Water Resour Manage (2007) 21:495–513
DOI 10.1007/s11269-006-9026-2

Potential assessment of the support vector regression technique in rainfall forecasting

Wei-Chiang Hong · Ping-Feng Pai

Received: 3 May 2005 / Accepted: 22 February 2006

Abstract Forecasting and monitoring of rainfall values are increasingly important for decreasing economic loss caused by flash floods. Based on statistical learning theory, support vector regression (SVR) has been used to deal with forecasting problems. Performing structural risk minimization rather than minimizing the training errors, SVR algorithms have better generalization ability than the conventional artificial neural networks. The objective of this investigation is to examine the feasibility and applicability of SVR in forecasting volumes of rainfall during typhoon seasons. In addition, Simulated Annealing (SA) algorithms are employed to choose parameters of the SVR model. Subsequently, rainfall values during typhoon periods in Taiwan's Wu–Tu watershed are used to demonstrate the forecasting performance of the proposed model. The simulation results show that the proposed SVRSA model is a promising alternative in forecasting amounts of rainfall during typhoon seasons.

Keywords Rainfall forecasting · Support vector regression · Simulated annealing algorithms · Water resources

1. Introduction

In Taiwan, heavy rains in the summer are typically associated with the southwest monsoon, monsoon trough and tropical cyclones. Unanticipated flash floods, particularly with high peak discharges produced by severe rainfalls, are the most destructive natural hazard threatening human lives and properties. To provide correct and prompt information containing the spatial and temporal distribution of rainfalls is the best way to avoid loss of lives and properties (Brath *et al.*, 1988; Lettenmaier and Wood, 1993). However, rainfall is one of the most

W.-C. Hong
School of Management, Da-Yeh University, 112 Shan-Jiau Road, Da-Tusen, Changhua, 51505, Taiwan

P.-F. Pai (✉)
Department of Information Management, National Chi Nan University, 1 University Rd., Puli, Nantou, 545, Taiwan
e-mail: paipf@ncnu.edu.tw

difficult meteorological variables in the hydrologic cycle. The formation mechanism and forecast of rainfall involve a rather complex physics that has not been completely understood so far (Andrieu *et al.*, 1996; Bustamante *et al.*, 1999; French *et al.*, 1992). To forecast rainfall accurately is a challenge for hydrologists and water resource engineers.

Generally, rainfall forecast technologies can be classified into two categories, namely the physical modeling approach and the pattern recognition methodology (Luk *et al.*, 2001). The physical modeling approaches are designed to approximate the physical mechanism of hydrologic processes. In addition, the physical modeling approaches usually offer the possibility of identifying processes or improving our knowledge in a specific catchment. These approaches are applied particularly when the hydrological data appear non-stationary (Druce, 2001; Kitanidis and Bras, 1980; Vieux *et al.*, 2004). However, the physical modeling approaches for rainfall forecasting may not be feasible due to the significant calibration data both in space and time (Yapo *et al.*, 1996; Luk *et al.*, 2001).

The pattern recognition methodology formulates the relationships between inputs and outputs without considering the physical structure processes. Developed by Box and Jenkins (1976), the autoregressive moving average with exogenous inputs (ARMAX) models have been one of the most popular approaches to time series forecasting. ARMAX models have been applied in hydrologic forecasting and obtained satisfactory performances in the past two decades (Bras and Rodriguez-Iturbe, 1985; Salas et al, 1980). The artificial neural network (ANN) is regarded as a powerful mathematical model for achieving the forecast of rainfall volume in time series and has become popular in recent years. French *et al.* (1992) employed ANN to build a rainfall simulation model and provided accurate rainfall information. Luk *et al.* (2000) employed ANN to forecast the temporal and spatial distribution of short-term rainfall for an urban catchment. Several ANN models with different orders of lag and spatial inputs were presented to compare the forecast accuracy. Luk *et al.* (2001) compared the performance of three ANN models, namely multilayer feedforward neural network, Elman's recurrent neural network and time-delay neural network, in forecasting the spatial distribution of rainfall for an urban catchment. They reported that the model with lower lag can yield more accurate forecast results. Valverde Ramírez *et al.* (2005) developed a feed-forward neural network with resilient propagation learning algorithm to forecast daily rainfall in São Paulo region, Brazil. The numerical results indicated that the proposed model is superior to a multiple linear regression model in terms of forecasting accuracy indices. Pan and Wang (2004) employed the state space neural network (SSNN) model to forecast short term rainfall-runoff in Taiwan's Wu-Tu watershed. The SSNN was used to strengthen the linkage between weights and networks as well as to interchange the network states into a time-variant model. Empirical results revealed that the SSNN model is suitable for hydrological forecasts. Lin and Chen (2004) applied the radial basis function network (RBFN) to rainfall-runoff forecasting in Taiwan's Fei-Tsui reservoir watershed. They claimed that the RBFN model is feasible in explaining relationships between rainfall and runoff. Chiang *et al.* (2004) used back propagation neural networks, trained by traditional conjugate gradient (CG) algorithm and real-time recurrent learning (RTRL) algorithm, respectively, to forecast the rainfall-runoff in Taiwan's Lan-Yang River. The CG algorithm is used for static networks and the RTRL algorithm is for dynamic networks. Their investigation indicated that RTRL algorithms with the capability of updating dynamic data continuously outperformed CG algorithms. Castellano-Méndez *et al.* (2004) presented a multilayer perceptron neural network with one hidden layer to model the behavior of the runoff. The simulation results showed that the proposed neural network had more accurate forecasting results than the Box-Jenkins models.

Support vector machines (SVM) were originally developed to solve pattern recognition and classification problems. With the introduction of Vapnik's ε-insensitive loss function, support vector machines have been extended to solve nonlinear regression estimation problems and have been successfully employed to solve forecasting problems in many fields. Tay and Cao (2001) applied SVM to forecasting financial time series. Their numerical results demonstrated that SVM is superior to a multi-layer back-propagation neural network in financial time series forecasting. Wang *et al.* (2003) used SVM to predict air quality. They reported that SVM outperforms conventional radial basis function networks. Cao and Gu (2002) presented a dynamic SVM model for solving non-stationary time series problems. Their experimental results indicated that DSVM outperforms standard SVM in forecasting non-stationary time series. In the same year, Tay and Cao (2002) developed a C-ascending SVM to model non-stationary financial time series. Their experimental results showed that the C-ascending SVM with sample data of actual orders consistently outperform the standard SVM. Cao (2003) applied SVM experts to forecasting time series. Generalized SVM experts have a two-stage neural network architecture. The numerical results demonstrated that SVM experts can achieve better generalization than single SVM models. Mohandes *et al.* (2004) used SVM to predict wind speed. Their experimental results indicated that the SVM model outperformed multilayer perceptron neural networks as measured by root mean square errors. Pai and Lin (2004a) used SVM to forecast production values of the machinery industry in Taiwan. They reported that SVM performed better than the seasonal autoregressive integrated moving average model and the general regression neural network model. Pai and Lin (2004b) proposed a hybrid model with the strength of an autoregressive integrated moving average model and the SVM model for forecasting the stock prices. By using 10 stocks to examine the performance, numerical results show that the proposed hybrid model provided more accurate forecasting results than the autoregressive integrated moving average model and random walk model.

