Engineering Applications of Artificial Intelligence 26 (3): 997-1007 2013

Prediction of Rainfall Time Series Using Modular Soft Computing Methods

C.L. Wu and K. W. Chau*
Dept. of Civil and Structural Engineering, Hong Kong Polytechnic University,
Hung Hom, Kowloon, Hong Kong, People's Republic of China
(*Email: cekwchau@polyu.edu.hk)

ABSTRACT

In this paper, several soft computing approaches were employed for rainfall prediction. Two aspects were considered to improve the accuracy of rainfall prediction: (1) carrying out a data-preprocessing procedure, and (2) adopting a modular modeling method. The proposed preprocessing techniques included moving average (MA) and singular spectrum analysis (SSA). The modular models were composed of local support vectors regression (SVR) models or/and local artificial neural networks (ANN) models. In the process of rainfall forecasting, the ANN was first used to choose data-preprocessing method from MA and SSA. Modular models involved preprocessing the training data into three crisp subsets (low, medium and high levels) according to the magnitudes of the training data, and finally two SVRs were performed in the medium and high-level subsets whereas ANN or SVR was involved in training and predicting the low-level subset. For daily rainfall record, the low-level subset tended to be modeled by the ANN because it was overwhelming in the training data, which is based on the fact that the ANN is very efficient in training large-size samples due to its parallel information processing configuration. Four rainfall time series consisting of two monthly rainfalls and two daily rainfalls from different regions were utilized to evaluate modular models at 1-day, 2-day, and 3-day lead-time with the persistence method and the global ANN as benchmarks. Results showed that the MA was superior to the SSA when they were coupled with the ANN. Comparison results indicated that modular models (referred to as ANN-SVR for daily rainfall simulations and MSVR for monthly rainfall simulations) outperformed other models. The ANN-MA also displayed considerable accuracy in rainfall forecasts compared with the benchmark.

KEYWORDS

Rainfall prediction, Moving Average, Singular spectral analysis, Fuzzy C-Means clustering, ANN, SVR, Modular model

1. Introduction

An accurate and timely rainfall forecast is crucial for reservoir operation and flooding prevention because it can provide an extension of lead-time of the flow forecast, larger than the response time of the watershed, in particular for small and medium-sized mountainous basins.

Rainfall prediction is a very complex problem. Simulating the response using conventional approaches in modelling rainfall time series is far from a trivial task since the hydrologic processes are complex and involve various inherently complex predictors such as geomorphologic and climatic factors, which are still not well understood. As such, the artificial neural network algorithm becomes an attractive inductive approach in rainfall prediction owing to their highly nonlinearity, flexibility and data-driven learning in building models without any prior knowledge about catchment behavior and flow processes. They are purely based on the information retrieved from the hydro-meteorological data and act as black box.

Many studies have been conducted for the quantitative precipitation forecast (QPF) using diverse techniques including numerical weather prediction (NWP) models and remote sensing observations (*Davolio et al., 2008; Diomede et al. 2008*; *Ganguly and Bras, 2003; Sheng, et al., 2006; Yates et al., 2000*), statistical models (*Chan and Shi, 1999;Chu and He,*

1995;DelSole and Shukla,2002;Li and Zeng, 2008;Munot, 2007; Nayagam et al., 2008), chaos-based approach (*Jayawardena et al., 1994*), non-parametric nearest-neighbors method (*Toth et al, 2000*), and soft computing-based methods including artificial neural networks (ANN), support vector regression (SVR) and fuzzy logic (FL) (*Brath et al 2002; Dorum et al. 2010; Guhathakurta 2008; Nasseri et al. 2008; Pongracz et al. 2001; Sedki et al. 2009; Silverman and Dracup 2000; Sivapragasam et al. 2001; Surajit, and Goutami, 2007; Talei et al. 2010; Toth et al 2000; Venkatesan et al. 1997).* The contemporary studies focused on soft computing-based methods. Several examples of such methods can be mentioned. Venkatesan et al. (*1997*) employed the ANN to predict the all India summer monsoon rainfall with different meteorological parameters as model inputs. Chattopadhyay and Chattopadhyay (*2008a*) constructed an ANN model to predict monsoon rainfall in India depending on the rainfall series alone. The fuzzy logic theory was applied to monthly rainfall prediction by Pongracz et al. (*2001*). Toth et al. (*2000*) applied three time series models, auto-regressive moving average (ARMA), ANN and k-nearest-neighbors (KNN) method, to short-term rainfall prediction. The results showed that the ANN performed the best in the improvement of the runoff forecasting accuracy when the predicted rainfall was used as inputs of the rainfall-runoff model. ANN has also been applied on general circulation model (GCM). Chadwick et al. (2011) employed an artificial neural network approach to downscale GCM temperature and rainfall fields to regional model scale over Europe. Sachindra et al. (2011) developed a model with various soft computing techniques capable of statistically downscaling monthly GCM outputs to catchment scale monthly streamflows, accounting for the climate change.

Recently, models based on combining concepts have been paid more attention in hydrologic forecasting. Depending on different combination methods, combining models can be categorized into ensemble models and modular (or hybrid) models. The basic idea behind the ensemble models is to build several different or similar models for the same process and to combine them in a combining method (*Abrahart and See, 2002; Kim et al, 2006; Shamseldin et al., 1997; Shamseldin, and O'Connor, 1999; Xiong et al., 2001*). For example, Xiong et al. (*2001*) used a Takagi-Sugeno fuzzy technique to combine several conceptual rainfall-runoff models. Coulibaly et al. (*2005*) employed an improved weighted-average method to coalesce forecasted daily reservoir inflows from the KNN model, conceptual model and ANN model. Kim et al. (*2006*) investigated five combining methods for improving ensemble streamflow prediction.

Physical processes in rainfall and/or runoff are generally composed of a number of sub-processes so that their accurate modeling by the building of a single global model is often not possible. Modular models are therefore proposed where sub-processes are first of all identified and then separate models (also called local or expert model) are established for each of them (*Solomatine, and Ostfeld, 2008*). In these modular models, the split of training data can be soft or crisp. The soft split means the dataset can be overlapped and the overall forecasting output is the weighted-average of each local model (*Shrestha and Solomatine, 2006; Zhang and Govindaraju, 2000; Wang et al., 2006; Wu et al., 2008*). Zhang and Govindaraju (*2000*) examined the performance of modular networks in predicting monthly discharges based on the Bayesian concept. Wu et al. (*2008*) employed a distributed SVR for daily river stage prediction. On the contrary, there is no overlap of data in the crisp split and the final forecasting output is generated explicitly from one of the local models (*Corzo and*

Solomatine, 2007; Jain and Srinivasulu,2006; See and Openshaw, 2000; Sivapragasam and Liong, 2005; Solomatine and Xue, 2004). Solomatine and Xue (*2004*) used M5 model trees and neural networks in a flood-forecasting problem. Sivapragasam and Liong (*2005*) divided the flow range into three regions, and employed different SVR models to predict daily flows in high, medium and low regions.

Apart from the adoption of the modular model, the improvement of predictions may be expected by suitable data preprocessing techniques. Besides the conventional rescaling or standardization of training data, preprocessing methods from the perspective of signal analysis are also crucial because rainfall time series may be also viewed as a quasi-periodic signal, which is contaminated by various noises. Hence techniques such as singular spectrum analysis (SSA) were recently introduced to hydrology field by some researchers (*Marques et al., 2006; Partal and Kişi, 2007; Sivapragasam et al., 2001*). Sivapragasam et al. (*2001*) established a hybrid model of support vector machine (SVM) and the SSA for rainfall and runoff predictions. The hybrid model resulted in a considerable improvement in the model performance in comparison with the original SVM model. The application of wavelet analysis to precipitation was undertaken by Partal and Kişi (*2007*). Their results indicated that the wavelet analysis was highly promising. In addition, the issue of lagged predictions in the ANN model was mentioned by some researchers (*Dawson and Wilby, 2001; Jain and Srinivasulu, 2004; de Vos and Rientjes, 2005; Muttil and Chau, 2006*). A main reason on lagged predictions was the use of previous observed data as ANN inputs (*de Vos and Rientjes, 2005*). An effective solution was to obtain new model inputs by moving average over the original data series.

The scope of this study was to investigate the effect of the MA and SSA as data-preprocessing techniques and to couple with modular models in improving model performance for rainfall prediction. The modular model included three local models which were associated with three crisp subsets (low-, medium- and high-intensity rainfall) clustered by fuzzy C-mean (FCM) method. Tthe ANN was first used to choose data-preprocessing method from MA and SSA. Depending on the selected data-preprocessing technique, modular models were employed to perform rainfall prediction. Generally, the ANN is very efficient in processing large-size training samples due to its parallel information processing configuration. The biggest drawback is that the model outputs are variable because of the random initialization of weights and biases. The SVR holds a good generalization and more stable model outputs. However, it is suitable for a small-size training sample (e.g. below 200) because the training time exponentially increases with the size of training samples. For the current rainfall data, the majority of subsets after data split belong to a small-size sample except for the low-intensity daily rainfall. Therefore, three local SVRs (hereafter referred as to MSVR) were employed for monthly rainfall data whereas two local SVRs and one ANN (hereafter referred to as ANN-SVR) were adopted for daily rainfall data. For daily rainfall record, the low-intensity subset was modeled by the ANN because it was overwhelming in the training data. For the comparison purpose, the global ANN and the persistence model were used as benchmarks. To ensure generalization of this study, four cases consisting of two monthly rainfall series and two daily rainfall series from India and China, were explored.

