. Hence, one needs to rescale the output back to its original domain; this was by inverting Equation (3) and using as

$$\hat{y}_t' = \frac{(M_y - m_y)\hat{y}_t - (M_y L_y -}{U_y - L_y} \qquad (4)$$

In order to draw conclusions on the ANN model performance, parameters for statistical analyses (e.g. Root Mean Squared Error (RMSE), Mean Squared Relative Error (MSRE), Mean Absolute Error (MAE), Coefficient of Efficiency (CE), and Coefficient of Determination (R^2) were used to evaluate the ANN model predictions. Here, special attention is on the ANN model performance in terms of extreme events, that is, maximum and minimum flows. In this regard, the coefficient of correlation R as in Equation (5) was used:

$$R = \frac{\frac{1}{v}\sum_{t=1}^{v}[y_t - \mu_y][\hat{y}_t - \hat{\mu}_y]}{\left[\frac{1}{v}\sum_{t=1}^{v}[y_t - \mu_y]^2\right]^{1/2}\left[\frac{1}{v}\sum_{t=1}^{v}[\hat{y}_t - \hat{\mu}_y]^2\right]^{1/2}} \qquad (5)$$

where v = the number of output data points, y_t = the observed flow, $\hat{y}_t$ = predicted flow, μ_y = mean of observed flow, and $\hat{\mu}_y$ = mean of predicted flows. In terms of the measures of forecast accuracy with respect to extreme values, the ratio of the forecasted maximum to the observed maximum (peak) was determined as

$$R_{max} = \frac{\hat{y}_t}{max\{y_t\}} \times 100 \quad (6a)$$

where max{y_t} = max{$y_1,\ldots,y_v$} and $\hat{y}_t$ is the forecast corresponding to such maximum; and R_{max} = 100%, means that the observed peak is perfectly reproduced by the model. Forecasts with values of R_{max} about 100% are considered to be very accurate, while $R_{max} < 100\%$ indicates that the model underestimates the peak value; and $R_{max} > 100\%$ indicates overestimation. Similarly, the ratio of the forecasted to the observed minimum,

$$R_{min} = \frac{\hat{y}_t}{min\{y_t\}} \times 100 \quad (6b)$$

where $\hat{y}_t$ now represents the forecast corresponding to the minimum observed value, was also used to judge the forecast capability of the model.

MODEL FORECAST PERFORMANCE AND DISCUSSION

Artificial neural networks (ANNs) are like conventional hydrological models in that different attributes of the hydrograph are simulated to varying degrees of success. Considering the issues involved in modelling within this context, as in any other forecasting procedure, forecasting based on ANNs has an associated uncertainty. The forecast uncertainty arises not only because of the model but could be due to limited sample size used for training. Within the premise of the focus of this study, the model uncertainty may be due to two factors; first, the streamflow generation which is directly intertwined with hydroclimatic forcing, and second, probably, the model order. Looking at this issue further, the type of forecast model selected herein has been pre-defined, i.e. a neural network type of model, is a mathematical artefact that has some practical appeal but no physical basis, and hence not without uncertainty. Based on the forecast results, the uncertainty associated with the model is minimal since its architecture was determined based on reconstruction of phase-space dynamics of the input data series.

The ANN model forecast performances are as reported in Tables 1 and 2, for both the training and validation periods, respectively; Table 3 shows the forecast performance of the ANN model (see Fig. 4: 8-7-5 feed-forward architecture with bias) in terms of extreme events. Based on the statistical details of the forecast performance as presented in Tables 1 and 2, the overall performance based on CE and R^2 indexes, the feed-forward MLP ANN (8-7-5) proposed here is robust enough and do indicate that probable short-term forecasts can be made if proper forecast function is developed accordingly.