The selection of the three parameters (σ, C, and ε) in a SVR model influences the forecasting accuracy significantly. Owing to a lack of structured ways in determining the three free parameters in the SVR model, the simulated annealing algorithms are employed to determine the values of the three parameters in the SVR model. In this investigation, a SVR model with SA is proposed to forecast rainfall of Wu-Tu Watershed in northern Taiwan. The rest of the paper is organized as follows. Section 2 introduces the basic concept of the SVR model. Section 3 presents the SA that is used to select the parameters of the SVR model. Three forecast models for comparing the forecasting performance with the proposed model are addressed in Section four. Section 5 illustrates a numerical example to demonstrate the forecasting performance of the proposed models. Conclusions are finally made in Section 6.

2. Support vector regression

The basic concept of the support vector regression is to map nonlinearly the original data x into a higher dimensional feature space. Hence, given a set of data $G = \{(x_i, a_i)\}_{i=1}^{N}$ (where x_i is the input vector; a_i is the actual value, and N is the total number of data patterns), the SVR function is

$$f = g(x) = w\psi(x_i) + b \tag{1}$$

where $\psi(x_i)$ is the feature of inputs, and both w and b are coefficients. The coefficients (w and b) are estimated by minimizing the following regularized risk function;

$$r(f) = C\frac{1}{N}\sum_{i=1}^{N}\Gamma_\varepsilon(a_i, f_i) + \frac{1}{2}\|w\|^2 \tag{2}$$

where

$$\Gamma_\varepsilon(a_i, f_i) = \begin{cases} 0 & \text{if } |a_i - f_i| \leq \varepsilon \\ |a_i - f_i| - \varepsilon & \text{otherwise} \end{cases} \tag{3}$$

and C and ε are prescribed parameters. In Equation (2), $\Gamma_\varepsilon(a_i, f_i)$ is called the ε-insensitive loss function. The loss equals zero if the forecasted value is within the ε-tube (Equation (3)). The second term, $\frac{1}{2}\,\|w\|^2$, measures the flatness (function curvature) of the function. Therefore, C is considered to specify the trade-off between the empirical risk and the model flatness. Both C and ε are user-determined parameters. Two positive slack variables ξ and ξ^*, which represent the distance from actual values to the corresponding boundary values of ε-tube, respectively are introduced. Then, Equation (2) is transformed into the following constrained form;

Minimize

$$r(w, \xi, \xi^*) = \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{N}(\xi_i + \xi_i^*)\right) \tag{4}$$

with the constraints,

$$\begin{aligned} w\psi(x_i) + b - a_i &\leq \varepsilon + \xi_i^*, \quad i = 1, 2 \ldots, N \\ a_i - w\psi(x_i) - b &\leq \varepsilon + \xi_i, \quad i = 1, 2 \ldots, N \\ \xi_i, \xi_i^* &\geq 0, \quad i = 1, 2 \ldots, N \end{aligned}$$

This constrained optimization problem is solved using the following primal Lagrangian form:

$$\begin{aligned} &L(w, b, \xi, \xi^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*) \\ &= \frac{1}{2}\,\|w\|^2 + C\left(\sum_{i=1}^{N}(\xi_i + \xi_i^*)\right) - \sum_{i=1}^{N}\beta_i[w\psi(x_i) + b - a_i + \varepsilon + \xi_i] \\ &- \sum_{i=1}^{N}\beta_i^*[a_i - w\psi(x_i) - b + \varepsilon + \xi_i^*] - \sum_{i=1}^{N}\left(\alpha_i\xi_i + \alpha_i^*\xi_i^*\right) \end{aligned} \tag{5}$$

Equation (5) is minimized with respect to primal variables w, b, ξ and ξ^*, and maximized with respect to nonnegative Lagrangian multipliers α_i, α_i^*, β_i and β_i^*, Therefore, Equations. (6)–(9) are obtained.

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{N}(\beta_i - \beta_i^*)\psi(x_i) = 0 \tag{6}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{N}(\beta_i^* - \beta_i) = 0 \tag{7}$$

$$\frac{\partial L}{\partial \xi_i} = C - \beta_i - \alpha_i = 0 \tag{8}$$

$$\frac{\partial L}{\partial \xi_i^*} = C - \beta_i^* - \alpha_i^* = 0 \tag{9}$$

Then, Karush–Kuhn–Tucker conditions are applied to the regression, and Equation (4) thus yields the dual Lagrangian by substituting Equations (6)–(9) into Equation (5), more detailed calculation is shown as Equation (10).

$$\begin{aligned}
L &= \frac{1}{2}\,\|w\|^2 + C\left(\sum_{i=1}^{N}(\xi_i+\xi_i^*)\right) - w\sum_{i=1}^{N}\beta_i\psi(x_i) - b\sum_{i=1}^{N}\beta_i + \sum_{i=1}^{N}\beta_i a_i - \varepsilon\sum_{i=1}^{N}\beta_i \\
&\quad -\sum_{i=1}^{N}\beta_i\xi_i - \sum_{i=1}^{N}\beta_i^* a_i + w\sum_{i=1}^{N}\beta_i^*\psi(x_i) + b\sum_{i=1}^{N}\beta_i^* - \varepsilon\sum_{i=1}^{N}\beta_i^* \\
&\quad -\sum_{i=1}^{N}\beta_i^*\xi_i^* - \sum_{i=1}^{N}\alpha_i\xi_i - \sum_{i=1}^{N}\alpha_i^*\xi_i^* \\
&= \frac{1}{2}\,\|w\|^2 + C\left(\sum_{i=1}^{N}(\xi_i+\xi_i^*)\right) - w\left(\sum_{i=1}^{N}\psi(x_i)(\beta_i-\beta_i^*)\right) \\
&\quad -\varepsilon\sum_{i=1}^{N}(\beta_i+\beta_i^*) + \sum_{i=1}^{N}a_i(\beta_i-\beta_i^*) + b\sum_{i=1}^{N}(\beta_i^*-\beta_i) - \sum_{i=1}^{N}\beta_i\xi_i - \sum_{i=1}^{N}\alpha_i\xi_i \\
&\quad -\sum_{i=1}^{N}\beta_i^*\xi_i^* - \sum_{i=1}^{N}\alpha_i^*\xi_i^* \\
&= \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{N}(\xi_i+\xi_i^*)\right) - w\cdot w - \varepsilon\sum_{i=1}^{N}(\beta_i+\beta_i^*) \\
&\quad +\sum_{i=1}^{N}a_i(\beta_i-\beta_i^*) - \sum_{i=1}^{N}(C-\alpha_i)\xi_i - \sum_{i=1}^{N}\alpha_i\xi_i - \sum_{i=1}^{N}(C-\alpha_i^*)\xi_i^* - \sum_{i=1}^{N}\alpha_i^*\xi_i^* \\
&= -\frac{1}{2}\,\|w\|^2 + C\left(\sum_{i=1}^{N}(\xi_i+\xi_i^*)\right) - \varepsilon\sum_{i=1}^{N}(\beta_i+\beta_i^*) + \sum_{i=1}^{N}a_i(\beta_i-\beta_i^*) \\
&\quad -\sum_{i=1}^{N}C\xi_i - \sum_{i=1}^{N}C\xi_i^* \\
&= -\frac{1}{2}\,\|w\|^2 + C\left(\sum_{i=1}^{N}(\xi_i+\xi_i^*)\right) - \varepsilon\sum_{i=1}^{N}(\beta_i+\beta_i^*) + \sum_{i=1}^{N}a_i(\beta_i-\beta_i^*) \\
&\quad -C\left(\sum_{i=1}^{N}(\xi_i+\xi_i^*)\right) \\
&= -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i-\alpha_i^*)(\beta_i-\beta_i^*)\psi(x_i)\psi(x_j) - \varepsilon\sum_{i=1}^{N}(\beta_i+\beta_i^*) + \sum_{i=1}^{N}a_i(\beta_i-\beta_i^*)
\end{aligned} \tag{10}$$