2. Methodology

2.1 Data-preprocessing techniques

(1) Moving Average (MA)

The moving average method smoothes data by replacing each data point with the average of the K neighboring data points, where K may be called the length of memory window. The basic idea behind the method is that any large irregular component at any point in time will exert a smaller effect if we average the point with its immediate neighbors (*Newbold et al., 2003*). The most common moving average method is the unweighted moving average, in which each value of the data carries the same weight in the smoothing process. For time series $\{x_1, x_2, \cdots, x_N\}$, the K-term unweighted moving average is written as $x_t^* = (\sum_{i=0}^{K-1} x_{t-i})/K$ (where $t = K, \ldots N$; x_t^* stands for the moving average value) when the backward moving mode is adopted (*Lee et al., 2000*). The choice of the window length K is by a trial and error procedure to minimize the ANN prediction error.

(2) Singular spectrum analysis (SSA)

The SSA is able to decompose the daily rainfall series into several additive components that typically can be interpreted as 'trend' components, various 'oscillatory' components, and 'noise' components (*Golyandina et al. 2001*). The basic algorithm of the SSA can be referred to Vautard *et al.* (*1992*) and Golyandina et al. (*2001*). Following the methodology in Vautard *et al.* (*1992*), four steps are performed for the explanation of the SSA algorithm on a univariate time series $\{x_1, x_2, \cdots, x_N\}$. The first step is to construct the 'trajectory matrix'. The 'trajectory matrix' results from the method of delays. In the method of delays, the coordinates of the phase space will approximate the dynamic of the system by using lagged copies of the time series. The 'trajectory matrix', denoted by $\mathbf{X}$, therefore reflects the evolution of the time series with a suitable choice of (τ, m) window, where m is the window length (also called singular number) and τ is the delay time.

The next step is the singular value decomposition (SVD) of $\mathbf{X}$. Let $\mathbf{S} = \mathbf{X}^{\mathrm{T}}\mathbf{X}$ (called the lagged-covariance matrix). With SVD, $\mathbf{X}$ can be written as $\mathbf{X} = \mathbf{DLE}^{\mathrm{T}}$ where $\mathbf{D}$ and $\mathbf{E}$ are left and right singular vectors of $\mathbf{X}$, and $\mathbf{L}$ is a diagonal matrix of singular values. $\mathbf{E}$ consists of orthonormal columns, and is also called the 'empirical orthonormal functions' (EOFs). Substituting $\mathbf{X}$ into the definition of $\mathbf{S}$ yields the formula of $\mathbf{S} = \mathbf{EL}^2\mathbf{E}^{\mathrm{T}}$. Further $\mathbf{S}=\mathbf{E}\wedge\mathbf{E}^{\mathrm{T}}$ since $\mathbf{L}^2=\wedge$ where $\wedge$ is a diagonal matrix consisting of ordered values $0 \le \lambda_1 \le \lambda_2 \le \cdots \lambda_m$. Therefore, the right singular vectors of $\mathbf{X}$ are the eigenvectors of $\mathbf{S}$. In other words, the singular vectors $\mathbf{E}$ and singular values of $\mathbf{X}$ can be respectively attained by calculating the eigenvectors and the square roots of the eigenvalues of $\mathbf{S}$.

The third step is to calculate the principal components (a_i^k's) by projecting the original time record onto the eigenvectors as follows,

$$a_i^k = \sum_{j=1}^{L} x_{i+(j-1)\tau} e_j^k, \text{ for } \quad i = 1, 2\cdots, N-m+1 \tag{1}$$

where e_j^k represents the jth component of the kth eigenvector. Each principal component is a filtered process of the original series with length n, where $n = N-m+1$.

The last step is to generate reconstruction components (RCs) with the length size being the same as the original series. The generation of each RC depends on a convolution of one principal component with the corresponding singular vector (*Vautard et al. 1992*). Therefore, The m RCs can be achieved if all m principal components and their associated eigenvectors are employed in the process of signal reconstruction. Also, the original record can be filtered by choosing $p\,(<m)$ RCs.

2.2 Forecasting models

(1) Persistence model

The persistence method takes the last observed value as the future rainfall estimate, i.e. $x_{t+T}^{F}=x_t$, for $\forall T$, where x_t is the observed record at instant time t, and x_{t+T}^{F} stands for the estimated rainfall at the lead-time T. For a modified version of the persistence model, each forecasted rainfall at the lead-time T equals to the mean value over the last T observations, given by $x_{t+T}^{F}=\sum_{\mathrm{i}=1}^{\mathrm{T}} x_{t-i+1}\big/T$. The modified version was adopted in this study.

(2) Artificial neural networks

The feed-forward multilayer perceptron (MLP) among many ANN paradigms is by far the most popular, which usually uses the technique of error back propagation to train the network configuration. The architecture of the ANN consists of the number of hidden layers and the number of neurons in input layer, hidden layers and output layer. ANNs with one hidden layer are commonly used in hydrologic modeling (*Dawson and Wilby, 2001; de Vos and Rientjes, 2005*) since these networks are considered to provide enough complexity to accurately simulate the nonlinear properties of the hydrologic process. A three-layer ANN is chosen for the current study, which comprises the input layer with *m* nodes (i.e., *m* past daily rainfall), the hidden layer with *h* nodes (neurons), and the output layer with one node. The hyperbolic tangent functions are used as transfer functions in the hidden layer and the output layer. The model architecture is described by the equation:

$$x_{t+T}^{F}=f(\mathbf{X_t},w,\theta,m,h)=\theta_0+\sum_{j=1}^{h} w_j^{out}\tanh\Big(\sum_{i=1}^{m} w_{ji}x_{t-i+1}+\theta_j\Big) \tag{2}$$

where x_{t-i+1} , $i=1,\cdots,m$ are the m elements in the input vector $\mathbf{X_t}$. w_{ji} are the weights defining the link between the ith node of the input layer and the jth of the hidden layer; θ_j are biases associated to the jth node of the hidden layer; w_j^{out} are the weights associated to the connection between the jth node of the hidden layer and the node of the output layer; and θ_0 is the bias at the output node. The Levernberg-Marquardt (LM) training algorithm is used to adjust the w and θ in view of being faster and less easily trapped in local minima compared with some local optimization methods (*Toth et al., 2000*).

(3) Support vector regression

SVR performs structural risk minimization (SRM) that aims at minimizing a bound on the generalization error (*Gunn, 1998; Kecman, 2001; Yu et al, 2006; Wu et al., 2008*). It creates a model with good generalization. The SVR can be divided into linear and nonlinear depending on the kernel function being linear or nonlinear. A nonlinear SVR was used in this study. Similar to Eq. (1), the underlying function $f(\square)$ in the context of the nonlinear SVR is given by

$$x_{t+T}^{F} = f(\mathbf{X_t}, \omega) = \omega \cdot \phi(\mathbf{X_t}) + b \tag{3}$$

where the input vector $\mathbf{X_t}$ in the input space is mapped to a high dimensional feature space via a nonlinear mapping function $\phi(\mathbf{X_t})$. The objective of the SVR is to find optimal ω, b and some parameters in kernel function $\phi(\mathbf{X_t})$ so as to construct an approximation function of the $f(\square)$.

When introducing Vapnik's ε-insensitivity error or loss function, the loss function $L_\varepsilon(\mathrm{y}, f(\mathbf{X_t}, \omega))$ on the underlying function can be defined as

$$L_\varepsilon(\mathrm{y}, f(\mathbf{X_t}, \omega)) = \left|\mathrm{y} - f(\mathbf{X_t}, \omega)\right|_\varepsilon = \begin{cases} 0 & if \left|\mathrm{y} - \left(\omega \cdot \phi(\mathbf{X_t}) + b\right)\right| \le \varepsilon \\ \left|\mathrm{y} - \left(\omega \cdot \phi(\mathbf{X_t}) + b\right)\right| - \varepsilon & otherwise \end{cases} \tag{4}$$

where y represents observed value. The nonlinear SVR problem can be expressed as the following optimization problem (*Kecman, 2001; Yu et al., 2006*):

$$\begin{aligned} &minimize \quad R_{\omega,\xi_i,\xi_i^*} = \frac{1}{2}\|\omega\|^2 + C\sum_{\mathrm{i}=1}^{n}(\xi_i + \xi_i^*) \\ &subject \quad to \quad \begin{cases} \mathrm{y_i} - f(\phi(\mathbf{X_i}), \omega) - b \le \varepsilon + \xi_i \\ f(\phi(\mathbf{X}_i), \omega) + b - \mathrm{y_i} \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \end{aligned} \tag{5}$$

where $\mathbf{X_i}$ represents $\mathbf{X_t}$ for simplicity, the term of $\frac{1}{2}\|\omega\|^2$ reflects generalization, and the term of $C\sum_{i=1}^{n}(\xi_i + \xi_i^*)$ stands for empirical risk. The objective in Eq. (5) is to minimize them simultaneously, which implements SVR to avoid underfitting and overfitting the training data. ξ_i and ξ_i^* are slack variables for measurements "above" and "below" an ε tube. Both slack variables are positive values. C is a positive constant that determines the degree of penalized loss when a training error occurs.

