Journal of Urban and Environmental Engineering, v.11, n.1, p.119-130

ISSN 1982-3932
doi: 10.4090/juee.2017.v11n1.119130

Journal of Urban and Environmental Engineering

www.journal-uee.org

DENOISING- JITTERING DATA PRE-PROCESSING TECHNIQUE TO IMPROVE ARTIFICIAL INTELLIGENCE BASED RAINFALL- RUNOFF MODELING

Afshin Partovian[1], Vahid Nourani[2,3*] and Mohammad Taghi. Aalami[2]

[1]*Faculty of Civil Engineering, Boukan Branch, Islamic Azad University, Boukan, Iran, Email: a. partoviyan@iau-boukan.ac.ir*
[2] *Department of Water Resources Engineering, Faculty of Civil Engineering, University of Tabriz, Tabriz, Iran*
[3] *Department of Civil Engineering, Near East University, P.O. Box: 99138, Nicosia, North Cyprus, Mersin 10*

Received 22 August 2016; received in revised form 23 May 2017; accepted 05 November 2017

Abstract: Successful modeling of hydro-environmental processes widely relies on quantity and quality of accessible data and noisy data might effect on the functioning of the modeling. On the other hand in training phase of any Artificial Intelligence (AI) based model, each training data set is usually a limited sample of possible patterns of the process and hence, might not show the behavior of whole population. Accordingly in the present article first, wavelet-based denoising method was used in order to smooth hydrological time series and then small normally distributed noises with the mean of zero and various standard deviations were generated and added to the smoothed time series to form different denoised-jittered training data sets, for Artificial Neural Network (ANN) and Adaptive Neuro-Fuzzy Inference System (ANFIS) modeling of daily rainfall – runoff process of the Oconee River watershed located in USA. To evaluate the modeling performance, the outcomes were compared with the results of multi linear regression (MLR) and Auto Regressive Integrated Moving Average (ARIMA) models. Comparing the achieved results via the trained ANN and ANFIS models using denoised-jittered data showed that the proposed data processing approach which serves both denoising and jittering techniques could improve performance of the ANN and ANFIS based rainfall-runoff modeling of the Oconee River Watershed up to 13% and 11% in the verification phase.

Keywords: Rainfall-Runoff modeling ; ANN; ANFIS; Wavelet denoising ; Jittered data; Oconee River watershed

* Correspondence to: Vahid NOURANI, Tel.: +98 413 339 2409; fax: +98 413 334 4287.
E-mail: nourani@tabrizu.ac.ir,vahid.nourani@neu.edu.tr

INTRODUCTION

Nowadays water resources management is vitally important task and optimum planning of irrigation projects, development and exploitation of water resources especially during drought and flood events will be strictly dependent to the accuracy of the used rainfall-runoff modeling tool. Therefore different models have been already developed and employed for modeling rainfall-runoff process of the watersheds. Owing to the large number of vague physical parameters in the hydrological processes**,** black box (lumped) models are mostly applied, since they may have some benefits over fully distributed models (Nourani and Mano 2007). For instance, successful hydro-environmental applications of auto regressive integrated moving average (ARIMA) and multi linear regression (MLR) models have been already reported by several researchers (e.g. see Wang et al. 2015; Salas et al. 1980; Zhang et al. 2011).

Although these models are linear and may sometimes not be accurate due to their incapability to deal with non-stationary and non-linearity, they are still applied in practice because they can be easily used to compare and evaluate the effectiveness of novel methods. As such black box models, Artificial Neural Network (ANN) has recently indicated great ability for rainfall-runoff modeling (e.g., Nourani and Saeidifarzad 2016; Nourani et al. 2014; Chau etal. 2015; Abrahart et al. 2012; ASCE 2000).

The models and data used to simulate the rainfall-runoff process usually involve uncertainty, which to overcome such uncertainty, a combination of ANN and fuzzy system may be considered which benefits both ANN and fuzzy concepts via a single framework of so called ANFIS (Adaptive Neuro- Fuzzy Inference System) model. The merit of ANFIS for rainfall-runoff modeling has been reported by a few studies (e.g., Chang et al. 2017; Chang et al. 2016). The efficiency of artificial intelligence (AI) techniques like ANFIS and ANN may be altered if noisy time series and data are used as inputs (Sang et al. 2009). Since the performance of any data-driven model is sensitive to the quality of the used data, different methods have been proposed for data denoising purpose, e.g. Wiener filter and Kalman filter (Kalman 1960; Wiener 1949), which are appropriate for linear systems but sometimes inappropriate for non-linear hydro-environmental processes. When classic methods for modeling hydrological time series do not meet the practical needs based on their limitations exposing to non-stationary characteristics and multi time scales, wavelet threshold denoising (WTD) method proposed by Donoho (1995) can be used as a reliable alternative. In hydrological practices, the WTD method is known more influential than conventional methods since it can contribute the illumination of the localized characteristics of non-stationary time series both in temporal and frequency domains (Jansen 2006). There have been only a few researches focusing on the use of wavelet based data denoising approaches in hydrological modeling (e.g., see Nejad and Nourani 2012; Nourani et al. 2014). On the other hand in training phase of an AI model, the training data set includes a limited sample of all data, so a set of selected data can not reflect all possible patterns of the process (Zhang 2007). Jittered data for calibration of an AI model can enlarge the sample size of training data set by its supplementation using extra generated data which are similar to, but different from the original observed data.

This can make it possible that the data are appeared more smoothly to an AI model and therefore enhance the model capability to learn the real patterns involved in the process (Zhang 2007). Furthermore, it can prevent over fitting of model by supplying extra constraints, and imposing the jittered data into the training patterns can lead to improvements of the AI modeling. Therefore, the jittered data obtained by the noise injection method can be a useful pre-processing technique for AI-based model building (Zur et al. 2004; Zhang 2007). The selection of a suitable noise size to be injected to the original time series to create jittering data has not been well described in technical literature.

Obviously the appropriate variance of noise should be a problem reliant as distinct time series may have different inherent noise levels. Consideration of high levels of noise can deform the underlying pattern while small noises might not have sufficient influence on the jittering performance.