Then, the dual Lagrangian, Equation (11), is obtained when kernel function is $K(x_i, x_j) = \psi(x_i)\psi(x_j)$,

$$\vartheta(\beta_i, \beta_i^*) = \sum_{i=1}^{N} a_i(\beta_i - \beta_i^*) - \varepsilon \sum_{i=1}^{N} (\beta_i + \beta_i^*) - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\beta_i - \beta_i^*)(\beta_j - \beta_j^*) K(x_i, x_j) \tag{11}$$

subject to the constraints,

$$\sum_{i=1}^{N} (\beta_i - \beta_i^*) = 0$$

$$0 \le \beta_i \le C, \quad i = 1, 2, \cdots, N$$

$$0 \le \beta_i^* \le C, \quad i = 1, 2, \cdots, N$$

The Lagrange multipliers in Equation (11) satisfy the equality $\beta_i * \beta_i^* = 0$. The Lagrange multipliers β_i and β_i^*, are calculated and an optimal desired weight vector of the regression hyperplane is,

$$w^* = \sum_{i=1}^{N} (\beta_i - \beta_i^*) \psi(x_i) \tag{12}$$

Finally, Equation (12) is then substituted into Equation (1), and the regression function is obtained as Equation (13).

$$g(x, \beta, \beta^*) = \sum_{i=1}^{N} (\beta_i - \beta_i^*) K(x, x_i) + b \tag{13}$$

Here, $K(x_i, x_j)$ is called the Kernel function. The value of the Kernel equals the inner product of two vectors, x_i and x_j , in the feature space $\psi(x_i)$ and $\psi(x_j)$, respectively; that is, $K(x_i, x_j) = \psi(x_i) * \psi(x_j)$. Any function that meets Mercer's condition (Vapink, 1995) can be used as the Kernel function. In this work, the Gaussian function, $\exp(-\frac{1}{2} * (\frac{\|x_i - x_j\|}{\sigma})^2)$, is used in the SVR model.

The selection of the three parameters, σ, ε and C, of a SVR model influence the accuracy of forecasting. However, structural methods for confirming efficient selection of parameters efficiently are lacking. Therefore, simulated annealing algorithm (SA) is used in the proposed SVR model to optimize parameter selection.

3. Simulated annealing algorithms in selecting parameters of the SVR model

The simulated annealing algorithm is an optimization technique, analogous to the annealing process of material physics. Boltzmann (Cercignani, 1988) pointed out if the system is in thermal equilibrium at a temperature T, then the probability $P_T(s)$ of the system being in a given state s is given by the Boltzmann distribution:

$$P_T(s) = \frac{\exp(-E(s)/kT)}{\sum_{w \in S} \exp(-E(w)/kT)} \tag{14}$$

where $E(s)$ denotes the energy of state s (The state is defined as the SVR forecasting error in each iteration here); k represents the Boltzmann constant and S is the set of all possible states. However, Equation (14) does not contain information on how a fluid reaches thermal equilibrium at a given temperature. Metropolis *et al.* (1953) developed an algorithm that simulates the process of Boltzmann. The Metropolis algorithm is summarized as follows. When the system is in original state s_{old} with energy $E(s_{\text{old}})$, a randomly selected atom is perturbed, resulting in a state s_{new} with energy$E(s_{\text{new}})$. This new state is either accepted or rejected depending on the Metropolis criterion: if $E(s_{\text{new}}) \leq E(s_{\text{old}})$, then the new sate is automatically accepted; in contrast, if $E(s_{\text{new}}) > E(s_{\text{old}})$, then the probability of accepting the new state is given by the following probability function.

$$P(\text{accept } s_{\text{new}}) = \exp\left(-\frac{E(s_{\text{old}}) - E(s_{\text{new}})}{kT}\right) \tag{15}$$

According to the studies of Kirkpatrick *et al.* (1983), it is claimed that the Metropolis approach is conducted for each temperature on the annealing schedule until thermal equilibrium is reached. Additionally, a prerequisite for applying SA is that a given set of the multiple variables defines a unique system state, for which the objective function can be calculated.

4. SA algorithm procedure and flowchart

The procedure of SA algorithm is described as follows and the flowchart is shown as Figure 1.

Step 1 (Initialization): Set upper bounds of the three SVR positive parameters, σ, C, and ε. Then, generate and feed the initial values of the three parameters into the SVR model. The forecasting error is defined as the system state (*E*). Here, the initial state (E_0) is obtained.

Step 2 (Provisional state): Make a random move to change the existing system state to a provisional state. Another set of three positive parameters is generated in this stage.

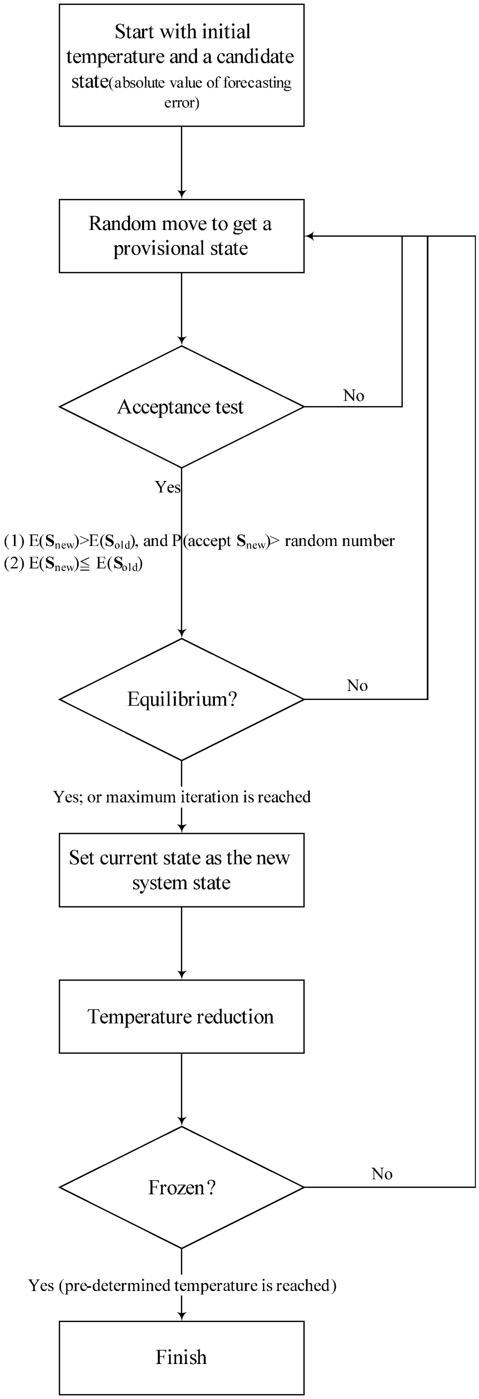
Step 3 (Acceptance tests): The following equation is employed to determine the acceptance or rejection of the provisional state (Metropolis *et al.*, 1953).

$$\begin{cases} \text{Accept the provisional state, if} E(s_{\text{new}}) > E(s_{\text{old}}), \text{ and } \quad p < P(\text{accept } s_{\text{new}}), \; 0 \leq p \leq 1. \\ \text{Accept the provisional state, } \textit{if } E(s_{\text{new}}) \leq E(s_{\text{old}}) \\ \text{Reject the provisional state, otherwise} \end{cases} \tag{16}$$

In Equation (16), the p is a random number for determining the acceptance of the provisional state. If the provisional state is accepted, then set the provisional state as the current state.