Table 1: ANN forecast performance during Training period

Transform	Lead	MAE	MAPE	RMSE	MSRE	CE	R^2
Resc-log							
	1	60.5122	0.0423	65.2061	0.0127	0.9948	0.9949
	2	78.1075	0.0667	72.4789	0.0230	0.9900	0.9901
	3	97.9276	0.0894	118.4867	0.0337	0.9845	0.9845
	4	120.7359	0.1105	129.2995	0.0478	0.9781	0.9782
	5	128.1495	0.1309	140.4846	0.0633	0.9713	0.9714
Standardised							
	1	35.6221	0.0528	60.9822	0.0192	0.9952	0.9952
	2	48.1363	0.0820	69.9252	0.0335	0.9905	0.9905
	3	90.5212	0.1096	105.9536	0.0483	0.9852	0.9850
	4	103.1272	0.1370	118.1302	0.0687	0.9793	0.9792
	5	109.8650	0.1656	123.3604	0.0923	0.9730	0.9731
Resc-raw data							
	1	68.9218	0.0494	73.8954	0.0186	0.9951	0.9950
	2	80.8409	0.0752	85.5560	0.0337	0.9905	0.9910
	3	98.1809	0.1017	120.6724	0.0499	0.9853	0.9852
	4	134.0759	0.1296	152.581	0.0745	0.9794	0.9794
	5	137.7988	0.1569	159.4555	0.1006	0.9731	0.9732

Resc-log: Rescaled-logarithmic transformed flow series; raw data: Original flow data; Standardised: the demeaned original flow series divided by standard deviation; Resc: rescaled.

Table 2: ANN forecast performance during Validation

Transform	Lead	MAE	MAPE	RMSE	MSRE	CE	R^2
Resc-log							
	1	70.2001	0.1509	72.7876	0.0703	0.9561	0.9566
	2	79.3483	0.1642	80.4672	0.0872	0.9472	0.9476
	3	105.0832	0.1799	129.1394	0.1027	0.9395	0.9398
	4	122.5459	0.1935	131.8075	0.1134	0.9319	0.9324
	5	130.8713	0.2075	145.3140	0.1264	0.8187	0.9270
Standardised							
	1	45.4193	0.1666	68.5223	0.1063	0.9572	0.9573
	2	53.3808	0.1737	80.0700	0.1131	0.9493	0.9495
	3	98.2346	0.1830	117.8108	0.1182	0.8408	0.9410
	4	109.9609	0.1921	120.2051	0.1199	0.8405	0.9324
	5	115.2344	0.2015	125.5316	0.1186	0.8401	0.9239
Resc-raw data							
	1	75.0910	0.1665	80.8146	0.1044	0.9552	0.9557
	2	83.1199	0.1747	86.8868	0.1147	0.9481	0.9483
	3	107.5458	0.1878	130.0229	0.1236	0.8138	0.9383
	4	136.5948	0.1997	157.8905	0.1287	0.8115	0.9284
	5	140.0822	0.2117	168.1761	0.1357	0.8037	0.9199

Resc-log: Rescaled-logarithmic transformed flow series; raw data: Original flow data; Standardised: the demeaned original flow series divided by standard deviation; Resc: rescaled.

Table 3: ANN model performance in terms of extreme events (Max. and Min. flow)

		Training		Validation	
Transform	Lead	$R_{max}(\%)$	$R_{min}(\%)$	$R_{max}(\%)$	$R_{min}(\%)$
Resc-log					
	1	98.03	137.47	99.40	114.67
	2	98.78	148.64	98.87	118.47
	3	99.46	149.28	98.56	118.55
	4	100.02	145.45	98.29	119.96
	5	100.48	136.23	97.80	124.18
Standardised					
	1	98.27	94.87	100.95	105.91
	2	97.63	94.30	101.47	91.75
	3	97.80	67.16	101.61	82.02
	4	98.22	56.04	101.43	62.84
	5	98.10	32.59	101.25	45.05
Resc-raw data					
	1	98.45	87.68	98.22	82.05
	2	97.92	49.92	98.18	68.43
	3	98.23	22.93	98.00	45.92
	4	98.65	22.58	97.59	12.62
	5	98.96	20.60	97.18	95.83