Most of the researches regarding the application of jittering data concentrate on classification problems and financial time series analysis and there is not any research in hydro-environmental modeling. The introduced data pre-processing approach in the present study is used for the first time in modeling of time series and especially in modeling of hydrological processes. Furthermore the impacts of denoising (smoothing) and noise injection (jittering) have been simultaneously investigated neither in hydrology nor in any other engineering fields. In the previous study (i.e., Nejad and Nourani, 2012), only data de-noising has been employed as data-preprocessing approach but here, both de-noising and jittering have been applied and examined via the modeling framework, as a new data-processing approach.

Actually such data jittering can be used to improve the quantity of the training samples.Thus it is necessary to produce more researches on this filed and providing suitable solution to model hydro-environmental phenomena which is addressed in this article.

MATERIAL AND METHODS

The proposed hybrid model

In the proposed method in this study first by applying wavelet based denoising approach on raw data, the outliers and systematic noises of time series are identified and shrunk to produce smooth hydrological time series. The magnitude of the shrinkage is controlled according to a threshold value. Then to have several time series with similar pattern to the original smoothed time series, jittered time series are generated by adding normally distributed noise time series with specified standard deviations to the smoothed time series of the hydrologic parameters. Finally, the produced jittered time series are imposed to the ANN or ANFIS forecasting model. In **Fig. 1**, schematic diagram of the proposed method is shown.

Wavelet denoising procedure

Wavelet data denoising method based on the thresholding to obtain denoised signals has been introduced by Donoho (1995). In this method, first a signal is decomposed into different sub-signals at different resolutions through controlling scaling and shifting coefficients by the wavelet transform. By this way, reliable localization properties which are caught in both time and frequency domains can be provided. Second a thresholding rule is applied on the sub-signals. The basic factors that must be respected in this method include: selection of a mother wavelet, decomposition level, thresholding rule and accurate estimation of threshold rule. For a mother wavelet ψ (t), the wavelet basis function can be considered as follow (Nourani et al. 2014):

$$\psi_{a,b}(t) = \frac{1}{\sqrt{|a|}}\psi\left(\frac{t-b}{a}\right) \qquad (a,b)\in R;\ a \neq 0 \qquad (1)$$

In this equation a,b and R indicate respectively scale and shift factors and the real number domain and $\psi_{a,b}(t)$ is the successive wavelet. The wavelet transform of a signal $f(t) \in L^2(R)$ can be written as (Nourani et al. 2014):

$$w_f(a,b) = \frac{1}{\sqrt{a}}\int_{-\infty}^{+\infty} f(t)\,\overline{\psi\left(\frac{t-b}{a}\right)}\,dt \qquad (2)$$

Which $\overline{\Psi(t)}$ is complex conjugate of ψ(t). As it is clear from **Eq. (2)**, the wavelet transform of a time series like *f(t)* decomposes it under various resolution levels. By applying successive wavelet transform, the main signal of *f(t)* is reconstructed using inverse transform using the wavelet coefficients of w_f (a,b), as (Sang et al. 2009):

$$f(t) = \left[\int_{-\infty}^{+\infty}\frac{|\hat{\psi}(\omega)|^2}{|\omega|}\,d\omega\right]^{-1}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\frac{1}{a^2}w_f(a,b)\,\psi_{a,b}(t)\,da\,db \qquad (3)$$

The wavelet based thresholding technique as a widely used data denoising approach is conducted through three steps as (Donoho 1995):

a) First a proper mother wavelet and a reasonable resolution level of N are chosen for the specified period of the study process to decompose the main time series to an approximation sub-series at level N and N detailed sub-series via wavelet transform .

b) In the second step the absolute values of the detailed sub-series in resolution level of i $d_i(t)$ (i = 1, 2,..., N) which are less than a specified threshold of T, will be changed to zero, but if the values of detailed sub-series at the same resolution level exceed this specified threshold, their difference with the threshold value are considered as the modified values of detailed sub-series. Which this thresholding procedure can be mathematically shown by (Donoho 1995):

$$\hat{d}_i(t) = \begin{cases} sgn(d_i(t))(|d_i(t)| - T) & |d_i(t)| > T \\ 0 & |d_i(t)| \le T \end{cases} \qquad (4)$$

where i refers to i th resolution level. **Eq. (4)** applies the thresholding at all resolution levels on detailed sub-series, but the approximation sub-series is not included in this thresholding procedure.

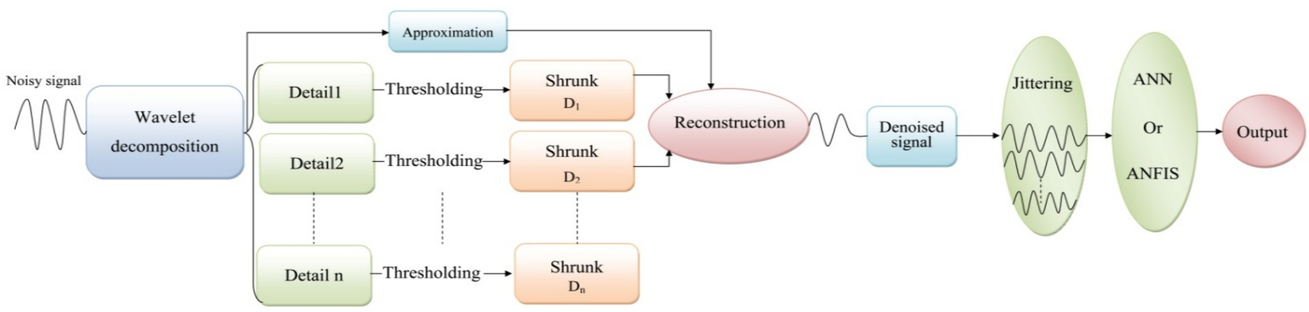

Figure. 1 Schematic diagram of the proposed model

Donoho and Johnstone (1995) proposed a formula to determine a general optimal threshold value for signals which are included white Gaussian noises as:

$$T = \sigma\sqrt{2\log_e(n)} \quad (5)$$

where the number of samples in the noisy signal is n and σ is the standard deviation of noises which may be obtained as (Donoho and Johnstone 1995):

$$\sigma = \left[\frac{\mathrm{median}(|d_t(t)|)}{0.6745}\right] \quad (6)$$

So that $|d_t(t)|$ represent detailed wavelet coefficient of main time series of first level.
c) At the third step, the denoised (smoothed) sub-series can be reconstructed by modified detailed sub-series at all resolution levels and approximation sub-series at resolution level N through the inverse wavelet transform (**Eq. 3**).