By introducing a dual set of Lagrange multipliers, α_i and α_i^*, the objective function in dual form can be represented as (*Gunn, 1998*)

$$\begin{aligned} &maximize \quad L_d(\alpha, \alpha^*) = -\varepsilon\sum_{\mathrm{i}=1}^{n}(\alpha_i^* + \alpha_i) + \sum_{\mathrm{i}=1}^{n}(\alpha_i^* - \alpha_i)y_i - \frac{1}{2}\sum_{\mathrm{i,j}=1}^{n}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)\left(\phi(\mathbf{X_i}) \cdot \phi(\mathbf{X_j})\right) \\ &subject \quad to \quad \begin{cases} \sum_{\mathrm{i}=1}^{n}(\alpha_i - \alpha_i^*) = 0 \\ 0 \le \alpha_i^* \le C, \qquad i = 1, \cdots, n \\ 0 \le \alpha_i \le C, \qquad i = 1, \cdots, n \end{cases} \end{aligned} \tag{4}$$

By using a "kernel" function $K(\mathbf{X_i}, \mathbf{X}_j) = \left(\phi(\mathbf{X}_i) \cdot \phi(\mathbf{X_j})\right)$ to yield inner products in feature space, the computation in input space can be performed. In the present study, Gaussian radial basis function (RBF) was adopted in the form of $K(\mathbf{X_i}, \mathbf{X}_j) = \exp(-\|\mathbf{X}_i - \mathbf{X}_j\|^2 / 2\sigma^2)$. Once parameters α_i, α_i^*, and b_0 are obtained, the final approximation function of the $f(\square)$ becomes

$$f(\mathbf{X}_i) = \sum_{i=1}^{n} (\alpha_k - \alpha_k^*) K(\mathbf{X}_k \cdot \mathbf{X}_i) + b_0 \ , \ k = 1, \cdots, s \tag{5}$$

where $\mathbf{X}_k$ stands for the support vector, α_k and α_k^* are parameters associated with support vector $\mathbf{X}_k$, n and s represent the number of training samples and support vectors respectively. Three parameters (C, ε, σ) need to be optimized in order to identify the optimal $f(\square)$ as Eq. (5). In the current study, a two-step genetic algorithm method was adopted for the optimization (*Wu et al., 2008*).

(4) Modular model

As described at the end of the introduction, the modular model consisted of three local models which were based on the crisp data split. The three models simulated three-type rainfall events, i.e., no (dry period) or low-intensity, medium-intensity, and high-intensity rainfalls (or storm events). The final output of the hybrid models was obtained directly from the output of a triggered local model. For daily rainfall simulation, the modular model was referred to as ANN-SVR, and the modular model was denoted as MSVR for monthly rainfall simulation.

2.3 Models coupled with MA or SSA

The combination of studied models (i.e. ANN or modular models) with MA was very simple where a new series was obtained by the moving average over the raw rainfall data and then the new series was used to construct the model inputs. On the other hand, the combination of studied models with the SSA was somewhat complex. The methodological procedures of the combination were summarized in the following steps:

- First, the raw rainfall series was decomposed into *m* components by the SSA.
- The cross-correlation functions (CCFs) between each RC and the raw rainfall series were computed. The mean CCF was then generated by averaging over all CCFs at the same lag.
- The CCFs were sorted in a descending or ascending order depending on the mean CCF was negative or positive (e.g. An ascending order was adopted if the mean CCF is negative)
- The sorted RCs were filtered by systematically deleting the RCs where all *m* RCs were originally remained and only the first RC was left in the end. A new series was constituted by summing the remaining RCs each time. The new series was used to construct model inputs. Repeating the filtering operation, the optimal RCs can be attained, which was associated with the optimal model performance.

2.4 Evaluation of model performances

As suggested by Legates and McCabe (*1999*), a complete assessment of model performance at least include absolute error measure and relative error measure. Therefore, the measures of model performance evaluation for the present study comprise root mean square error (RMSE), the Nash-Sutcliffe coefficient of efficiency (CE), the persistence index (PI) (*Kitanidis And Bras, 1980*), and Willmott's index (d) (*Chattopadhyay and Chattopadhyay 2008b*). They are respectively formulated as: $\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$, $\mathrm{CE} = 1 - \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \Big/ \sum_{i=1}^{n}(y_i - \bar{y})^2$, $\mathrm{PI} = 1 - \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \Big/ \sum_{i=1}^{n}(y_i - y_{i-T})^2$, and

$d = 1 - \left[\sum_{i=1}^{n} (\hat{y}_i - y_i)^2 / \sum_{i=1}^{n} (| \hat{y}_i - \bar{y} | + | y_i - \bar{y} |)^2 \right]$. In these formula, n =number of observations, $\hat{y}_i$ =forecasted rainfall, y_i =observed rainfall, $\bar{y}$ =average observed rainfall, and y_{i-T} is the rainfall estimate from the persistence model. CE and PI values of 1 stand for perfect fits. Higher *d* indices would be more efficient predictive models than those with lower *d* values.

3. Case study

Two daily mean rainfall series (at Zhenwan and Wuxi raingauge stations, respectively) from Zhenshui and Da'ninghe watersheds of China, and two monthly mean rainfall series from India and Zhongxian raingauge station of China, were analyzed in this study.

The Zhenshui basin is located in the north of Guangdong province and adjoined by Hunan province and Jianxi Province. The basin belongs to a second-order tributary of the Pearl River and has an area of 7554 km^2. The daily rainfall time series data of Zhenwan raingauge was collected between 1 January 1995 and 31 December 1998 (hereafter the rainfall series is referred to as Zhenwan).

The Da'ninghe basin, a first-order tributary of the Yangtze River, is located in the northwest of Hubei Province. The daily rainfall data from Jan. 1, 2004 to Dec. 31, 2007 were measured at six raingauges located at the upstream of the study basin. The upstream part of the Da'ninghe basin is controlled by Wuxi hydrology station, with a drainage area of around 2 000 km^2. The rainfall was spatially average with the Thiessen polygon method (hereafter the averaged rainfall series is referred to as Wuxi).

The all Indian average monthly rainfall was estimated from area-weighted observations at 306 land stations uniformly distributed over India. The data period spanned from January 1871 to December 2006 available at the website http://www.tropmet.res.in run by the Indian Institute of Tropical Meteorology.

The other monthly rainfall series was from Zhongxian raingauge which is located in Chongqing city, China. The basin containing this raingauge belongs to a first-order tributary of the Yangtze River. The monthly rainfall data were collected from January 1956 to December 2007.

Hence, Zhenwan and Wuxi series are of daily rainfall and India and Zhongxian are of monthly rainfall. Each rainfall series was partitioned into three parts as training set, cross-validation set and testing set. The training set served the model training and the testing set was used to evaluate the performances of models. The cross-validation set aimed to implement an early stopping approach for avoiding the overfitting. The same data partition format was adopted in four rainfall series: the first half of the entire data as training set and the first half of the remaining data as cross-validation set and the other half as testing set.

4. Applications to the rainfall data

4.1 Decomposition of rainfall data

The decomposition of the daily average rainfall series requires identifying the window length *m* (or the singular number) if the interval of neighboring points in discrete

time series is defaulted as the lag time (i.e. τ =1 day for daily rainfall data or 1 month for monthly rainfall data). The reasonable value of m should give rise to a clear resolution of the original signal. The present study does not need accurately resolve any trends or oscillations in the raw rainfall signal. A rough resolution can be adequate for the separation of signals and noises. Therefore, a small value of m was chosen for the present analysis. Fig. 1 displays the singular spectrum as a function of lag using various window lengths m from 5 to 10 for four rainfall series. Results show that the decomposition of the Wuxi rainfall was insensitive to m because singular values were distinguishable for all m. Thus, we set a criterion to decide the value of m, i.e. the SSA with the targeted m can generate clearly identified singular values. Therefore, the value of m for Wuxi was arbitrarily chosen from 5 to 10, say 5 here. The values of m were respectively 7, 6, and 7 for India, Zhongxian, and Zhenwan.

Taking Wuxi as an example, Fig. 2 presents five RCs and the raw rainfall records excluding the testing data. The RC1 represents an obvious low-frequency oscillation, which exhibits a similar mode to the raw rainfall. The other RCs reflect high-frequency oscillations, part of which can be deleted to improve the mapping between the ANN inputs and output. Fig. 3 depicts cross-correlation functions (CCF) between RC and the original rainfall data. The last plot in Fig. 3 denotes the average CCF, which was generated by averaging over all five CCFs at the same lag.

4.2 Identification of the ANN architecture

The ANN was used to choose data-preprocessing method from MA and SSA for modular models. Therefore, it is imperative to identify a reasonable ANN architecture before evaluating MA and SSA.

The architecture identification included the determination of model inputs and the number of nodes (or neurons) in the hidden layer when there was one model output. Input selection is one of the important tasks in the identification but is very subjective although many methods have been documented (*Bowden et al., 2005*). The statistical approach to examine auto- and partial-auto-correlation of the observed time series was recognized as a good and parsimonious method in the determination of model inputs (*Kişi, 2008; Sudheer et al., 2002*). The model inputs in the approach are mainly determined by the plot of partial-auto-correlation function (PACF). According to the approach, model inputs were respectively previous 12-month observations for India, 13-month observations for Zhongxian, 7-day observations for Wuxi, and 3-day observations for Zhenwan because the PACF values decayed within the confidence band around at these lags (Fig. 4). The ensuing task was to optimize the size of the hidden layer with identified model inputs and one output. The optimal size h of the hidden layer was found by systematically increasing the number of hidden neurons from 1 to 10. The identified ANN architectures were: 12-5-1 for India, 13-6-1 for Zhongxian, 7-6-1 for Wuxi and 3-4-1 for Zhenwan.

In the process of forecasting, the training data was rescaled to [-1, 1] due to the use of hyperbolic tangent function as transfer function in this study. Considering the instability of the ANN output, the ANN prediction was an average of the best 10 outputs from entire 30 runs. In addition, the same ANN was applied to multi-step lead-time forecasting in which a static method (directly having the multi-step-ahead prediction as output) was adopted.