Step 4 (Incumbent solutions): If the provisional state is not accepted, then return to Step 2. Furthermore, if the current state is not superior to the system state, then repeat Steps 2 and 3 until the current state is superior to the system state, and set the current state as the new system state. Previous studies (Kirkpatrick *et al.*, 1983; Van Laarhoven and Aarts, 1987) indicated that the maximum number of loops (Nsa) is 100d to avoid infinitely repeated loops, where d denotes the problem dimension. In this investigation, three parameters (σ, C, and ε) are used to determine the system states. Therefore, Nsa is set to be 300.

Fig. 1 Simulated annealing flowchart


Start with initial temperature and a candidate state(absolute value of forecasting error)
Random move to get a provisional state
Acceptance test
No
Yes
(1) $E(\mathbf{S}_{new}) > E(\mathbf{S}_{old})$, and P(accept $\mathbf{S}_{new}$)> random number
(2) $E(\mathbf{S}_{new}) \leqq E(\mathbf{S}_{old})$
Equilibrium?
No
Yes; or maximum iteration is reached
Set current state as the new system state
Temperature reduction
Frozen?
No
Yes (pre-determined temperature is reached)
Finish

Step 5 (Temperature reduction): After the new system state is obtained, reduce the temperature. The new temperature reduction is obtained by the Equation (17):

$$\text{New temperature} = (\text{Current temperature}\ ) \times \rho, \quad \text{where } 0 < \rho < 1 \tag{17}$$

(17) ρ is set to be 0.9 in this study (Dekkers and Aarts, 1991). If the pre-determined temperature is reached, then stop the algorithm and the latest state is an approximate optimal solution. Otherwise, go to Step 2.

In the investigation, the normalized mean squared error measure (NMSE), shown as Equation (18), serves as the criterion for identifying suitable parameters for use in the SVRSA model

$$\text{NMSE} = \frac{1}{n\delta^2} \sum_{i=1}^{n} (a_i - f_i)^2 \tag{18}$$

where $\delta^2 = \frac{1}{n-1} \sum_{i=1}^{n} (a_i - \bar{a})^2$; n is the number of forecasting periods; a_i is the actual rainfall depth value at period i; $\bar{a}$ denotes the mean of the actual rainfall depth value; and f_i is the forecasting rainfall depth value of Wu-Tu watershed at period i.

5. Models for comparing forecast performance

In this investigation three models are employed for comparing the forecasting accuracy with the proposed SVRSA model. The first model used is the Holt-Winters (HW) method (Holt, 1957; Winter, 1960). The Holt-Winters method is an extension of exponentially weighted moving average procedure. The exponentially weighted moving average approach forecasts future values based on past observations, and places more weight on the recent observations. The Holt-Winters method smoothes the trend values separately with two smoothing coefficients (with values between 0 and 1) and incorporates an explicit linear trend in the forecast. The approach of Holt -Winter linear exponential smoothing is as follows:

$$s_t = \alpha a_t + (1 - \alpha)(s_{t-1} + b_{t-1}) \tag{19}$$

$$b_t = \beta(s_t - s_{t-1}) + (1 - \beta) b_{t-1} \tag{20}$$

$$f_t = s_t + i b_t \tag{21}$$

where a_t is the actual value at time t; s_t is the smoothed estimate at time t; b_t is the trend value at time t; α is the level smoothing coefficient; and β is the trend smoothing coefficient.

Equation (19) lets the actual value be smoothed in a recursive manner by weighting the current level (α), and then adjusts s_t directly for the trend of the previous period, b_{t-1}, by adding it to the last smoothed value, s_{t-1}. This helps to eliminate the lag and brings s_t to the approximate base of the current data value.

Equation (20) updates the trend, which is expressed as the difference between the last two smoothed values. It modifies the trend by smoothing with β in the last period $(s_t - s_{t-1})$, and adding that to the previous estimate of the trend multiplied by $(1 - \beta)$.

Equation (21) is used to forecast ahead. The trend, b_t, is multiplied by the number of periods ahead to be forecast, i, and added to the base value, s_t. The forecast error (e_t) is defined as the actual value minus the forecast (fitted) value for time period t, that is:

$$e_t = a_t - f_t \tag{22}$$

The forecast error is assumed to be an independent random variable with zero mean and constant variance. Values of smoothing coefficients, α and β, are determined to minimize the forecast NMSE.

The second model used for comparing the forecast accuracy is the seasonal Holt and Winters' linear exponential smoothing (SHW) approach. The Holt-Winters method can not be extended to accommodate additive seasonality if the magnitude of the seasonal effects does not change with the series, or multiplicative seasonality if the amplitude of the seasonal pattern changes over time. The forecast for seasonal Holt and Winters' linear exponential smoothing is as follows:

$$s_t = \alpha \frac{a_t}{I_{t-L}} + (1-\alpha)(s_{t-1} + b_{t-1}) \tag{23}$$

$$b_t = \beta(s_t - s_{t-1}) + (1-\beta)b_{t-1} \tag{24}$$

$$I_t = \gamma \frac{a_t}{s_t} + (1-\gamma)I_{t-L} \tag{25}$$

$$f_t = (s_t + ib_t)I_{t-L+i} \tag{26}$$

where L is the length of seasonality; I is the seasonal adjustment factor; and γ is the seasonal adjustment coefficient.

Equation (23) differs slightly from Equation (24) in that the first term is divided by the seasonal numberI_{t-L}; this is done to deseasonalize a_t (eliminate seasonal fluctuations from a_t). Equation (25) is comparable to a seasonal index that is found as a ratio of current values of the series, a_t, divided by the smoothed value for the series, s_t. If a_t is larger than s_t, the ratio will be greater than 1, else, the ratio will be less than 1. in order to smooth the randomness of a_t, Equation (25) weights the newly computed seasonal factor with γ and the most recent seasonal number corresponding to the same season with $(1-\gamma)$.

The third mode is the recurrent neural network (RNN) model. The main concept of recurrent neural networks is that links may be established within the layers of any feed-forward neural network. In the RNN model, every unit is considered as the output of the network and provides the adjusted information as input in furthermore training process (Kechriotis *et al.*, 1994). RNN models are widely used in time series forecasting. Elman (1990), Jordan (1987), and Williams and Zipser (1989) had developed three different RNN models to solve nonlinear adaptive filtering, pattern recognition, and forecasting problems. In this investigation, the Elman network is adopted as a recurrent neural network framework. The Elman network is constructed as a regular feed-forward network (Elman, 1990). Figure 2 show the architecture of an Elman network. All neurons in one layer are connected with all neurons in the next layer except the context layer. A context layer is special case of a hidden layer. Interactions only happen between neurons of the hidden layer and the context layer. For an Elman network with P inputs and H hidden neurons, the output of the n th neuron,$f_n(t)$, is (Ayaz *et al.*, 2003; Connor *et al.*, 1994; Gencay and Liu, 1997; Mandic and Chambers, 2001):