As it has been shown in Figure 1, the de-noising procedure is not applied on the approximation sub-series in which this sub-series includes trend and effective large scale fluctuations of the process and are not affected by the de-noising of smaller scales sub-series. Clearly by changing the decomposition level (*N*), the degree of such large scale fluctuations included in the unchanged approximation sub-series are changed. In this study general (universal) thresholding method **(Eqs. 4–5)** was applied for de-noising procedure in which in this method the threshold is applied to all detail sub-series, however there are some other sophistical methods which as a level-based thresholding algorithms apply threshold only to some of detail sub-series (some of scale levels) rather to all scale levels (e.g. see, Nourani et al. 2014).

Jittered data generation

The process of generating random data with a specified statistical distribution usually consists of two steps. First random data with uniform distribution are generated, then these random numbers with uniform distribution are used to produce random numbers with an arbitrary distribution. Some methods such as reverse conversion method could be used in order to generate random numbers with such an arbitrary distribution. In this approach when random variable of x has a cumulative distribution of F(x), in this case u=F(x) will have uniform distribution of u(0,1) and vice versa if u~u (0,1), in this case, $x = F^{-1}(u)$ will have cumulative distribution function of F and therefore, for generating random variable of y with distribution function of G we will have (Bowker & Lieberman 1972):

$$y = \underbrace{G^{-1}(\overbrace{F(x)}^{\text{uniform distribution}})}_{\text{includes G cumulative distribution}} \quad (7)$$

Random numbers based on different distributions could be generated by softwares. In this study, NORMRND toolbox of MATLAB was used to produce normally distributed random time series of jittered noises with mean of zero and several small standard deviations consistent with the original time series of the hydrological parameters.

Artificial Neural Networks (ANNs)

Artificial neural networks (ANNs) are widely used for modeling and prediction of hydro- environmental processes. In this regard, feed forward ANN trained by the back-propagation algorithm including one input, one hidden and one output layers are more suitable option in compared to other ANN types in most engineering disciplines (ASCE 2000; Hornik 1988). This network has great ability to learn involved patterns within non-linear systems through only three layers. Neurons (nodes) in each layer are connected to all nodes in previous layer. Due to the feed forward framework, the path of signals is in forward direction and the outputs of input layer, create the input vector for hidden layer and similarly the outputs of the hidden layer make inputs for the output layer. The output value of a feed forward neural network with three layers can be obtained through the following equation (Kim and Valdes 2003):

$$\hat{y}_k = f_0\left[\sum_{j=1}^{M_N} G_{kj} \cdot f_h\left(\sum_{i=1}^{N_N} G_{ji} x_i + G_{jo}\right) + G_{ko}\right] \quad (8)$$

where **Eq. (8)** applies weight of G_{ji} on a node in hidden layer which connects *i*th node of the input layer to the *j* th node of the hidden layer and bias of G_{jo} on the *j* th hidden node. f_h is the activation function for all nodes of hidden layer, weight G_{kj} is applied on the output layer to the path where connects the *j* th node in hidden layer to the *k* th node of the output layer, G_{k0} is the bias of the *k* th output node, f_0 is the activation function for the output node, x_i denotes to the input value of i th node in input layer and $\hat{y}_k$, y show respectively calculated and observed values for target (output) node. Finally, N_N and M_N indicate respectively the number of input and hidden layers' nodes. The different bias and weights applied on the nodes of hidden and output layers are tuned through the calibration phase of modeling.

Adaptive Neuro-Fuzzy Inference System (ANFIS)

ANFIS as a hybrid model is formed of a fuzzy system combined with a feed forward network (Jang et al.

1997). The fuzzy system is created according to the logic rules of conditions and outputs which may not be analyzed using conventional probability concept. A fuzzy system includes five units, fuzzification, rules, setting, inference engine and defuzzification. Among various fuzzy inference engines for fuzzy operation purpose, sugeno and mamdani schemes (Jang et al. 1997) are most commonly used engines in which the sugeno engine was used in this study. To show typical mechanism of ANFIS to create target (output) function of f, for instance with two input vectors of x and y, the first order sugeno inference engine may be applied to, two fuzzy if–then rules as written (Aqil et al. 2007):

Rule (1): If μ(x) is A_1 and μ(y) is B_1;
then $f_1 = p_1x+q_1y+r_1$
Rule (2): If μ(x) is A_2 and μ(y) is B_2;
then $f_2= p_2x+q_2y+r_2$

in which A_1, A_2 and B_1, B_2 are respectively the membership functions of inputs x and y. p_1, q_1, r_1 and p_2, q_2, r_2 are coefficient of target function. The ANFIS performs modeling through five layers. The first layer contains input neurons which provide membership degree of μ for each input value. Considering Gaussian membership function for the i th neuron, its output could be expressed by following Eq.(Jang and Sun 1995):

$$Q_i^1 = \mu_{A_i}(x) = \frac{1}{1 + ((x - c_i)/a_i)^{2b_i}} \tag{9}$$

In which Q_i^k is output of i th neuron in k th layer (k=1 for the first layer) and {a_i, b_i, c_i} are tunable premise coefficient.
In the second layer, the output of each neuron will be the product of entering values to that neurons as (Jang and Sun 1995):

$$Q_i^2 = w_i = \mu_{A_i}(x) . \mu_{B_i}(y) i = 1,2 \tag{10}$$

Each neuron in the third layer, computes relative weight as (Jang and Sun 1995):

$$Q_i^3 = \bar{w}_i = \frac{w_i}{w_1 + w_2} \qquad i = 1,2 \tag{11}$$

The neuron i in the fourth layer computes ration of each rule with regard to the output of model in the following form (Jang and Sun 1995):

$$Q_i^4 = \bar{w}_i(p_i x + q_i y + r_i) = \bar{w}_i f_i \tag{12}$$

The final output of ANFIS model is computed by a neuron in the fifth layer as (Jang and Sun 1995):

$$Q_i^5 = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{13}$$

The ANFIS is calibrated by a combined training approach to optimize both premise parameters set {a_i, b_i, c_i} and consequent parameters set {p_i, q_i, r_i}.This hybrid training algorithm includes both the gradient descent and least squared methods (Aqil et al. 2007).In the forward path of training, the outputs of neuron proceed ahead until layer four and the least squared method is used to estimate the consequent parameters. Thereafter, the errors are propagated in the backward path to update the premise parameters by using the gradient descent method. Jang and sun (1995) and Jang et al.(1997) may be referred for details regarding ANFIS model and its training algorithm.