4.3 Implementation of models

(1) ANN-MA

The window length K in MA can be determined by trial and error with various K from 1 to 10. The targeted value of K was associated with the optimal ANN performance in terms of RMSE. Table 1 shows the values of K at one-, two-, and three-day-ahead predictions for all studied cases.

(2) ANN-SSA

Following the methodological procedure in Section 2.3, the filter of RCs for one-day-ahead prediction was described below using the Wuxi rainfall data.

- First, the average CCF at lag 1 can be found 0.13 (Fig. 3).
- Then, RCs 5 was sorted in a descending order according to the CCF value at lag 1 of each RC plot in Fig 3. The new order was RC1, RC2, RC3, RC4, and RC5, which was the same as the original order.
- A five-time trial and error was finally used to select the $p\,(\leq m)$. It was found that the ANN had the minimum RMSE when p was 3 (i.e. the preserved RCs were RC1, RC2 and RC3).

Fig.5 shows the results of the SSA filter at three prediction horizons using the rainfall data from India and Wuxi. It can be seen that the number of the remained RCs can be different at various prediction horizons. For instance, the numbers of chosen RCs in three forecasting horizons were respectively 3, 4, and 3 for India, and 3, 1, and 3 for Wuxi. The results of the SSA filter for all four rainfall case are presented in Table 1.

(3) Modular models

The MA was more effective than the SSA when they were in conjunction with the ANN (see Tables 2 and 3 below). Therefore, modular models were only coupled with the MA in the current rainfall prediction. Moreover, the identified parameter K in MA by the ANN was also applied to modular models.

5. Results and discussion

5.1 Results

The overall performances of each model in terms of RMSE, CE, and PI are presented in Table 2 for two monthly rainfall series and Table 3 for two daily rainfall series. It can be seen that two benchmark models of persistence and ANN demonstrated very poor performances for all four cases except for India. The performances from ANN-MA and ANN-SSA indicate that data-preprocessing methods resulted in considerable improvement in the accuracy of the rainfall forecasting. Moreover, the MA seems superior to the SSA except for the Zhongxian rainfall at three-month lead prediction, even where the MA and the SSA also showed similar performance. Coupled with the MA, modular models performed the best among all models for each rainfall series.

Fig. 6 shows the hyetograph graphs and scatter plots of the results of one-month lead prediction from the ANN, ANN-MA, and MSVR using the Zhongxian data. For Zhenwan, one-day lead rainfall estimates of the ANN, ANN-MA, and ANN-SVR are depicted in Fig. 7 in the form of hyetographs and scatter plots (the former was plotted in a selected range for better visual inspection). As seen from the hyetograph graphs that the modular models, MSVR or ANN-SVR, were able to reproduce the corresponding observed rainfall data better

than ANN and ANN-MA. It can be also seen from the scatter plots that the predictions of the modular models were much closer to the exact fit line than those of ANN and ANN-MA. The scatter plots from the modular model and ANN-MA indicate that three local models better approximated different rainfall characteristics than a single global model. It is worth noting that the modular model generated some negative estimates at low-intensity rainfall points although it well pursued most of observed rainfall data. As far as ANN-MA is concerned, the prediction accuracy was substantially improved. However, the model underestimated quite a number of moderate peak rainfalls although low-intensity rainfalls were mostly well simulated.

5.2 Discussion

The poor performances of ANN (Tables 2 and 3) imply that the ANN fed by the original data is less viable for the rainfall forecasting, in particular using daily rainfall data. Actually, the ANN mainly captured the zero or low-intensity rainfall patterns (dry periods) in daily rainfall series because the type of pattern was dominant when using the original rainfall data to construct model input/output pairs. The MA and the SSA filter on the raw rainfall records substantially eliminated those patterns. Therefore, the trained ANN coupled with the MA or SSA was able to pay more attention to medium or high-intensity rainfall patterns, which improve the ANN generalization.

It may be noted that, the ANN performed better using the India data than using the Zhongxian data although both of them were monthly rainfall series (Table 2). This implies that the data characteristic may be an important factor to the performance of ANN. Some studies have indicated that considerations of statistical principles may improve ANN model performance (*Cheng and Titterington, 1994; Sarle, 1994*). For example, the training data was recommended to be normally distributed (*Fortin et al., 1997*). Sudheer et al. (*2003*) suggested that the issue of stationarity should be considered in the ANN development because the ANN cannot account for trends and heteroscedasticity in the data. Their results showed that data transformation to reduce the skewness of data can significantly improve the model performance. Depending on the statistical properties of the studied data series, the method of n-th root data transformation was used. In the meantime, for the purpose of comparison with the rescaling method, the standardization/normalization was also considered as a data-preprocessing approach for transfer functions in the ANN. When the standardization approach was adopted, we used the linear transfer function (e.g. purelin) instead of the hyperbolic tangent function in the output layer. Therefore, four preprocessing procedures including the original rescaling were addressed:

- Rescaling the raw data (referred to as Resc_raw);
- Rescaling the n-th root transformed data (referred to as Resc_nth_root);
- Standardizing the original data (referred to as Std_raw);
- Standardizing the n-th root transformed data (referred to as Std_nth_root).

The ANN forecasts with these procedures are presented in Table 4. Compared with the Resc_raw, it can be seen that using the root of n-th degree as data transformation was ineffective for the improvement of the ANN performance. However, the standardization method was effective for the Zhongxian data, which considerably improved the forecast accuracy. It can be concluded that standardization/normalization is a good alternative to rescaling for the ANN in the present study.

Table 5 compares the statistical parameters of rainfall series, skewness coefficient and kurtosis, in three scenarios of no treatment, moving average and root of n-th degree. Each original rainfall series were extremely non-stationary except for the India data being more stationary. The nth-root data transformation did not result in an expected improvement of the ANN performance although the transformed rainfall series was closer to a normal distribution. On the contrary, the moving average truly improved the ANN performance but the transformed rainfall series was still far from a normal distribution. Seemingly, the requirement of a normal distribution on studied data is not necessary.

The combination of SSA and MA was also investigated by using the moving average over the filtered rainfall series. Table 6 demonstrates one-day lead prediction for four case studies. The ANN with the single MA still outperformed the ANN with the combination of SSA and MA.

In summary, the optimal data-preprocessing for the ANN was the MA and the standardization in the present study. The former was for the purpose of smoothing the raw rainfall whereas the latter was needed by the transfer function in the ANN.

6. Conclusions

The purpose of this study was to investigate the effect of modular models coupled with data-preprocessing techniques in improving the accuracy of rainfall forecasting. The modular models consisted of three local SVR and/or ANN. A three-layer feed-forward ANN was used to examine two data-preprocessing techniques, MA and SSA. Results show that the MA was superior to the SSA. Four rainfall records, India, Zhongxian, Wuxi and Zhenwan, from India and China, were used as testing cases.

With the help of the MA, modular models showed the best performance when compared with the ANN-MA and two baseline models, the persistence model and the ANN. Reasonable rainfall estimates were also obtained from the ANN-MA model. The model, however, underestimated quite a number of moderate peak rainfalls although low-intensity rainfalls were mostly well simulated. The ANN model, directly fed by the original data, seemed unsuitable for the current rainfall series except for India. As far as the daily rainfall data were concerned, the ANN mainly captured the zero or low-intensity rainfall patterns (dry periods) in daily rainfall series because the type of pattern was dominant when using the original rainfall data to construct model input/output pairs. The MA or the SSA filter on the raw rainfall records substantially eliminated those patterns. Therefore, the trained ANN with the help of MA or SSA was able to pay more attention to medium or high-intensity rainfall patterns, which improve the ANN generalization.

In addition, the effect of other data-preprocessing techniques including data-transformation and standardization on the ANN performance was also examined. It was found that the standardization method was able to substitute for the rescaling method from the perspective of the transfer function in the ANN. However, the data-transformation method to meet an approximately normal distribution seemed to be unnecessary for the ANN.

References

Abrahart, R. J., and See, L. (2002). Multi-model data fusion for river flow forecasting: An evaluation of six alternative methods based on two contrasting catchment, ***Hydrology Earth Syst. Sci.*** 6(4), 655–670.

Bowden, G.J., Dandy, G.C., Maier, H.R. (2005). Input determination for neural network models in water resources applications: Part 1—background and methodology. ***Journal of Hydrology***, 301, 75–92.

Brath, A., Montanari, A., and Toth, E. (2002). Neural networks and non-parametric methods for improving real time flood forecasting through conceptual hydrological models. ***Hydrology and Earth System Sciences***, 6(4), 627-640.

Chadwick, R., Coppola, E., and Giorgi, F. (2011). An artificial neural network technique for downscaling GCM outputs to RCM spatial scale. ***Nonlinear Processes in Geophysics*** 18 (6), 1013-1028

Chan, J.C.L. and Shi, J.E. (1999). Prediction of the summer monsoon rainfall over South China. ***International Journal of Climatology***, 19 (11), 1255-1265.

Chattopadhyay, S. and Chattopadhyay, G. (2008a). Identification of the best hidden layer size for three-layered neural net in predicting monsoon rainfall in India. ***Journal of Hydroinformatics***, 10(2), 181-188.

Chattopadhyay, S. and Chattopadhyay, G. (2008b), Comparative study among different neural net learning algorithms applied to rainfall time series. ***Meteorological Applications*** 15: 273-280. doi: 10.1002/met.71

Cheng, B, and Titterington DM. (1994). Neural networks: A review from a statistical perspective. ***Statistical Sciences***, 9(1), 2–54.

Chu, P.S., and He, Y.X. (1995). Long-Range Prediction of Hawaiian Winter Rainfall Using Canonical Correlation-Analysis. ***International Journal of Climatology***, 15 (1), 2-2.