$$f_n(t) = \sum_{i=1}^{H} W_i \varphi(t) + b_i(t) \tag{27}$$

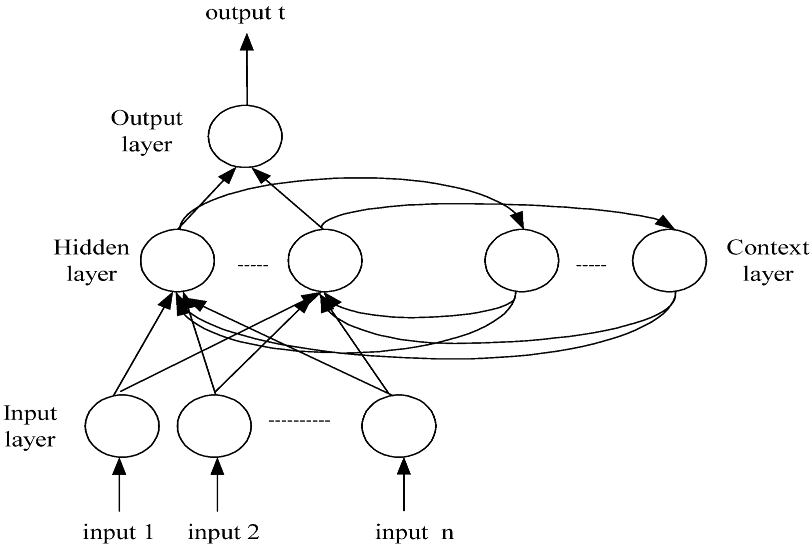


Fig. 2 Elman RNNs (Elman, 1990)

where W_i are weights between the hidden and the output layer; $\varphi(t)$ is the output function of hidden neurons, which is computed as

$$\varphi_i(t) = g\left(\sum_{j=1}^{P} v_{ij} x_j(t) + \sum_{k=1}^{q} \sum_{m=1}^{H} w_{\mathrm{ikm}} \varphi_m(t-k) + b_i(t)\right) \tag{28}$$

where $v_{ij}$are weights between the input and the hidden layer; $w_{\mathrm{ikm}}$are weights between the context and the hidden layer with delay k periods; q is the total numbers of the context layer of past output data.

Back-propagation is a procedure to obtain gradients for adapting weights of a neural network. Back-propagation algorithm presents as follows. First, the output of the n th neuron in Equation (27) is rewritten as

$$f_n(t) = h(x^T(t)\,\phi\,(t)) \tag{29}$$

where $h(\cdot)$ is nonlinearity function of $x^T(t)$and $f_n(t)$; $x^T(t) = [x_1(t), \ldots, x_P(t)]^T$ is the input vector; $\phi(t) = [\phi_1(t), \ldots \phi_P(t)]^T$ is the weight vector, then, a cost function is proposed to be the instantaneous performance index,

$$J(\phi(t)) = \frac{1}{2}\left[d(t) - f_n(t)\right]^2 = \frac{1}{2}\left[d(t) - h(x^T(t)\phi(t))\right]^2 \tag{30}$$

where $d(t) = [d_1(t), \ldots, d_P(t)]^T$ is the desired output.

Secondly, the instantaneous output error at the output neuron and revised weight vector in the next moment are presented as Equations (31) and (32) respectively.

$$e(t) = d(t) - f_n(t))) = d(t) - h(x^T(t)\phi(t))) \tag{31}$$

$$\phi(t+1) = \phi(t) - \eta\nabla_\phi J(\phi(t))) \tag{32}$$

where η is the learning rate.

Third, the gradient $\nabla_\phi J(\phi(t))$ can be calculated as

$$\nabla_\phi J(\phi(t)) = \frac{\partial J(\phi(t))}{\partial \phi(t)} = e(t) \times \frac{\partial e(t)}{\partial \phi(t)} = -e(t)h'(x^T(t)\phi(t))x(t) \tag{33}$$

where $h'(\cdot)$ is the first derivation of the nonlinearity $h(\cdot)$. Finally, the weight is revised as

$$\phi(t+1) = \phi(t) + \eta e(t)h'(x^T(t)\phi(t))x(t) \tag{34}$$

Finally, by substituting Equation (34) into Equation (29), the output value of the n-th neuron is obtained.

6. Experimental setup and simulation results

The rainfall data of Wu-Tu watershed were used as a case study for developing rainfall forecasting model in this investigation. The Wu-Tu watershed is located in northern Taiwan and its size is about 204 km^2. Within the watershed, the annual average rainfall is about 2,500 mm. There are three telemetric rain gauges (Wu-Tu, Jui-Fang, and Huo-Shao-Liao) and one discharge site (Wu-Tu) within the Wu-Tu Watershed. Locations of these gauges are shown in Figure 3. Hourly volumes of rainfall (from August 1985 to August 1997) of the three rain gauges are served as experimental data in this study. During this period, nine typhoon events occurred. In this investigation, the hourly volumes of rainfall brought by typhoons Nelson, Abby, and Sarah were employed as training data set. The validation data set included the hourly volumes of rainfall brought by typhoons Ruth, Polly, and Seth. The hourly volumes of rainfall brought by the other three typhoons served as the testing data set. Table 1 shows the experimental data used in this investigation.

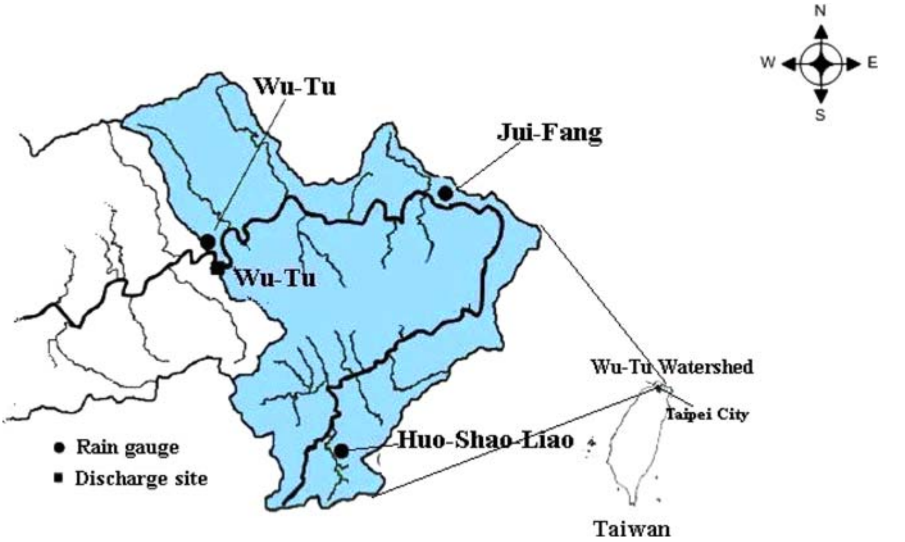


Fig. 3 The Wu-Tu watershed in northern Taiwan

Table 1 Information of typhoon selected from Wu-Tu discharge site

Number	Name	Date (yyyy/mm/dd)	Rainfall duration(h)	Rainfall depth (mm)	Remark
1	Nelson	1985/08/22	41	341.4	Training data set
2	Abby	1986/09/17	88	521.3	
3	Sarah	1989/09/10	71	322.5	
4	Ruth	1991/10/28	72	448.6	Validation data set
5	Polly	1992/08/29	85	500.6	
6	Seth	1994/10/09	49	300.7	
7	Herb	1996/07/31	43	313.6	Testing data set
8	Zane	1996/09/27	77	440.6	
9	Winnie	1997/08/17	47	343.5	

6.1. Measurement of forecasting accuracy

The normalized mean squared error (NMSE), as given by Equation (18), was used as the measurement of forecasting accuracy. Additionally, the accurate efficiency was measured by the coefficient of efficiency (CE) and the coefficient of correlation (CC), given by Equation (35) and Equation (36) respectively. The values of NMSE indicate the deviations between actual values and forecast values. The maximum values of CE and CC are one. The CE measures the efficiency of the forecasting model. The larger CE means the forecasting model is more efficient. The higher values of CC indicate that the proposed model can capture the tendency of the forecasting data.

$$\mathrm{CE} = 1 - \frac{\sum_{i=1}^{n} (a_i - f_i)^2}{\sum_{i=1}^{n} (a_i - \bar{a})^2} \tag{35}$$

$$\mathrm{CC} = \frac{\sum_{i=1}^{n} (a_i - \bar{a})\left(f_i - \bar{f}\right)}{\sqrt{\sum_{i=1}^{n} (a_i - \bar{a})^2 * \sum_{i=1}^{n} \left(f_i - \bar{f}\right)^2}} \tag{36}$$

where a_i and f_i represent the actual and forecast volumes of rainfall, respectively; $\bar{a}$ and$\bar{f}$represent the actual and forecast mean volumes of rainfall, respectively; and n is the number of forecasting periods.