Study area and data

The data used in this paper are from Oconee River watershed, located in southeastern United States at Georgia State, Baldwin County, Milledgeville station (Latitude 33°05'22, Longitude 83°12'56). The drainage area is 7637 square kilometers **Fig. 2**. The daily rainfall and runoff data for 15 years (from 2000 to 2015) which were used in this research are available at the United States Geological Survey website (USGS) (http: //water.usgs.gov/data/). The statistical parameters of the rainfall and runoff data such as the mean, standard deviation, maximum and minimum values (i.e., Xmean, Sd, Xmax and Xmin, respectively) are given in **Table 1**. Due to the training and verification goals, data set was divided into two parts. The first division as 70% of total data included the training set and the rest 30% data set was used for the verification purpose.

Table 1. Statistics of time series for calibration, verification and all

Statistical parameters	All data		Training data		Verifying data	
	Runoff(m^3/S)	Rainfall (mm)	Runoff(m^3/S)	Rainfall (mm)	Runoff(m^3/S)	Rainfall (mm)
Xmean	66.21	2.15	72.44	2.50	51.69	1.33
Xmax	1500.79	103.88	1500.79	103.88	940.12	57.15
Xmin	6.22	0	6.22	0	6.79	0
Sd	101.98	7.74	110.61	8.47	76.29	5.57

Efficiency criteria

In this study, to evaluate the accuracy of the proposed methodology and to compare different models, coefficient of determination (DC) and root mean square error (RMSE) were employed as **Eqs (14)** and (15), respectively (Nourani et al. 2009).

$$DC = 1 - \frac{\sum_{i=1}^{n}(O_{obs_i} - O_{com_i})^2}{\sum_{i=1}^{n}(O_{obs_i} - \bar{O}_{obs})^2} \quad (14)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(O_{obs_i} - O_{com_i})^2}{n}} \quad (15)$$

where O_{obsi}, O_{comi}, $\bar{O}_{obs}$, and n are observed data, computed values, mean of observed data and number of observations, respectively. As much as RMSE and DC values are closer to zero and one, respectively accuracy of the model will be higher.

The generated noise time series may include negative quantities, therefore in order to prevent the producing of negative hydrological data and to scale the data, the time series were normalized between 0.1 and 0.9 by the **Eq. (16)** (Rajurkar et al. 2002): with regard to **Eq. (16)**, X_i, X_{min}, X_{max} and N_i are observed variables, minimum and maximum values and the normalized variable, respectively.

$$N_i = 0.8\left[\frac{(X_i - X_{min})}{X_{max} - X_{min}}\right] + 0.1 \quad (16)$$

RESULTS AND DISCUSSION

As the first step, for modeling of rainfall – runoff process, the raw data prior to apply the data pre-processing approaches were entered to the FFNN model which is the most common ANN for hydrological modeling and to ANFIS model, respectively. The used algorithm for training the FFNN was back propagation with Levenberg– Marquardt training scheme and 3 to 10 neurons were examined for the hidden layer. No great improvement in model performance was found when the number of hidden neurons was increased from a threshold. In ANFIS modeling, Sugeno fuzzy inference system was considered and trained using hybrid optimization algorithm. The ANFIS model contains a number of rules with some membership function parameters. In the current study Gaussian membership function with 3 membership function was found to be appropriate for simulation of rainfall- runoff process via ANFIS. Five different combinations of input data for runoff prediction considered as follow:

Comb. 1: R_t, Q_t
Comb. 2: R_t, Q_{t-1}, Q_t
Comb. 3 : R_t,R_{t-1}, Q_{t-1},Q_t
Comb. 4 :R_t ,Q_{t-2},Q_{t-1},Q_t
Comb. 5 : R_{t-1}, R_t, Q_{t-2}, Q_{t-1}, Q_t

In all cases the output was the discharge at the next time step Q_{t+1} where R_t presents rainfall value at time step t. The results of ANN and ANFIS models with original noisy data are shown in **Tables 2–3**, respectively. It should be noticed that only the results of the best structures have been presented in the tables and numbering of a-b-c in neural network structure denotes to number of input layer, hidden layer and outputlayer neurons.

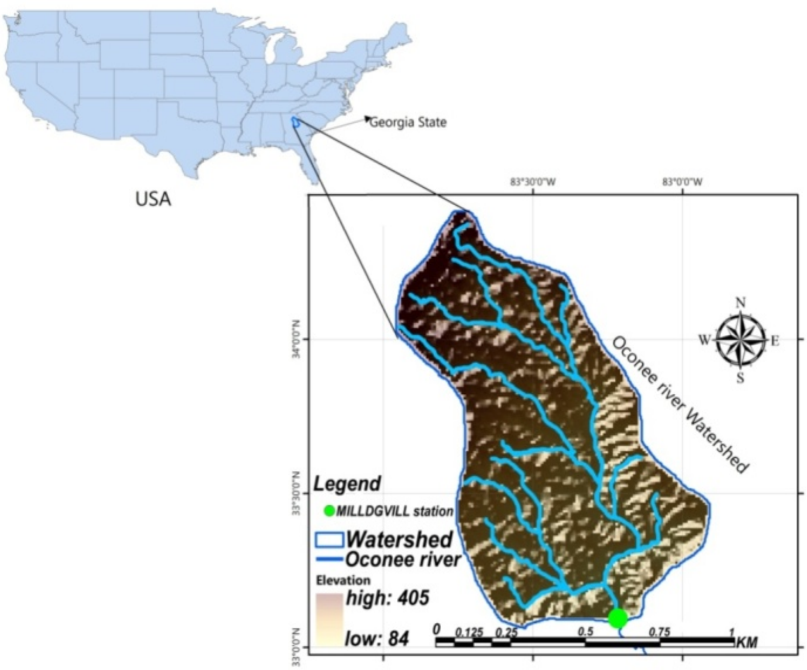

Figure. 2 Oconee River watershed

Table 2. Results and structures of ANN model for the different input combinations