Though the overall accuracy of the model in terms of the statistical parameters CE, R^2, and RMSE (Tables 1 and 2) are seemingly good, they do not really reveal the distribution of the forecast errors since there are global statistics. The values of MSRE and MAE in the validation period increase appreciably with the lead time (in days) indicating the distortion in the distribution of the forecast errors. This aspect in the forecast behaviour during the validation period is paramount since from a practical stand point they serve to assess and quantify the forecast errors of the ANN forecast model. Table 3 succinctly illustrates this distortion with regards to forecast of extreme events. It does provide an intuitive outlook on ANN model prediction when a univariate time series is used. In general, Table 3 also showed that in terms of R_{min} and R_{max}, it is obvious that ANN model forecasts high flows much better that low flows. This underscores the need for the inclusion of exogenous input (precipitation) in the network input variables. Peaks corresponding to larger values of discharge are always generated by rainfall that are heavy and of long duration and intermediate peaks, on the other hand, are caused by heavy rainfall of short duration; thus the non-inclusion of precipitation in the input data set which might mitigate this phenomenon probably explains the distortion. Comprehensively though, comparative hydrographs (Fig. 5) of the observed and forecasted streamflow for one-day-ahead depicts the goodness-of-fit for the network trained using the ANN structure as determined in this case.

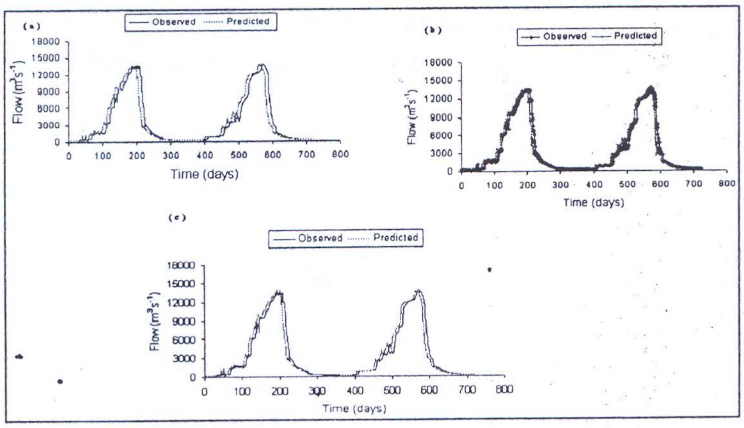

Fig. 5: ANN model 1-day-ahead forecasts: (a) Rescaled logarithmic-transformed flow series, (b) Rescaled raw flow series, (c) Standardised raw flow data

It is paramount not to only evaluate model forecast performance on the basis of statistical parameters, but to also consider the impact data pre-processing might have on ANN model forecasts (Wang, 2006). It is recognised that data pre-processing can have a significant effect on model performance (e.g. Maier and Dandy, 2000). It is commonly considered that, because the outputs of some transfer functions are bounded, the outputs of an MLP ANN will be in the interval [0, 1] or [-1, 1] depending on the transfer function used in the neurons. Reports in literature suggest using smaller intervals for streamflow modelling as [0.1, 0.85] (Shamseldin, 1997), and [0.1, 0.9] (Abrahart and See, 2000), so that extreme (low and high) flow events occurring outside the range of the calibration data may be accommodated. However, the advantage of rescaling the data into a small interval is not supported as illustrated in Table 3. In this case, the general performance of the MLP-ANN with standardisation pre-processing is much better in the overall; especially for low and high flows (i.e., extreme events in terms of minimum and maximum flows) during validation stage; this result is in agreement with similar findings by Wang (2006). This could be explained against the backdrop of the behaviour of the transfer function. For instance, to rescale the input data to [-1, 1] would limit the output range of the $\tan sig(x)$ function approximately to [-0.7616, 0.7616] (Wang, 2006). Similarly, to rescale the input range to [-0.9, 0.9] would further shrink the output range approximately to [-0.7163, 0.7163] (Wang, 2006). Both 0.7616 and 0.7163 are still far away from the extreme limits of the $\tan sig(x)$ function: such a small output data range will make the output less sensitive to the change of the weights between the hidden layer and output layer, and will therefore possibly make the training process more difficult. In addition, in line with Wang (2006), since the neurons in an ANN structure are combined linearly with a lot of weights, any rescaling of the input vector can be offset the more, as corresponding weights and biases are changed.