6.2. Parameter determination of different models and forecasting results

In this investigation, a rolling-based forecasting procedure was conducted and a one-hour-ahead forecasting policy was adopted. Based on the forecasting policy, several types of data-rolling are considered as a time series to feed into the SVRSA model in for forecasting rainfall depth in the next hour. In the training stage, the rainfall data contains three typhoon events, the number of rainfall data fed into SVRSA model is designed by considering the following two issues: the capture of rainfall data pattern from each typhoon event, and prevention of the over-fitting problem. Hence, the number of rainfall data fed into SVRSA model equals to the total rainfall hours of the previous two typhoons (129 rainfall data) or one typhoon (41 rainfall data). Table 2 shows parameters of SVRSA models and forecast results. It is illustrated that feeding SVRSA model by 129 rainfall data provides a smaller NMSE value in the testing stage. Figures 4–6 compare the actual rainfall values and one-hour ahead forecasting values for training data, validation data and testing data, respectively.

Table 2 Forecasting results and associated parameters of the SVRSA model

Number of input data	Parameters σ	C	ε	NMSE of training	NMSE of validation	NMSE of testing
41*	0.35515	17.230	0.65167	1.1870	0.7032	0.9948
129**	0.05206	28.867	0.87811	1.1529	0.4112	0.3963

*: Including the total rainfall durations of one typhoon event, Nelson, in training stage.
**Including the total rainfall durations of two typhoon events, Nelson and Abby, in training stage.

Table 3 Forecasting measurements and parameters of four models

	Parameters	NMSE	CE	CC
SVRSA model	$\sigma = 0.05206, \quad C= 28.867\ \varepsilon = 0.87811$	0.3963	0.6037	0.7988
HW model	$\alpha = 0.45, \beta$=0.035	0.4758	0.5242	0.7459
SHW model	$L = 41, \alpha$=0.51 $\beta = 0.03, \quad \gamma = 0.1$	0.4633	0.5367	0.7892
RNN model	Number of nodes in the hidden layer = 3	0.4435	0.5538	0.8937

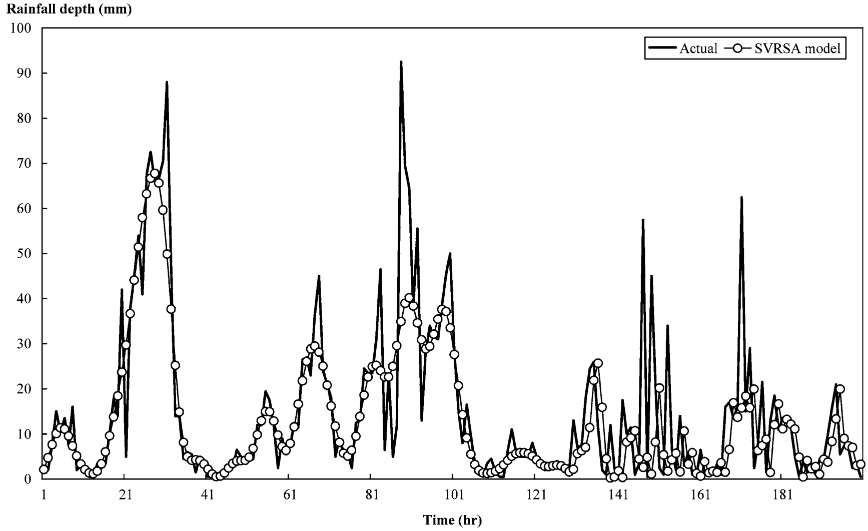


Fig. 4 Actual and forecasting amounts of rainfall for training data (SVRSA model)

For the Holt-Winters method, the α value and β value are 0.45 and 0.035 respectively. The appropriate parameters for the SHW models are 41, 0.51, 0.03, and 0.1 for L, α, β and γ correspondingly. The number of nodes in the hidden layer is used as a validation parameter of the RNN models. The most suitable number of hidden nodes of a RNN model is three. Figures 7–9 show the one-hour ahead forecasting rainfall values of the HW model, the SHW model, and the RNN model respectively. For the SVRSA model, parameters resulting in the minimum validation NMSE value were selected as the most suitable model for the present example.

Table 3 lists the forecasting measurements and suitable parameters for the four models and illustrates the forecasting accuracy and efficiency of the four models in terms of

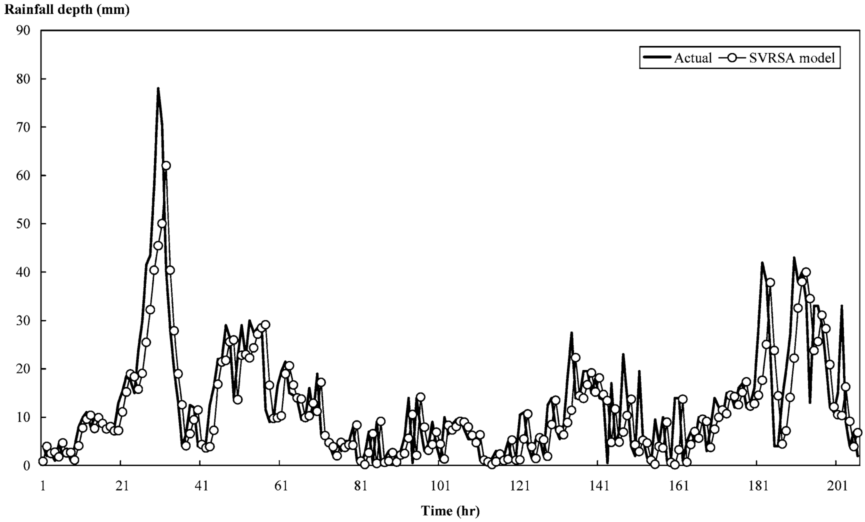


Fig. 5 Actual and forecasting amounts of rainfall for validation data (SVRSA model)