Input Variables[a] (rainfall and runoff)	Network Structure	Epoch	RMSE (Normalized)		DC	
			Calibration	Verification	Calibration	Verification
Comb.1	(2-7-1)	30	0.038	0.028	0.58	0.50
Comb.2	(3-3-1)	90	0.0314	0.0227	0.718	0.689
Comb.3	(4-8-1)	50	0.0292	0.0205	0.867	0.753
Comb.4	(4-8-1)	80	0.0289	0.0184	0.890	0.815
Comb.5	(5-4-1)	40	0.033	0.0244	0.673	0.641

Table 3. Results of ANFIS model for the different input combinations

Input Variables	MF	RMSE (Normalized)		DC	
		Calibration	Verification	Calibration	Verification
Comb.1	Gaussian	0.0272	0.0181	0.753	0.737
Comb.2	Gaussian	0.0216	0.0166	0.90	0.78
Comb.3	Gaussian	0.0233	0.0180	0.845	0.743
Comb.4	Gaussian	0.0246	0.020	0.792	0.684
Comb.5	Gaussian	0.0261	0.021	0.760	0.658

According to the obtained evaluation criteria, it is clear that inputs Comb. (4) and Comb. (2) could lead to better performance in ANN and ANFIS modeling, respectively and thereafter used for ANN and ANFIS modeling. The difference between these two combinations is in $Q_{t-2,}$ that ANFIS due to handling uncertainties, could lead to similar results to ANN but using one less input in comparison to ANN. Actually, although the training scheme is similar in both methods, in ANFIS method, fuzzy representation of data instead of crispy representation of data which is used by ANN, may lead to different outcomes for these two methods. In the next step of modeling, in order to eliminate the outliers and systematic large noises of the observed data, wavelet-based denoising approach was applied on raw data. Since the type of used mother wavelet and decomposition level can alter denoising performance, wavelet denoising was performed and compared using Daubechies mother wavelets (Haar or Db1, Db2, Db3 and Db4) at two different resolution levels of 8 and 9 (Walker 1999). The reason for choosing these two resolution levels is that one year includes 365 days which is between two dyadic modes of 2^8 and 2^9, therefore these two possibilities focus on annual period intensity. The denoising procedure of hydrological time series was performed using different mother wavelets and decomposition levels of 8 and 9 and specified threshold obtained through equation 5, then the models were trained using such smoothed input combination set determined in sensitivity analysis step (Comb. 4 and Comb. 2 in ANN and ANFIS, respectively). The results of ANN and ANFIS modeling using denoised (smoothed) input data have been summarized in **Tables 4–5**, respectively.

Table 4. Results and structures of ANN modeling using denoised data (using input Comb.4)

Mother Wavelet	D.L[a]	Threshold (Normalized)	Network Structure	Epoch	RMSE (Normalized)		DC	
					Calibration	Verification	Calibration	Verification
Haar	8	0.107	4-4-1	100	0.0324	0.0185	0.873	0.814
Db2	9	0.107	4-8-1	60	0.0286	0.0171	0.892	0.835
Db3	8	0.107	4-8-1	80	0.0268	0.0167	0.90	0.841
Db4	9	0.107	4-5-1	130	0.0226	0.0159	0.92	0.853

Table 5. Results of ANFIS modeling using denoised data (using input Comb.2)

Mother Wavelet	Decomposition Level	Threshold (Normalized)	RMSE (Normalized)		DC	
			Calibration	Verification	Calibration	Verification
Haar	9	0.107	0.0205	0.0162	0.901	0.79
Db2	8	0.107	0.0195	0.0158	0.921	0.805
Db3	8	0.107	0.0191	0.0153	0.925	0.823
Db4	8	0.107	0.0193	0.0154	0.923	0.815

As it can be seen in **Tables 2–5**, the obtained results indicate improvement of about 5 % in both ANN and ANFIS modeling in verification phase when using smooth time series as inputs. The results show that both Db4 and Db3 mother wavelets could lead to reliable results. In the third step of modeling, several jittered input time series with similar pattern to the original time series were produced by adding normally distributed generated noises with zero mean and different standard deviations to the smoothed time series of the hydrologic parameters (obtained in second step of modeling). In this manner the time series would have unique and similar trend (approximation) to the original time series but with different stochastic terms represented by the added small generated noises. Therefore via the training phase of AI modeling, the AI model (ANN and ANFIS in this study) could see and learn different stochastic situations of process which in turn this could enhance the performance of modeling in the verification step (for the unseen data). For this purpose, normally distributed noise time series with mean of zero and standard deviations of 0.0001, 0.001, 0.003, 0.005 and 0.01 (normalized value) were generated and injected to the smoothed hydrological time series (obtained in second step of modeling) and the AI modeling was performed by these jittered input time series. After try and error for several models, it was obtained that these five mentioned standard deviations of noise could lead to better performance in modeling and thereafter used for the modeling. In this stage, normally distributed noise time series with mean of zero and standard deviations of 0.0001, 0.001, 0.003, 0.005 and 0.01 (normalized value) were generated and injected to the smoothed hydrological time series obtained of Db4 and Db3 mother wavelets at resolution levels of 9 and 8, respectively in ANN and ANFIS modeling. Then, according to the best input combination set determined in the first step of modeling (Comb. 4 and Comb. 2, respectively in ANN and ANFIS models), different input combinations with appropriate lag were produced. Therefore, the input combinations in ANN modeling were considered as:

Comb. 1: R_t , Q'_{1t},Q_{t-2D},Q_{t-1D},Q_{tD}
Comb. 2: R_t , $Q'_{2t,}$ Q'_{1t},Q_{t-2D},Q_{t-1D},Q_{tD}
Comb. 3 :R_t ,Q'_{3t} ,$Q'_{2t,}$ Q'_{1t},Q_{t-2D},Q_{t-1D},Q_{tD}
Comb. 4 :$R_{t,}$ Q'_{4t} ,Q'_{3t} ,$Q'_{2t,}$ Q'_{1t},Q_{t-2D},Q_{t-1D},Q_{tD}
And for the ANFIS modeling as:
Comb. 1: R_t , Q'_{1t}, ,Q_{t-1D},Q_{tD}
Comb. 2: R_t , $Q'_{2t,}$ Q'_{1t}, Q_{t-1D},Q_{tD}
Comb. 3 :R_t ,Q'_{3t} ,$Q'_{2t,}$ Q'_{1t}, Q_{t-1D},Q_{tD}
Comb. 4 :$R_{t,}$ Q'_{4t} ,Q'_{3t} ,$Q'_{2t,}$ Q'_{1t}, Q_{t-1D},Q_{tD}

where, Q_{tD} represents value of smooth time series at time step t, and Q'_t indicates the denoised-jittered time series.The indexes 1, 2, 3 and 4 indicate different generated noise (with same standard deviation) added to smoothed time series. For instance the original (Q_t) and three samples of jittered time series generated by noises with standard deviation of 0.003 ($Q'_{1,}$ Q'_2 and Q'_3) are depicted in **Fig. 3**. The obtained results of modeling are shown in **Tables 6–7** respectively for ANN and ANFIS models. It should be noticed that for each of noise time series with a specified standard deviation, different time series (up to four) were generated and different combinations (Comb., 1, 2, 3,4) were produced but only the results of the input combination which led to best results have been presented in the Tables.

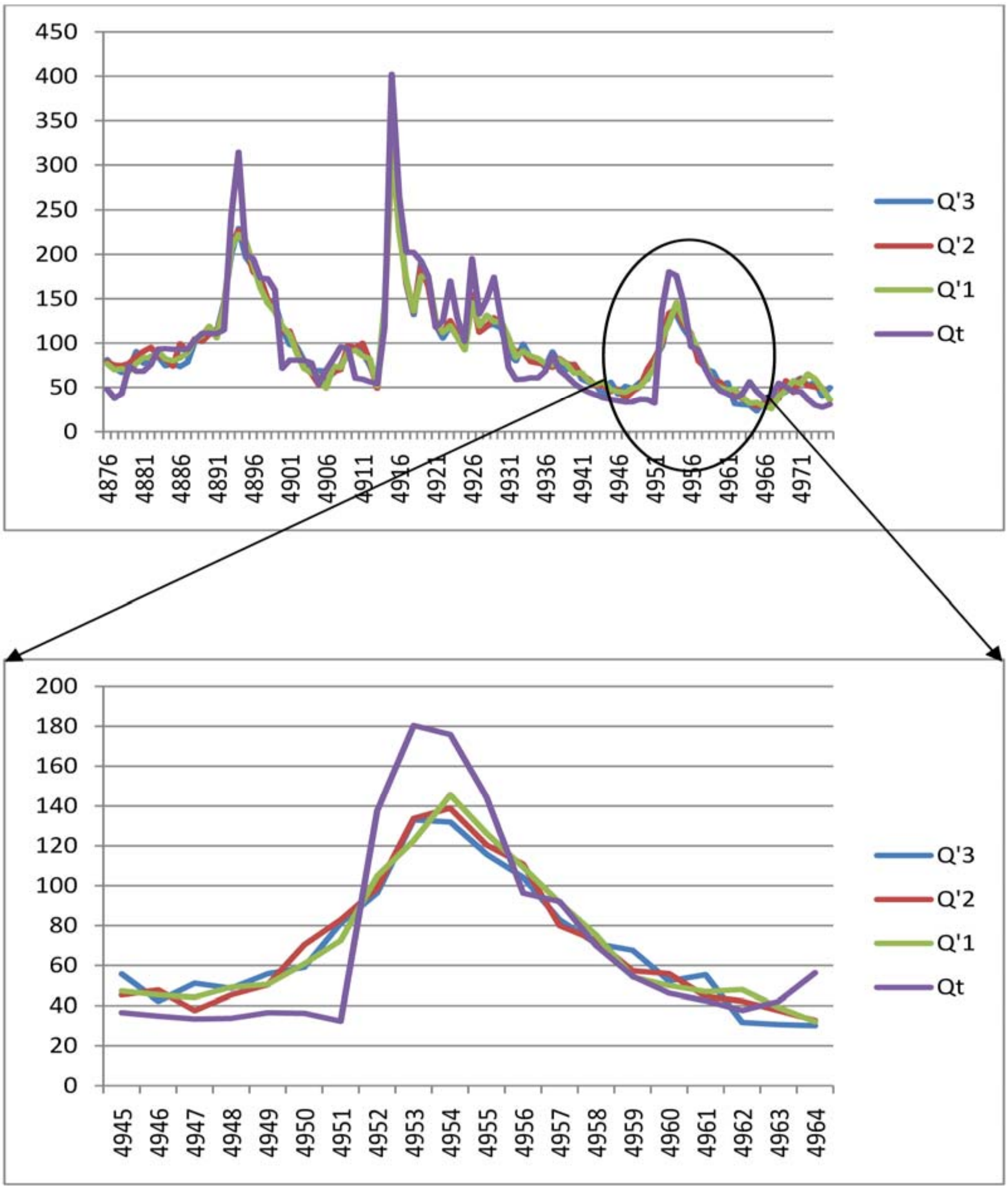

Figure. 3 The original and three samples of generated jittred time series with noise standard deviation of 0.003

Table 6. Results of ANN model using the denoised-jittered data

Standard deviation of noise	Input Structure	Network Structure	Epoch	RMSE (Normalized)		DC	
				Calibration	Verification	Calibration	Verification
0.0001	Comb. 3	(7-8-1)	80	0.0217	0.0151	0.924	0.865
0.001	Comb. 3	(7-8-1)	80	0.020	0.0134	0.935	0.892
0.003	Comb. 3	(7-9-1)	50	0.0163	0.0115	0.95	0.921
0.005	Comb. 3	(7-9-1)	50	0.019	0.0121	0.937	0.911
0.01	Comb. 3	(7-7-1)	60	0.0226	0.0158	0.92	0.855

Table 7. Results of ANFIS model using the denoised-jittered data

Standard deviation of noise	RMSE (Normalized)			DC	
	Input structure	Calibration	Verification	Calibration	Verification
0.0001	Comb. 2	0.0189	0.0149	0.927	0.831
0.001	Comb. 2	0.0175	0.0145	0.941	0.845
0.003	Comb. 2	0.0161	0.0139	0.955	0.866
0.005	Comb. 2	0.018	0.0148	0.936	0.837
0.01	Comb. 2	0.0194	0.0152	0.922	0.825

Based on the efficiency criteria, it is clear that inputs Comb. (3) and Comb. (2) could lead to better performance in ANN and ANFIS models, respectively including generated noise with standard deviation 0f 0.003**,** so that the proposed methodology, in comparison to the situation in which the modeling was done by un-preprocessed data, indicates an improvement of 13 and 11percents in testing phase for ANN and ANFIS models respectively. The scatter plot of optimum ANN and ANFIS models in verification phase are shown in **Figs. 4–5**.