Corzo, G. and Solomatine, D. P. (2007). Baseflow separation techniques for modular artificial neural network modelling in flow forecasting. ***Hydrological Science Journal***, 52 (3), 491–507.

Coulibaly, P., Haché, M., Fortin, V., and Bobée, B. (2005). Improving daily reservoir inflow forecasts with model combination. ***Journal of Hydrologic Engineering***, 10(2), 91-99.

Dawson, C. W., and Wilby, R. L. (2001). Hydrological Modeling Using Artificial Neural Networks. ***Progress in Physical Geography***, 25(1), 80-108.

Davolio, S., Miglietta, M. M., Diomede, T., Marsigli, C., Morgillo, A., and Moscatello, A. (2008). A meteo-hydrological prediction system based on a multi-model approach for precipitation forecasting. ***Natural hazards and earth system sciences***, 8 (1): 143-159.

DelSole, T, and Shukla, J. (2002). Linear prediction of Indian monsoon rainfall. ***Journal of Climate***, 15 (24): 3645-3658.

De Vos, N.J. and Rientjes, T.H.M. (2005). Constraints of artificial neural networks for rainfall-runoff modeling: trade-offs in hydrological state representation and model evaluation. ***Hydrology and Earth System Sciences***, 9, 111-126.

Diomede, T.,Davolio, S., Marsigli, C., Miglietta, M. M., Moscatello, A., Papetti, P., Paccagnella, T., Buzzi, A., Malguzzi, P. (2008). Discharge prediction based on multi-model precipitation forecasts. ***Meteorology and Atmospheric Physics***, 101 (3-4): 245-265.

Dorum, A., Yarar, A., Sevimli, M.F., Onüçyildiz M. (2010). Optimized scenario for rainfall forecasting using genetic algorithm coupled with artificial neural network. ***Expert Systems with Applications***, 37(9): 6587-6593.

Fortin V., Quarda T.B.M.J.,and Bobee, B. (1997). Comments on 'The use of artificial neural networks for the prediction of water quality parameters' by Maier H.R., and Dandy, G.C. ***Water Resources Research***, 33(10): 2423–2424.

Ganguly, A.R. and Bras, R.L. (2003). Distributed quantitative precipitation forecasting (DQPF) using information from radar and numerical weather prediction models. ***Journal of Hydrometeorology***, 4 (6): 1168-1180.

Golyandina, N., Nekrutkin, V., and Zhigljavsky, A. (2001). ***Analysis of Time Series Structure: SSA and the related techniques***. Boca Raton, Fla.: Chapman & Hall/CRC.

Guhathakurta, P. (2008). Long lead monsoon rainfall prediction for meteorological sub-divisions of India using

deterministic artificial neural network model. ***Meteorology and Atmospheric Physics***, 101 (1-2): 93-108.

Gunn, S.R. (1998). ***Support vector machines for classification and regression. Image, Speech and Intelligent Systems Tech. Rep***. University of Southampton, U.K.

Jain, A., and Srinivasulu, S., (2004). Development of effective and efficient rainfall-runoff models using integration of deterministic, real-coded genetic algorithms and artificial neural network techniques, ***Water Resour. Res.***, 40, W04302.

Jain, A., and Srinivasulu, S. (2006). Integrated approach to model decomposed flow hydrograph using artificial neural network and conceptual techniques. ***Journal of Hydrology***, 317, 291–306.

Jayawardena, A.W; Lai, F.Z. (1994). Analysis and Prediction of Chaos in Rainfall and Stream-Flow Time-Series. ***Journal of Hydrology***, 153 (1-4): 23-52.

Kecman, V. (2001). ***Learning and soft computing: support vector machines, neural networks, and fuzzy logic models***. Cambridge, Massachusetts: MIT press.

Kim, Y.O., Jeong, D., and Ko, I.H. (2006). Combining rainfall-runoff model outputs for improving ensemble streamflow prediction. ***Journal of Hydrologic Engineering***, 11 (6), 578-588.

Kişi, O. (2008). Constructing neural network sediment estimation models using a data-driven algorithm. ***Mathematics and Computers in Simulation***, 79, 94–103.

Kitanidis, P. K. and Bras, R. L. (1980). Real-time forecasting with a conceptual hydrologic model, 2, applications and results. ***Water Resource Research***, 16 (6), 1034–1044.

Lee, C.F., Lee, J.C. and Lee, A.C., (2000). ***Statistics for business and financial economics (2nd version)***. Singapore: World Scientific.

Legates, D.R., McCabe Jr., G.J. (1999). Evaluating the use of goodness-of-fit measures in hydrologic and hydroclimatic model validation. ***Water Resources Research***, 35 (1), 233–241.

Li, F., and Zeng, Q.C. (2008). Statistical prediction of East Asian summer monsoon rainfall based on SST and sea ice concentration. ***Journal of the meteorological society of Japan***, 86 (1), 237-243.

Marques, C.A.F., Ferreira, J., Rocha, A., Castanheira, J., Gonçalves, P., Vaz., N., and Dias, J.M. (2006). Singular spectral analysis and forecasting of hydrological time series. ***Physics and Chemistry of the Earth***, 31, 1172-1179.

Munot, A. A. and Kumar, K.K. (2007). Long range prediction of Indian summer monsoon rainfall. ***Journal of Earth System Science***, 116 (1), 73-79.

Muttil, N. and Chau, K.W., (2006). Neural network and genetic programming for modelling coastal algal blooms. ***Int. J. Environment and Pollution***, 28, 3/4, 223–238.

Nasseri, M., Asghari, K., and Abedini, M.J. (2008). Optimized scenario for rainfall forecasting using genetic algorithm coupled with artificial neural network. ***Expert Systems with Applications***, 35(3), 1415-1421.

Nayagam, L.R., Janardanan, R., and Mohan, H.S.R. (2008). An empirical model for the seasonal prediction of southwest monsoon rainfall over Kerala, a meteorological subdivision of India. ***International Journal of Climatology***, 28 (6), 823-831.

Newbold, P., Carlson, W. L., and Thorne, B.M. (2003). ***Statistics for business and economics (fifth version)***, Prentice Hall, Upper Saddle River, N.J.

Partal, T. and Kişi, Ö. (2007). Wavelet and Neuro-fuzzy conjunction model for precipitation forecasting. ***Journal of Hydrology***, 342, (2), 199-212.

Pongracz, R, Bartholy, J. and Bogardi, I. (2001). Fuzzy rule-based prediction of monthly precipitation. ***Physics and Chemistry of the Earth Part B-Hydrology Oceans and Atmosphere***, 26 (9), 663-667.

Sachindra, D.A., Huang, F., Barton, A.F., and Perera, B.J.C. (2011). Statistical downscaling of General

Circulation Model outputs to catchment streamflows ***MODSIM 2011 - 19th International Congress on Modelling and Simulation - Sustaining Our Future: Understanding and Living with Uncertainty***, 2810-2816.

Sarle WS. (1994). ***Neural networks and statistical models***. In Proceedings of the Nineteenth Annual SAS Users Group International Conference. SAS Institute: Cary, NC; 1538–1550.

See, L., and Openshaw, S. (2000). A hybrid multi-model approach to river level forecasting. ***Hydrological Sciences Journal***, 45 (3), 523–536.

Sedki, A., Ouazar, D., El Mazoudi, E. (2009). Evolving neural network using real coded genetic algorithm for daily rainfall–runoff forecasting. ***Expert Systems with Applications***, 36(3): 4523-4527.

Shamseldin, A. Y., O'Connor, K. M., and Liang, G. C. (1997). Methods for combining the outputs of different rainfall-runoff models. ***Journal of Hydrology***, 197, 203–229.

Shamseldin, A. Y., and O'Connor, K. M. (1999), A real-time combination method for the outputs of different rainfall-runoff models. ***Hydrological Sciences Journal*** 44 (6), 895-912.

Sheng, C., Gao, S., and Xue, M. (2006). Short-range prediction of a heavy precipitation event by assimilating Chinese CINRAD-SA radar reflectivity data using complex cloud analysis. ***Meteorology and Atmospheric Physics***, 94 (1-4), 167-183.

Shrestha, D. L. & Solomatine, D. P. (2006). Experiments with AdaBoostRT, an improved boosting scheme for regression. ***Neural Comput.***, 17.

Silverman, D., and Dracup, J.A. (2000). Artificial neural networks and long-range precipitation prediction in California. ***Journal of Applied Meteorology***, 39 (1), 57-66.

Sivapragasam, C., Liong, S.Y., and Pasha, M.F.K. (2001). Rainfall and discharge forecasting with SSA-SVM approach. ***Journal of Hydroinformatics***, 3(7), 141–152.

Sivapragasam, C., and Liong, S. Y. (2005). Flow categorization model for improving forecasting. ***Nordic Hydrology***, 36 (1), 37–48.

Solomatine, D. P., & Ostfeld, A. (2008). Data-driven modelling: Some past experiences and new approaches. ***Journal of Hydroinformatics***, 10(1), 3-22.

Solomatine, D. P. and Xue, Y. I. (2004). M5 model trees and neural networks: application to flood forecasting in the upper reach of the Huai River in China. ***Journal of Hydrological Engineering***, 9 (6), 491–501.

Sudheer, K.P., Gosain, A.K., and Ramasastri, K.S. (2002). A data-driven algorithm for constructing artificial neural network rainfall-runoff models. ***Hydrological Processes***, 16, 1325–1330.

Sudheer, K. P., Nayak, P. C. and Ramasastri, K. S. (2003). Improving Peak Flow Estimates in Artificial Neural Network River Flow Models. ***Hydrological Processes***, 17, 677-686.