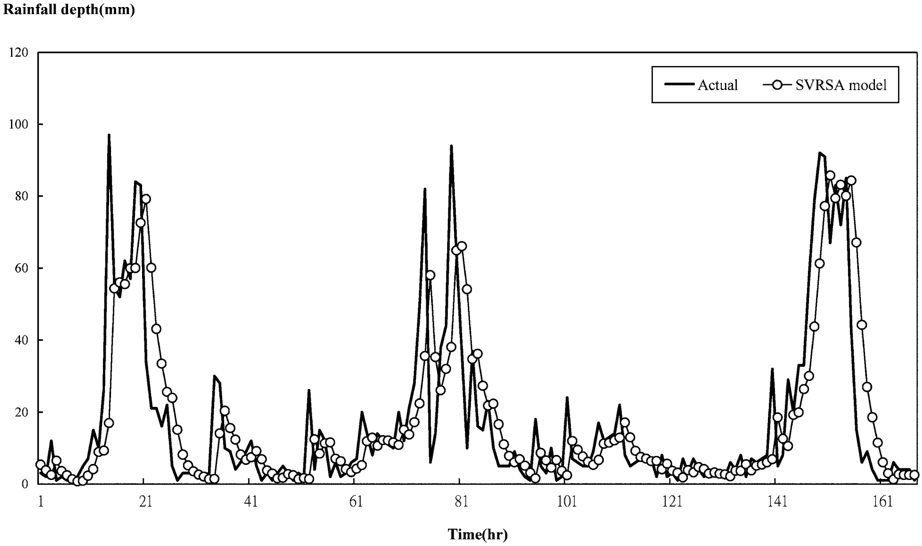


Fig. 6 Actual and forecasting amounts of rainfall for testing data (SVRSA model)

various evaluation indices. For NMSE accuracy index, the proposed SVRSA model with satisfactory forecasting performance and is capable to be employed to forecast rainfall depth during typhoon period. Similarly, for CE efficiency index, the proposed SVRSA model is also deserved to be confident. For CC efficiency index, the forecasting rainfall depth values from SVRSA model have higher correlation relationship with actual rainfall depth values than both HW and SHW models. However, RNN model has a higher

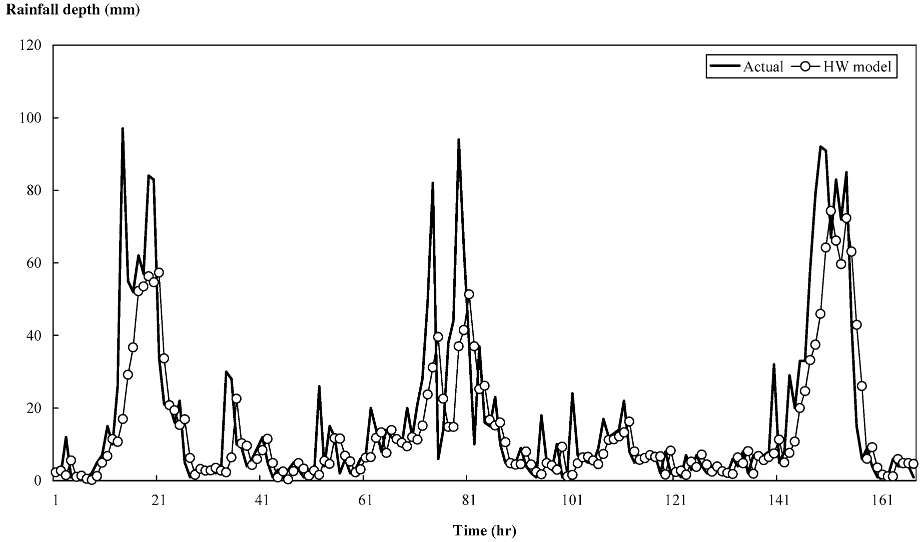


Fig. 7 Actual and forecasting amounts of rainfall for testing data (HW model)

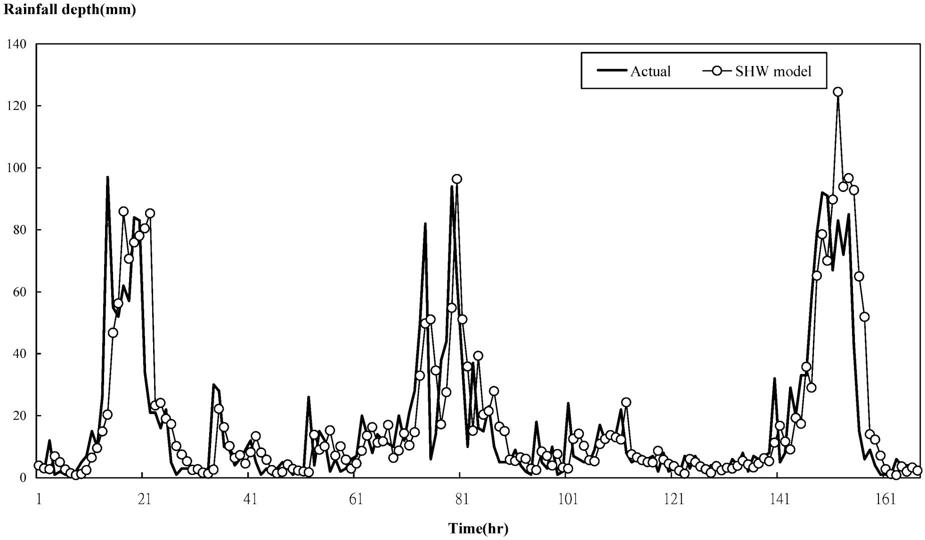


Fig. 8 Actual and forecasting amounts of rainfall for testing data (SHW model)

CC than the SVRSA model. In addition, it is observed that SVRSA model can capture the data pattern of rainfall during the peak periods. However, the other three models can not follow the data pattern successfully. Therefore, the nonlinear mapping ability and the proper selection of SVR parameters make the SVRSA successful in rainfall forecasting.

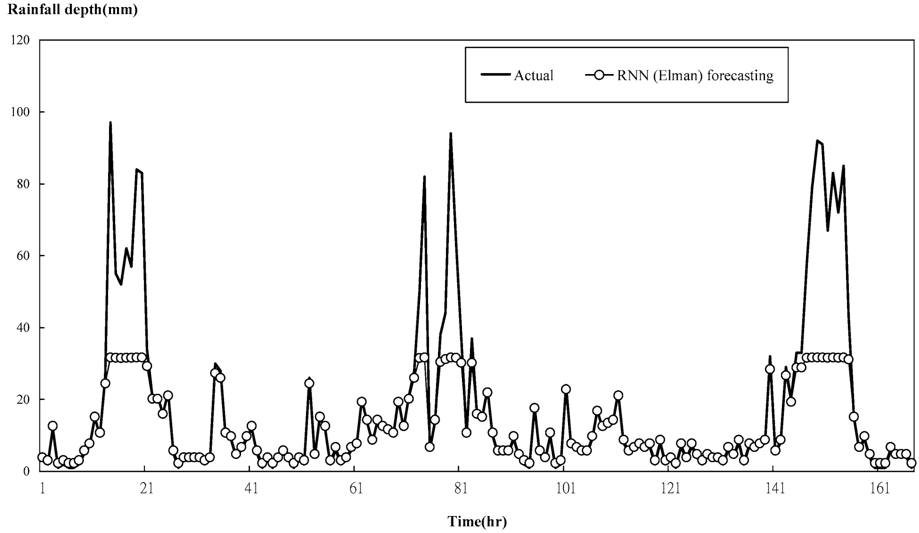


Fig. 9 Actual and forecasting amounts of rainfall for testing data (RNN model)

7. Conclusions

To avoid loss of human lives and properties, accurate rainfall forecasting is crucial for a tropical country with frequent-unanticipated flash floods, like Taiwan. The historical rainfall data of Wu-Tu watershed in northern Taiwan show a fluctuating trend, particularly during typhoon seasons. Although this is a common phenomenon in tropical countries, over-prediction or under-prediction amounts of rainfall influence a lot the social capability and costs in precaution against flash floods. This study introduces a novel forecasting technique, SVRSA, for forecasting amounts of rainfall during typhoon seasons in northern Taiwan. The experimental results reveal that the SVRSA model is a promising alternative in forecasting amounts of rainfall. Reasons for the superior performance of the SVRSA model can be as follows. First, the SVR conducts structural risk minimization rather than minimizing the training errors. Therefore, the SVR model has good generalization ability. Second, simulated annealing algorithms can select the three parameters of the SVRSA model properly to improve the forecasting accuracy.