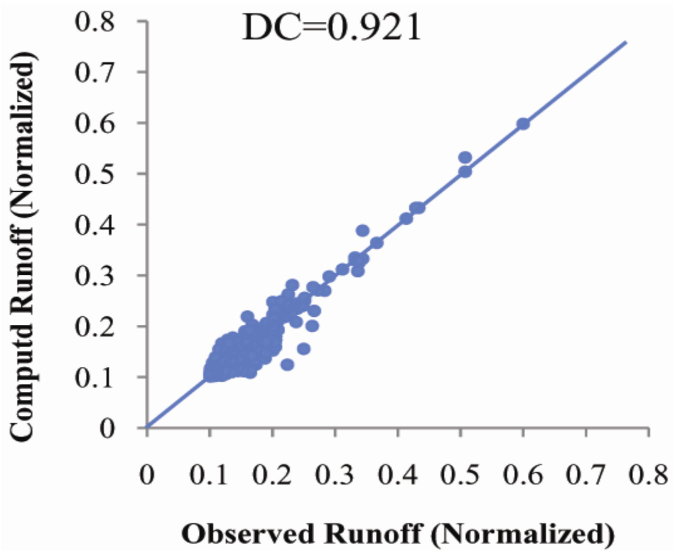

Fig. 4 The verification scatter plot of ANFIS results

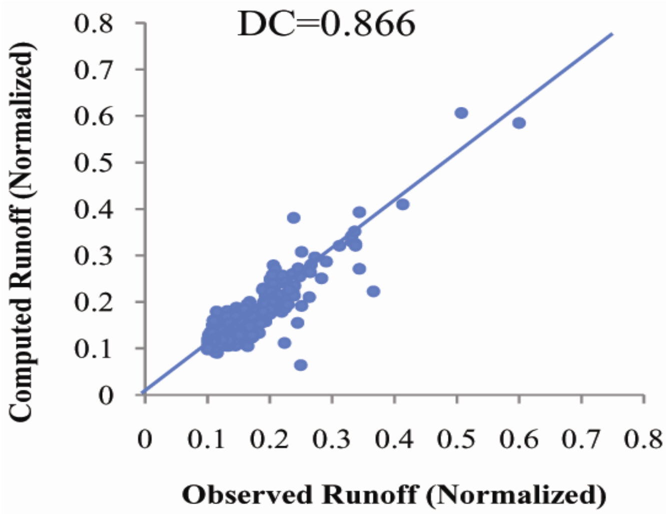

Fig. 5 The verification scatter plot of ANN results

In order to evaluate the ability of proposed modeling, some comparisons with classic linear models of ARIMA (Salas et al., 1980) and MLR (Snedecor and Cochran, 1981) were also conducted in modeling the watershed rainfall–runoff process. ARIMA and MLR modeling have been done by denoised-jittered time series as well. The comparison results are presented in **Table 8**. The results indicate poor outcomes of ARIMA and MLR models with regard to the proposed model. This is due to the limited ability of linear models in modeling non-linearity and non-stationary time series and on the other hand, high dependence of data-driven models to quantity and quality of the used data.

Table 8. Comparison of different rainfall–runoff modeling approaches

Model	RMSE (Normalized)		DC	
	Calibration	Verification	Calibration	Verification
ARIMA	0.0234	0.0195	0.836	0.751
MLR	0.0305	0.0229	0.763	0.685
ANN	0.0163	0.0115	0.95	0.921
ANFIS	0.0161	0.0139	0.955	0.866

CONCLUSIONS

In this study via data pre-processing techniques, the wavelet-based denoised-jittered data were employed in AI-based rainfall-runoff modeling. Accordingly, first it was tried to smooth the hydrological time series by eliminating the outliers and large noises of raw observed time series, which may be due to human or tool measurement error or systematic error. Then different training time series were generated by noise injection to the smoothed time series, and uesd to train ANN and ANFIS models for rainfall-runoff modeling. The comparison of results obtained using processed and unprocessed data, indicates the merit of applied data pre-processing approaches due to robust identification of hidden patterns in data, so that the developed models could simulate and predict runoff values with lower margin of error and higher confidence and the best results were achieved by employing the denoised-jittered data via producing more different training time series with the same pattern of original time series.

For future study, it is recommended to examine the efficiency of the proposed data pre-processing method in rainfall-runoff modeling of other watersheds since it is expected that the merit of the method is more highlighted where the quality of the gathered data is blurred due to the technical limitations.

Furthermore, it is suggested to evaluate the efficiency of the proposed method in modeling the process at other time scales and also for modeling other hydrological processes which may involved distinict noise level and pattern regarding to the type of process.

REFERENCES

Abrahart, R.J., Anctil, F., Coulibaly, P., Dawson, C.W., Mount, N.J., See, L.M., Shamseldin, A.Y., Solomatine, D.P., Toth, E. & Wilby, R.L. (2012) Two decades of anarchy? Emerging themes and outstanding challenges for neural network river forecasting. *Progress Phys. Geogr.* **36**, 480-513. doi: 10.1177/0309133312444943.

Aqil, M., Kita, I., Yano, A. & Nishiyama, S. (2007) Analysis and prediction of flow from local source in a river basin using a Neuro-fuzzy modeling tool. *J. Environm. Manag.,* **85**, 215–223. doi: 10.1016/j.jenvman.2006.09.009.

ASCE task Committee on Application of Artificial Neural Networks in hydrology.(2000) Artificial Neural Networks in hydrology 2: hydrologic applications. *J. Hydrol. Enging.,* **5** (2), 124–137. doi: 10.1061/ (ASCE) 1084-0699(2000)5:2(124).

Bowker, A.H. & Lieberman, G.J. (1972) *Engineering Statistics*. Prentic-Hall. G0132794551I5N01.