CONCLUSIONS

It is evident from the ANN model forecast performance that the application of the knowledge of evolution of a dynamical system in a multi-dimensional state space is a veritable way in determining the size of input in a neural network model, especially with a univariate time series as it does not involve the analysis of extensive model sensitivity to the input data. The ANN model forecast performance showed that reliable short term forecasts, 5 day - ahead can be made for the daily streamflow series, using multiple-output regime. However, on the general question of the suitability of ANN model application for streamflow forecasting as applied in this study (i.e., daily streamflow), though the neural network could simulate the different attributes of the flow hydrograph, its relative forecast performance of high flows is robustly better than the case of low flows; it grossly under predicts and over predicts low flows depending on the particular network input data pre-processing schema.

Analysis of the influence of different data pre-processing schema namely, rescaling, and standardisation on the ANN model forecast performance brought to the fore the associated encumbrances that the modeller might face; especially, in drawing up an objective conclusion if proper data processing was not done. Concisely, it is evident that, for MLP networks with a tan-sigmoid transfer function, standardising the data by subtracting the mean and dividing by the standard deviation is better than rescaling the data to a small interval of particular range.

REFERENCES

Abrahart, J. R., and See, L. (2000). Comparing Neural Network and Autoregressive Moving Average techniques for provision of continuous river flow forecasts in two contrasting catchments. *J. Hydro. Processes*, 14: 2157-2172.

Bishop, C. M. (1994). Neural networks and their applications. *Rev. Sci. In Strum,* 65: 1803-1832.

Box, G. E. P., and Jenkins, G. M. (1976). Time Series Analysis Forecasting and Control. Holden-Day Press, San Francisco, California, USA.

Delleur, J. W., Tao, P. C, and Kavvas, M. L. (1976). An evaluation of the practicality and complexity of some rainfall and runoff time series models. *Water Resour. Res.* 12(5): 953-970.

Elena, T., and Armando. B. (2002). Flood forecasting using artificial neural networks in Black-box and Conceptual rainfall-runoff modelling, www.iemss.org /iemss2002/proceedings/pdf/.volume%20due/370_toth.pdf.

Haykin, S. (1994). Neural Network, a comprehensive foundation, IEEE, New York.

Hecht-Nielsen, R. (1991). Neurocomputing, Addison-Wesley, New York.

Hurst, H. (1951). Long-term storage capacity of reservoirs. *Trans. Am. Soc. Civ. Eng.,* 116: 770-799.

Kennel, M.B., Brown, R., and Abarbanel, H. D. (1992). Determining embedding dimension for Phase-space reconstruction using geometrical construction. *Phys. Rev.* A 45:: 3403-3411.

Maier, H. R., Dandy, G. C. (2000). Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environ. Model Softw.* 15: 101-123.

Matalas, N. C. (1967). Mathematical assessment of symmetric hydrology; *Water Resour. Press.* 3 (4): 937-945.

Ozgur, K. (2004). River Flow Modelling Using Artificial Neural Networks; *J. Hydrol. Eng.* 9(1): 60-63.

Packard, N. H., Crutchfield, J. P., Farmer, J. D., and Shaw, R. S. (1980). Geometry from a time series. *Phys. Rev. Lett.*, 45: 712-716.

Shamseldin, A. Y. (1997). Application of a neural network technique to rainfall-runoff modelling. *J. Hydrol.* 199: 272-294.

Singh, V. P. (1988). Hydrologic Systems: Rainfall-Runoff Modelling, Vol. 1, Prentice-Hall, Englewood Cliffs, N. Jersey.

Takens, F. (1981). Detecting strange attractors in turbulence in Lecture Notes in Mathematics, edited by D.A. Rand and L.S. Young, Vol. 898, pp: 366-381, Springer-Verla, New York.

Wang, W. (2006). Stochasticity, nonlinearity and forecasting of streamflow processes. Deft University Press, Amsterdam – Netherlands, ISBN 1-58603-621-1.