This investigation demonstrates that the proposed SVRSA model offers a valid alternative for application in hydrology. In the future, more other meteorological variables during typhoon seasons, such as atmospheric pressure, temperature, convection condition as well as wind speed and direction, can be considered in the SVRSA model for forecasting amounts of rainfall. In addition, some other optimization techniques, such as tabu search algorithms, ant search algorithms, immune algorithms, particle swarm optimization algorithms, can be employed to select the SVR parameters. Furthermore, the relation between the forecasting performance by using different optimization techniques in determining SVR parameters and the types of forecasting data will be a challenging issue for future study.

Acknowledgements This research was conducted with the support of National Science Council (NSC 93-2213-E-212- 001& NSC 93-2745-H-212-001-URD). The rainfall data was provided by the Water Resource Agency, Ministry of Economic Affairs, Taiwan, R.O.C.

References

Andrieu H, French MN, Thauvin V, Krajewski WF (1996), Adaptation and application of a quantitative rainfall forecasting model in a mountainous region. J Hydrology 184:243–259

Ayaz, E, Seker, S, Barutcu, B, Türkcan E (2003) Comparisons between the various types of neural networks with the data of wide range operational conditions of the Borssele NPP, Progress in Nuclear Energy 43:381–387

Box, GEP, Jenkins GM (1976) Time series analysis: Forecasting and control, Holden-Day, San Francisco

Bras, RL, Rodriguez-Iturbe I (1985) Random Functions and Hydrology, Addison-Wesley, Massachusetts

Brath, A, Burlando, P, Rosso R (1988) Sensitivity analysis of real-time flood forecasting to on-line rainfall predictions. In Siccardi F, Bras RL (eds), Selected Papers from the Workshop on Natural Disasters in European- Mediterranean Countries, Perugia, Italy, pp 469–488

Bustamante J, Gomes J, Chou S, Rozante J (1999) Evaluation of April 1999 Rainfall Forecasts Over South America Using the ETA Model Climanalise, Divulgação Científica, Cachoeira Paulista, Brazil

Cao L (2003) Support vector machines experts for time series forecasting. Neurocomputing 51:321–339

Cao L, Gu Q (2002) Dynamic support vector machines for non-stationary time series forecasting. Intelligent Data Analysis 6:67–83

Castellano-Méndez M, González-Manteiga W, Febrero-Bande M, Prada-Sánchez JM, Lozano-Calderón R (2004) Modelling of the monthly and daily behaviour of the runoff of the Xallas river using Box-Jenkins and neural networks methods. J hydrology 296:38–58

Cercignani C (1988) The Boltzmann equation and its applications. Springer-Verlag, Berlin

Chiang YM, Chang LC, Chang FJ (2004) Comparison of static-feedforward and dynamic- feedback neural networks for rainfall-runoff modeling, J Hydrology 290:297–311

Connor JT, Martin RD, Atlas LE (1994) Recurrent neural networks and robust time series prediction. IEEE Transactions on Neural Networks 5:240–254

Dekkers A, Aarts EHL (1991) Global optimization and simulated annealing, Mathematical Programming. 50:367–393

Druce DJ (2001) Insights from a history of seasonal inflow forecasting with a conceptual hydrologic model. J Hydrology 249:102–112

Elman JL (1990) Finding structure in time, Cognitive Science 14:179–211

French MN, Krajewski WF, Cuykendall RR (1992) Rainfall forecasting in space and time using a neural network. J Hydrology 137:1–31

Gencay R, Liu T (1997) Nonlinear modeling and prediction with feedforward and recurrent networks. Physica D, 108:119–134

Holt CC (1957) Forecasting Seasonal and Trends by Exponentially Weighted Averages, Carnegie Institute of Technology, Pittsburgh PA

Jordan MI (1987) Attractor dynamics and parallelism in a connectionist sequential machine, Proceeding of 8th Annual Conference of the Cognitive Science Society, Hillsdale, pp 531–546

Kechriotis G, Zervas E, Manolakos ES (1994) Using recurrent neural networks for adaptive communication channel equalization. IEEE Transaction on Neural Networks 5:267–278

Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220:671–680

Kitanidis PK, Bras RL (1980) Real-time forecasting with a conceptual hydrologic model. Water Resources Research 16:1034–1044

Lin GF, Chen LH (2004) A non-linear rainfall-runoff model using radial basis function network. J Hydrology 289:1–8

Lettenmaier DP, Wood EF (1993) Hydrology forecasting. In Maidment DR (ed), Handbook of Hydrology, McGraw-Hill, New York

Luk KC, Ball JE, Sharma A (2000) A study of optimal model lag and spatial inputs to artificial neural network for rainfall forecasting. Journal of Hydrology 227: 56–65

Luk KC, Ball JE, Sharma A (2001) An application of artificial neural networks for rainfall forecasting. Mathematical and Computer Modelling 33:683–693

Mandic DP, Chambers JA (2001) Recurrent Neural Networks for Prediction, John Wiley and Sons. New York

Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH (1953) Equations of state calculations by fast computing machines. J Chemical Physics 21:1087–1092

Mohandes MA, Halawani TO, Rehman S, Hussain AA (2004) Support vector machines for wind speed prediction. Renewable Energy 29:939–947

Pai PF, Lin CS (2004a) Using support vector machines in forecasting production values of machinery industry in Taiwan. International Journal of Advanced Manufacturing Technology in press, DOI: 10.1007/s00170-004-2139-y

Pai PF, Lin CS (2004b) A hybrid ARIMA and support vector machines model in stock price forecasting. OMEGA-International Journal of Management Science 33:497–505

Pan TY, Wang RY (2004) State space neural networks for short term rainfall-runoff forecasting. J Hydrology 297:34–50

Salas JD, Delleur JW, Yevjevich V, Lane WL (1980) Applied Modeling of Hydrologic Time Series. Water Resources Publication, Littleton, Colo

Tay FEH, Cao L (2001) Application of support vector machines in financial time series forecasting, OMEGA-International Journal of Management Science 29:309–317

Tay FEH, Cao L (2002) Modified support vector machines in financial time series forecasting. Neurocomputing 48:847–861

Valverde Ramírez MC, de Campos Velho HF, Ferreira NJ (2005) Artificial neural network technique for rainfall forecasting applied to the São Paulo region. Journal of Hydrology 301:146-162

Van Laarhoven, PJM, Aarts EHL (1987) Simulated annealing: Theory and Applications, Kluwer Academic Publishers, Dordrecht

Vapnik, V (1995) The Nature of Statistical Learning Theory, Springer-Verlag, New York

Vieux, BE, Cui, Z, Gaur A (2004) Evaluation of a physics-based distributed hydrologic model for flood forecasting, Journal of Hydrology 298:155–177

Vojislav K (2001) Learning and Soft Computing–Support Vector Machines. Neural Networks and Fuzzy Logic Models, The MIT Press, Massachusetts

Wang W, Xu Z, Lu JW (2003) Three improved neural network models for air quality forecasting. Engineering Computations, 20:192–210

Yapo P, Gupta VK, Sorooshian S (1996) Automatic calibration of conceptual rainfall-runoff models: sensitivity to calibration data Journal of Hydrology, 181:23–48

Williams R, Zipser D (1989) A learning algorithm for continually running fully recurrent neural networks. Neural Computation, 1:270-280

Winters PR (1960) Forecasting sales by exponentially weighted moving averages. Management Science 6:324–342