Surajit, C., and Goutami, C. (2007). Identification of the best hidden layer size for three layered neural net in predicting monsoon rainfall in India. ***Journal of Hydroinformatics***, 10(2), 181-188.

Talei, A., Chua, L.H.C., Quek C. (2010). A novel application of a neuro-fuzzy computational technique in event-based rainfall–runoff modeling. ***Expert Systems with Applications***, 37(12), 7456-7468.

Toth, E, Brath, A. and Montanari, A. (2000). Comparison of short-term rainfall prediction models for real-time flood forecasting. ***Journal of Hydrology***, 239 (1-4), 132-147.

Vautard, R., Yiou, P. & Ghil, M. (1992). Singular-spectrum analysis: a toolkit for short, noisy and chaotic signals. ***Physica D***, 58, 95–126.

Venkatesan, C., Raskar, S.D., Tambe, S.S., Kulkarni, B.D., and Keshavamurty, R.N. (1997). Prediction of all India summer monsoon rainfall using error-back-propagation neural networks. ***Meteorology and Atmospheric Physics***, 62 (3-4), 225-240.

Wang, W., van Gelder, P.H.A.J.M., Vrijling, J.K., and Ma, J. (2006). Forecasting Daily Streamflow Using Hybrid ANN Models, ***Journal of Hydrology***, 324, 383-399.

Wu, C.L., Chau, K.W., and Li, Y.S. (2008). River stage prediction based on a distributed support vector regression. ***Journal of Hydrology***, 358, 96-111.

Xiong, L. H., Shamseldin, A. Y. and O'Connor, K. M. (2001). A non-linear combination of the forecasts of rainfall–runoff models by the first-order Takagi-Sugeno fuzzy system, ***Journal of Hydrology***, 245 (1–4), 196–217.

Yates, D.N., Warner, T.T., and Leavesley, G.H. (2000). Prediction of a flash flood in complex terrain. Part II: A comparison of flood discharge simulations using rainfall input from radar, a dynamic model, and an automated algorithmic system. ***Journal of Applied Meteorology***, 39 (6), 815-825.

Yu, P.S., Chen, S.T., and Chang, I.F. (2006). Support vector regression for real-time flood stage forecasting. ***Journal of hydrology***, 328,704-716.

Zhang, B., and Govindaraju, R. S. (2000). Prediction of watershed runoff using Bayesian concepts and modular neural networks. ***Water Resources Research***, 36(3), 753-762.

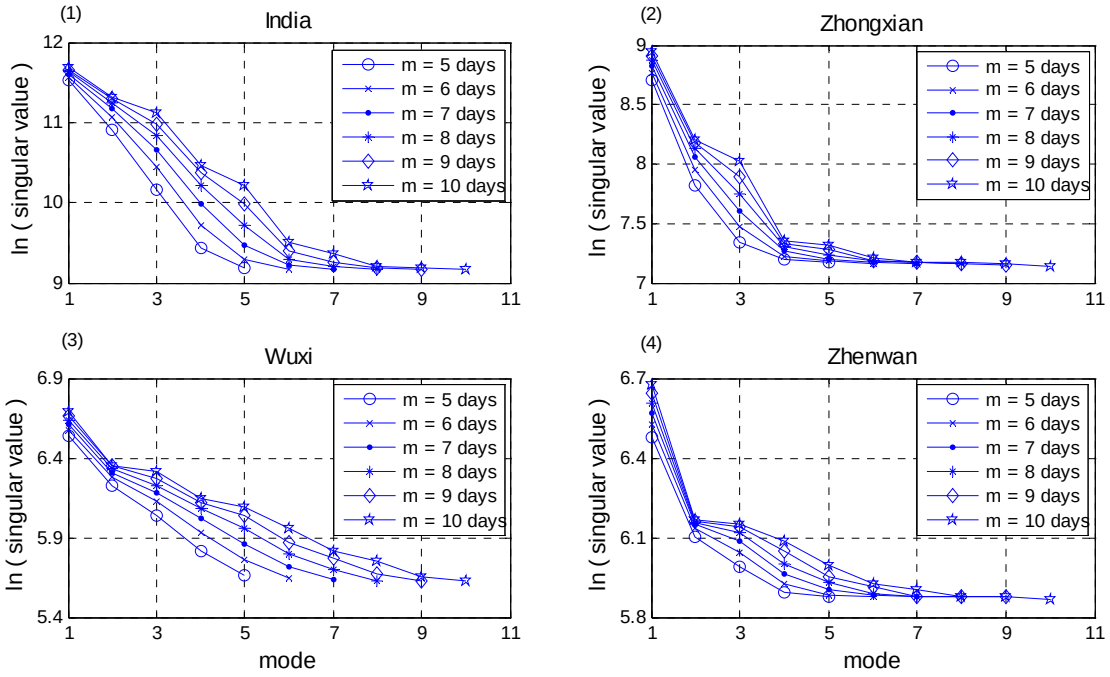

Figure 1. Singular Spectrum as a function of lag using various *m* for (1) India, (2) Zhongxian, (3) Wuxi and (4) Zhenwan