Chang, F.J. & Tsai, M.J. (2016) A nonlinear spatio-temporal lumping of radar rainfall for modeling multi-step-ahead inflow forecasts by data-driven techniques. *J. Hydrol.,* **535**, 256-269. doi:10.1016/j.jhydrol.2016.01.056.

Chang, T.K., Talei, A., Alaghmand, S. & Ooi, M.P. (2017) Choice of rainfall inputs for event-based rainfall-runoff modeling in a catchment with multiple rainfall stations using data-driven techniques. *J. Hydrol.* **545**, 100-108. DOI: 10.1016/j.jhydrol.2016.12.024.

Chau, K.W., Wang, W.C. & Chen, X.Y. (2015) A novel hybrid neural network based on continuity equation and fuzzy pattern-recognition for downstream daily river discharge forecasting. *J. Hydroinf.*, **17**(5), 733-744. doi: 10.2166/hydro.2015.095

Donoho, D.L. (1995) Denoising by soft-thresholding. *IEEE Trans. Infor. Theory,* **41**(3), 613–617. doi:10.1109/18.382009.

Donoho, D.L. & Johnstone, I.M. (1995) Ideal spatial adaptation via wavelet shrinkage. *Biometrik,* **81**(3),425-455. doi: 10.1093/biomet/81.3.425.

Hornik, K. (1988) Multilayer feed-forward networks are universal approximators. *Neural Networks,* **2**(5), 359–366. doi:10.1016/0893-6080(89)90020-8.

Jang, J.S.R. Sun, C.T. (1995) Neuro-fuzzy modeling and control. *Proceedings of the IEEE* **83**, 378– 406. doi: 10.1109/5.364486.

Jang, J.S.R., Sun, C.T. & Mizutani, E.(1997) *Neuro-fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence,* Third Ed. Prentice-Hall, New Jersey, U.S.A. 614 pages. Dp/0132610663.

Jansen, M. (2006) Minimum risk thresholds for data with heavy noise. *IEEE Signal Processing Letters,* **13**, 296-299. doi: 10.1109/LSP.2006.870355.

Kalman, R.E. (1960) A new approach to linear filtering and prediction problems. *J. Basic Enging-T ASME,* 35-45. doi:10.1115/1.3662552.

Kim, T. & Valdes, J.B. (2003) Nonlinear model for drought forecasting based on a conjunction of wavelet transforms and neural networks. *J. Hydrol. Enging.,* **8**(6), 319-328. doi: 10.1061/ (ASCE) 1084-0699(2003)8:6(319).

Nejad, F. & Nourani, V. (2012) Elevation of wavelet de-noising performance via an ANN-based streamflow forecasting model. *Int. J. Comp. Sci. Manag. Res.,* **1**(4), 764–770.

Nourani, V. & Mano, A. (2007) Semi-distributed flood runoff model at the subcontinental scale for southwestern Iran. *Hydrol. Processes* **21**, 3173–3180. doi: 10.1002/hyp.6549.

Nourani, V., Komasi, M. & Mano, A. (2009) A multivariate ANN–wavelet approach for rainfall–runoff modeling. *Water Resour. Manag.,* **23**, 2877–2894. doi: 10.1007/s11269-009-9414-5.

Nourani, V., Mehrvand, M. & Hosseini Baghanam, A. (2014*)* Implication of SOM-ANN based clustering for multistation rainfall-runoff modeling. *J. Urban Environm. Enging.,* **8**(2), *198-210.* . doi: 10.4090/juee.2014.v8n2.198210.

Nourani, V., Hosseini-Baghanam, A., Yahyavi-Rahimim, A. & HassanNejad, F. (2014*)* Evaluation of wavelet- based de-noising approach in hydrological models linked to artificial neural networks. *Comp. Intellig. Techn. Earth Environm. Sci.,* **8642**, 3-12. doi: 10.1007/978-94-017-8642-3_12.

Nourani, V. & Saeidifarzad, B. (2016) Detection of land use/cover change effect on watershed's response in generating runoff using computational intelligence approaches. *Stochastic Environm. Res. Risk Assess*., **31**(6), 1-17. doi:10.1007/s00477-016-1220-z.

Rajurkar, M.P., Kothyari, U.C., & Chaube, U.C. (2002) Artificial neural networks for daily rainfall-runoff modeling. *Hydrological Sciences Journal* **47**(6), 865-877. doi: 10.1080/02626660209492996.

Salas, J.D., Delleur, J.W., Yevjevich, V. & Lane, W.L. (1980) *Applied Modeling of Hydrological Time Series, first ed*. Water Resources Publications, Littleton. Colorado, Highlands Ranch, CO, U.S.A. 484 pages. 1980.

Sang, Y.F., Wang, D., Wu, J.C., Zhu, Q.P. & Wang, L. (2009) Entropy-based wavelet de-noising method for time series analysis. *Entropy* **11**, 1123–1147. doi: 10.3390/ e11041123.

Snedecor, G.W. & Cochran, W.G. (1981) *Statistical methods.* Iowa: Iowa State University Press. seventh ed.. U.S.A. 503 pages. DOI: 0813815614.

Walker, J.S. (1999) *A primer on wavelets and their scientific applications*. New York: Chapman and Hall, CRC. doi: 1584887451.

Wang, W.C.H., Chau, K.W., Xu, D.M. & Chen, X.Y. (2015) Improving Forecasting Accuracy of Annual Runoff Time Series Using ARIMA Based on EEMD Decomposition. *Water Resour. Manag.,* **29**, 2655–2675. doi 10.1007/s11269-015-0962-6.

Wiener, N. (1949) *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*: Technology Press of the Massachusetts Institute of Technology, Cambridge, MA. The MIT Press. 176 pages.

Zhang, G.P. (2007) A neural network ensemble method with jittered training data for time series forecasting. *Inform. Sciences* **177**, 5329–5346. doi: S0020025507003003.

Zhang, Q., Wang, B.D., He, B., Peng, Y. & Ren, M.L. (2011) Singular spectrum analysis and ARIMA hybrid model for annual runoff forecasting. *Water Resour. Manag.,* **25**, 2683–2703.

Zur, R.M., Jiang, Y. & Metz, C.E. (2004) Comparison of two methods of adding jitter to artificial neural network training. *Int. Congress Series,* **1268**, 886–889. doi: 10.1016/j.ics.2004.03.238.