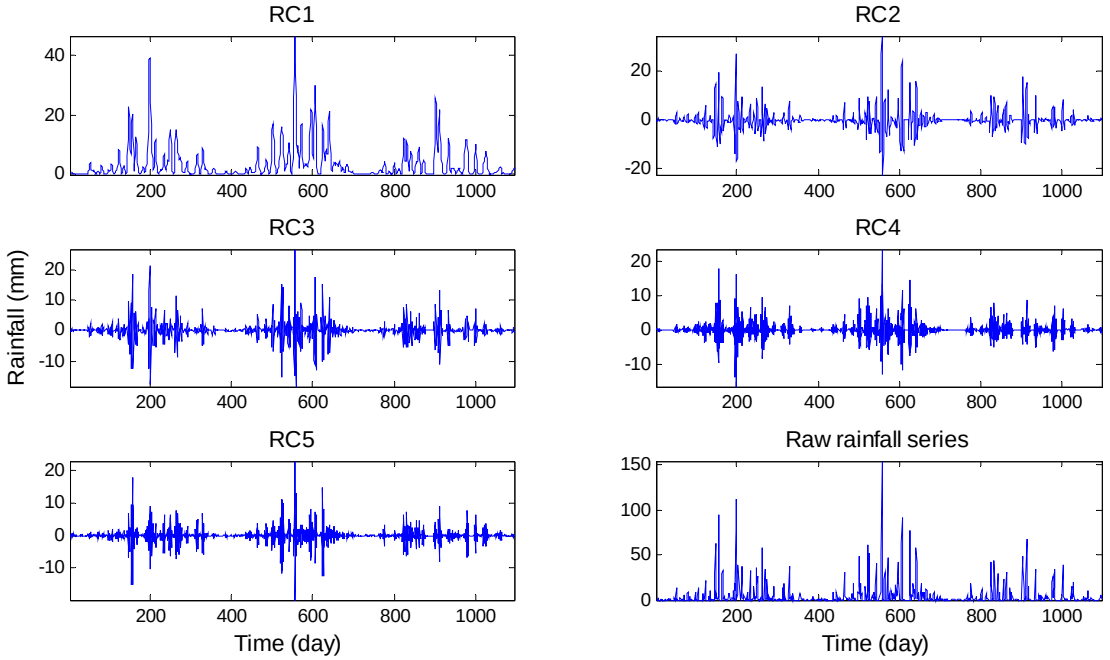

Figure 2. Reconstructed components (RCs) and raw rainfall series for Wuxi

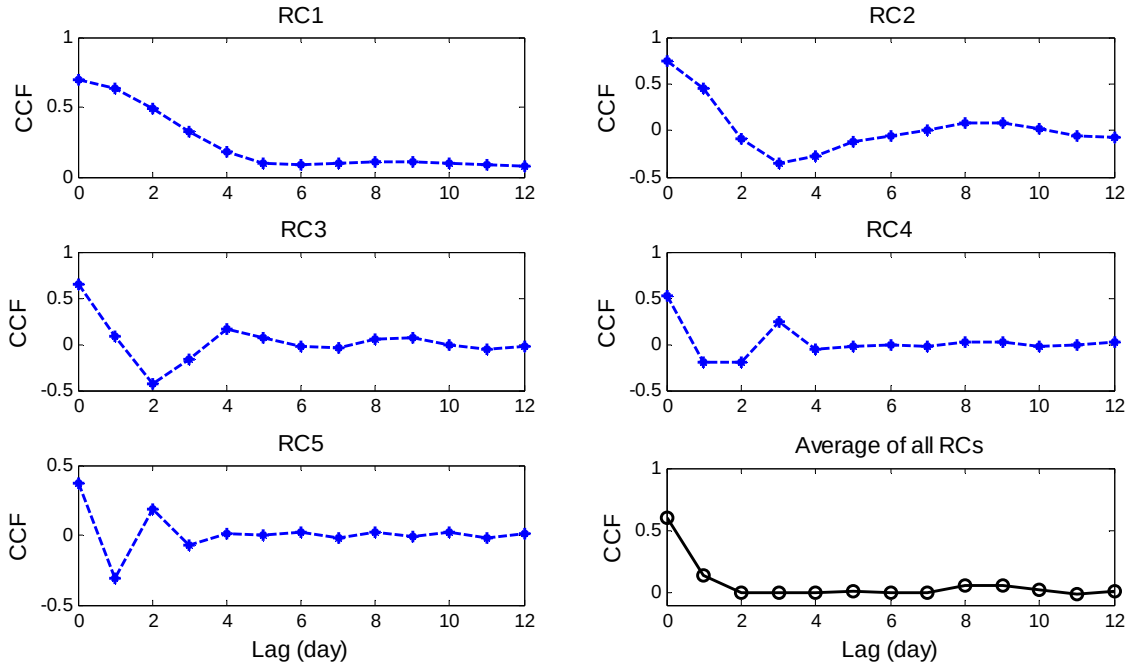

Figure 3. Plots of CCF between each RC and the raw rainfall data for Wuxi

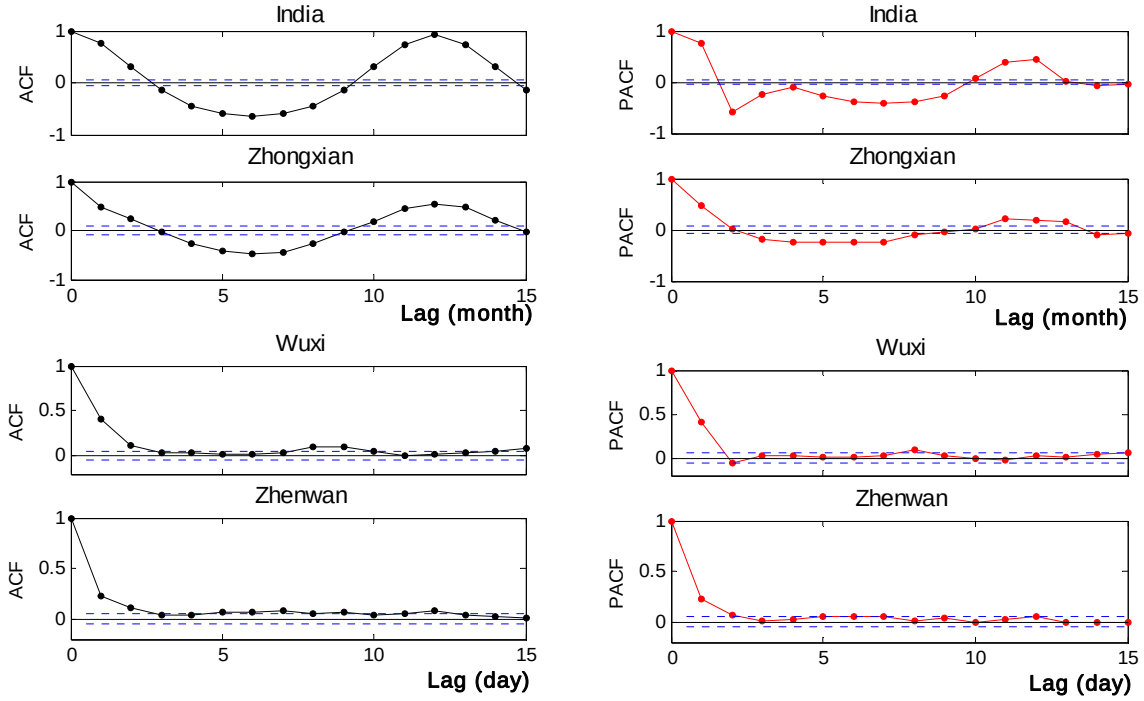

Figure 4. Plots of ACF and PACF of four rainfall series with the 95% confidence bounds denoted by the dashed lines (ACF at the left column and PACF at the right column)

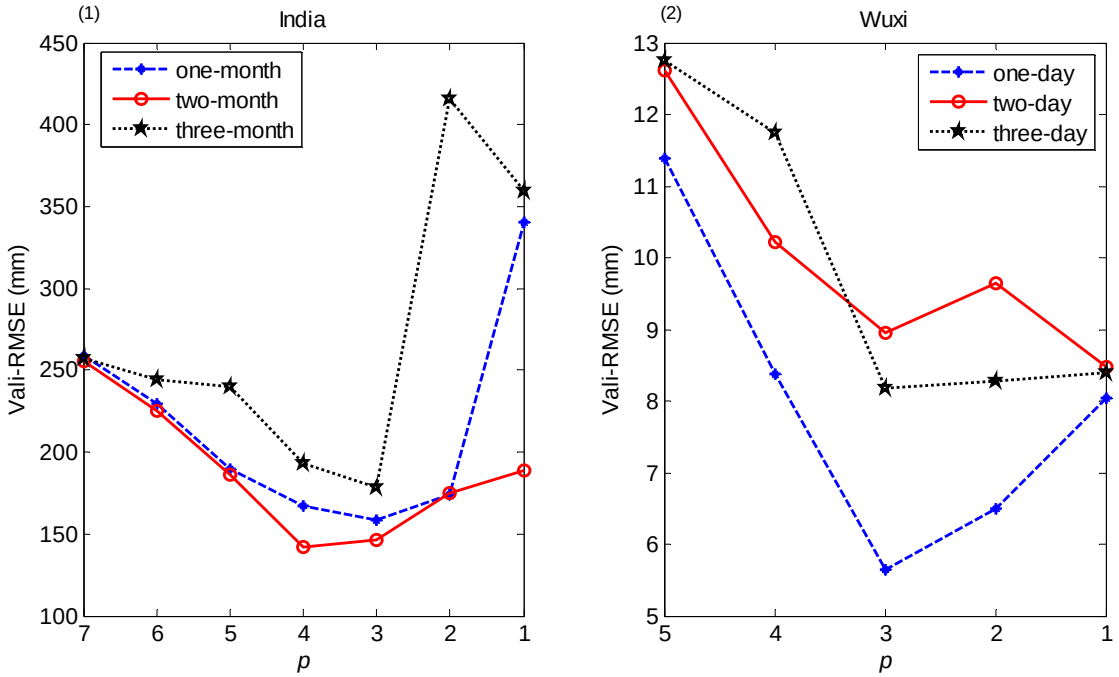

Figure 5. Performances of ANN-SSA as a function of p ($\leq m$) at various prediction horizons using (1) India rainfall data and (2) Wuxi rainfall data

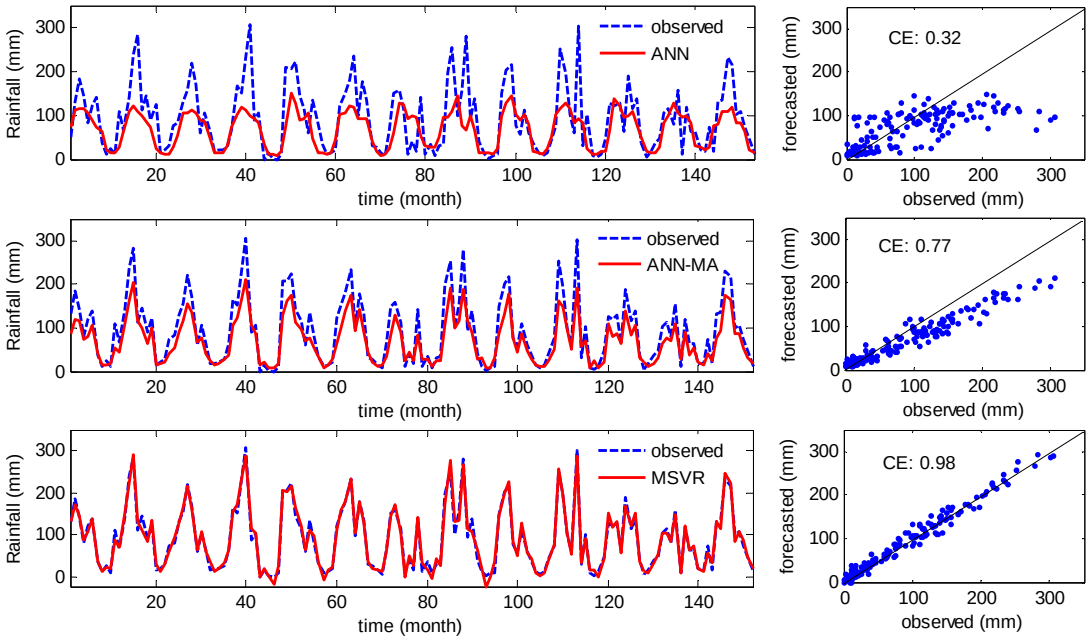

Figure 6. One-month lead predictions of ANN, ANN-SSA and MSVR for Zhongxian

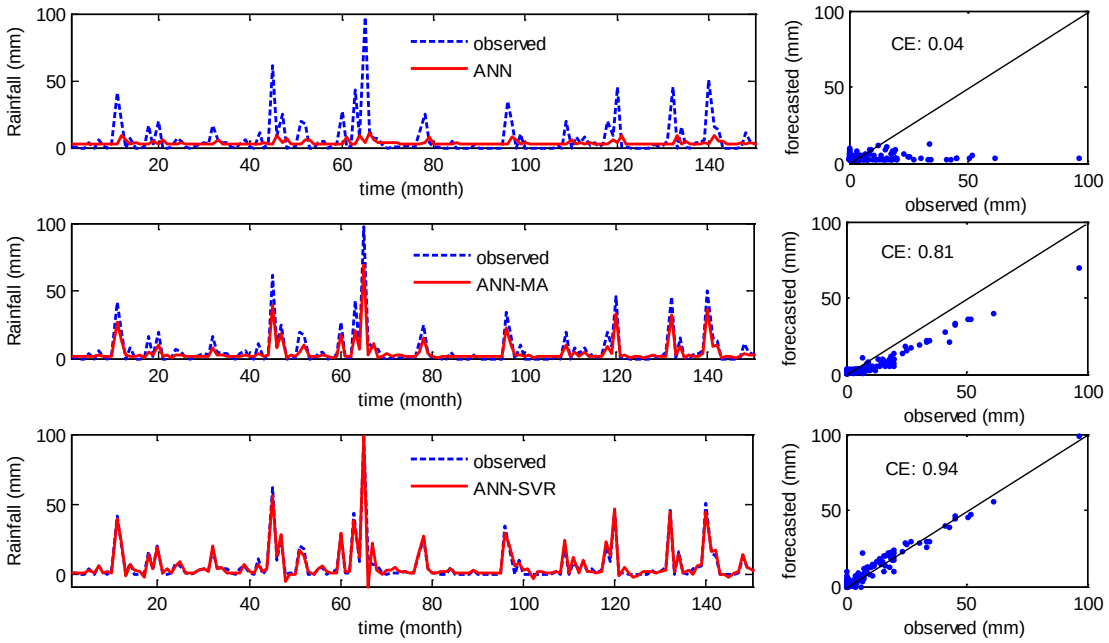

Figure 7. One-day lead predictions of ANN, ANN-SSA and ANN-SVR for Zhenwan

Table 1 Parameters in MA and SSA at various forecasting steps for four rainfall series

watershed	Model	*K* in MA			*p* from *m* RCs in SSA		
		1-step	2-step	3-step	1-step	2-step	3-step
India							
	ANN-MA	3	5	7			
	ANN-SSA				3/7*	4/7	3/7
	DSVR	3	5	7			
Zhongxian							
	ANN	3	5	7			
	ANN-SSA				2/6	2/6	1/6
	DSVR	3	5	7			
Wuxi							
	ANN-MA	3	5	7			
	ANN-SSA				3/5	1/5	3/5
	ANN-SVR	3	5	7			
Zhenwan							
	ANN	3	5	9			
	ANN-SSA				4/7	3/7	4/7
	ANN-SVR	3	5	9			

* Denominator stands for all RCs, and numerator represents the number of selected RCs.

Table 2 Comparison of model performances at various forecasting horizons using monthly rainfall data from India and Zhongxian

watershed	Model	RMSE			CE			PI			d		
		1	2	3*[a]	1	2	3	1	2	3	1	2	3
India													
	Persistence	643.1	1210.2	1483.8	0.52	-0.70	-1.55	0.00	-0.24	-0.14	0.24	0.20	0.18
	ANN	259.3	256.6	253.6	0.92	0.92	0.93	0.84	0.94	0.97	0.39	0.39	0.37
	ANN-SSA	156.6	141.0	177.5	0.97	0.98	0.96	0.94	0.98	0.98	0.42	0.42	0.41
	ANN-MA	69.2	80.9	99.9	0.99	0.99	0.99	0.99	0.99	0.99	0.43	0.43	0.43
	MSVR*[b]	48.7	55.1	76.6	1.00	1.00	0.99	0.99	0.99	0.99	0.44	0.43	0.42
Zhongxian													
	Persistence	75.7	93.6	110.0	-0.03	-0.56	-1.14	0.00	-0.04	-0.03	0.06	0.06	0.06
	ANN	61.6	75.9	76.5	0.32	-0.02	-0.04	0.35	0.31	0.50	0.15	0.14	0.13
	ANN-SSA	55.7	57.1	49.1	0.45	0.42	0.57	0.46	0.42	0.79	0.21	0.20	0.19
	ANN-MA	35.6	54.9	54.7	0.77	0.46	0.47	0.78	0.64	0.74	0.34	0.30	0.31
	MSVR*[b]	11.6	24.4	21.4	0.98	0.89	0.92	0.98	0.90	0.92	0.42	0.40	0.41

*[a] The number of '1, 2, and 3' denote one-, two-, and three-day-ahead forecasts; *[b] MSVR was coupled with MA.

Table 3 Comparison of model performances at various forecasting horizons using daily rainfall data from Wuxi and Zhenwan

watershed	Model	RMSE			CE			PI			d		
		1	2	3*[a]	1	2	3	1	2	3	1	2	3
Wuxi													
	Persistence	12.2	15.1	15.0	0.05	-0.43	-0.40	0.00	0.11	0.18	0.16	0.15	0.16
	ANN	11.5	12.6	12.7	0.17	0.01	-0.01	0.12	0.38	0.41	0.24	0.23	0.23
	ANN-SSA	5.8	8.5	8.5	0.79	0.55	0.54	0.77	0.72	0.73	0.36	0.28	0.30
	ANN-MA	5.3	5.7	6.4	0.82	0.80	0.74	0.81	0.87	0.85	0.37	0.36	0.34
	ANN-SVR*[b]	3.3	5.0	5.1	0.93	0.84	0.84	0.94	0.91	0.91	0.40	0.38	0.39
Zhenwan													
	Persistence	12.4	11.4	11.8	-0.53	-0.28	-0.37	0.00	0.14	0.26	0.11	0.13	0.14
	ANN	9.8	10.0	10.1	0.04	-0.01	-0.01	0.37	0.33	0.45	0.19	0.17	0.20
	ANN-SSA	7.2	6.7	6.2	0.48	0.55	0.62	0.66	0.70	0.79	0.23	0.24	0.27
	ANN-MA	4.4	5.2	4.4	0.81	0.73	0.81	0.81	0.82	0.90	0.36	0.32	0.35
	ANN-SVR*[b]	2.4	3.8	3.1	0.94	0.86	0.91	0.96	0.91	0.94	0.41	0.37	0.38

*[a] The number of '1, 2, and 3' denote one-, two-, and three-day-ahead forecasts; *[b] ANN-SVR was coupled with MA.

1
2

3 Table 4 Model performances at various data preprocessing methods using the ANN model

watershed	Data preprocessing	RMSE			CE			PI			d		
		1	2	3*[a]	1	2	3	1	2	3	1	2	3
India													
	Resc_raw	259.3	256.6	253.6	0.92	0.92	0.93	0.84	0.94	0.97	0.39	0.39	0.40
	Resc_nth_root	259.1	254.9	252.8	0.92	0.92	0.93	0.84	0.94	0.97	0.39	0.40	0.41
	Std_raw	259.2	254.1	252.8	0.92	0.92	0.93	0.84	0.95	0.97	0.39	0.41	0.41
	Std_nth_root	265.0	254.8	253.0	0.92	0.92	0.93	0.83	0.94	0.97	0.40	0.40	0.41
Zhongxian													
	Resc_raw	61.6	75.9	76.5	0.32	-0.02	-0.04	0.35	0.31	0.50	0.12	0.12	0.13
	Resc_nth_root	62.8	76.4	75.2	0.30	-0.04	0.00	0.32	0.30	0.52	0.13	0.14	0.14
	Std_raw	52.4	52.7	53.2	0.51	0.51	0.50	0.53	0.67	0.76	0.18	0.19	0.22
	Std_nth_root	52.3	55.0	54.2	0.51	0.46	0.48	0.53	0.64	0.75	0.19	0.21	0.23
Wuxi													
	Resc_raw	11.5	12.6	12.7	0.17	0.01	-0.01	0.12	0.38	0.41	0.17	0.18	0.19
	Resc_nth_root	11.5	13.5	13.3	0.16	-0.15	-0.11	0.12	0.29	0.36	0.17	0.20	0.19
	Std_raw	11.5	13.3	13.0	0.17	-0.10	-0.06	0.13	0.31	0.38	0.23	0.19	0.21
	Std_nth_root	11.9	13.2	13.2	0.11	-0.10	-0.10	0.06	0.32	0.36	0.21	0.22	0.22
Zhenwan													
	Resc_raw	9.8	10.0	10.1	0.04	-0.01	-0.01	0.37	0.33	0.45	0.16	0.16	0.20
	Resc_nth_root	10.3	10.6	10.7	-0.06	-0.11	-0.13	0.31	0.25	0.38	0.18	0.16	0.18
	Std_raw	9.9	10.6	10.6	0.03	-0.12	-0.10	0.37	0.25	0.40	0.21	0.16	0.21
	Std_nth_root	10.3	10.6	10.7	-0.05	-0.11	-0.13	0.32	0.25	0.38	0.18	0.17	0.19

4 *[a] The number of '1, 2, and 3' denote one-, two-, and three-day-ahead forecasts.

Table 5 Statistical parameters of data transformation

watershed	Methods of data transformation	Skewness	Kurtosis	Data length
India				
	No treatment	0.9	2.5	Jan.,1871-Dec.,2006
	Moving average	0.7	2.0	
	Cubic root	0.3	1.8	
Zhongxian				
	No treatment	1.2	5.6	Jan.,1956-Dec.,2007
	Moving average	0.5	2.6	
	Square root	0.2	2.4	
Wuxi				
	No treatment	5.8	49.7	Jan. 1,1995-Dec. 31,1998
	Moving average	4.4	29.8	
	Fifth root	0.6	2.3	
Zhenwan				
	No treatment	5.2	47.2	Jan. 1,2004-Dec. 31,2007
	Moving average	2.8	12.8	
	Fifth root	0.6	2.0	

Table 6 Performances of the ANN using combined data-preprocessing method of SSA and MA and relevant parameters

watershed	Methods of data preprocessing	RMSE	CE	PI	d	*p* from *m* in SSA	*K* in MA
India							
	MA	69.2	0.99	0.99	0.43		3
	SSA+MA	104.4	0.99	0.97	0.41	3/7*	7
Zhongxian							
	MA	35.6	0.77	0.78	0.34		3
	SSA+MA	44.9	0.64	0.65	0.33	2/6	5
Wuxi							
	MA	5.3	0.82	0.81	0.38		3
	SSA+MA	5.8	0.79	0.77	0.37	3/5	1
Zhenwan							
	MA	4.4	0.81	0.81	0.37		3
	SSA+MA	5.6	0.69	0.80	0.35	4/7	3

* Denominator stands for all RCs, and numerator represents the number of selected RCs.