New formulation for forecasting streamflow: evolutionary polynomial regression vs. extreme learning machine
Mohammad Rezaie-Balf and Ozgur Kisi

ABSTRACT
Streamflow forecasting is crucial in hydrology and hydraulic engineering since it is capable of optimizing water resource systems or planning future expansion. This study investigated the performances of three different soft computing methods, multilayer perceptron neural network (MLPNN), optimally pruned extreme learning machine (OP-ELM), and evolutionary polynomial regression (EPR) in forecasting daily streamflow. Data from three different stations, Soleyman Tange, Perorich Abad, and Ali Abad located on the Tajan River of Iran were used to estimate the daily streamflow. MLPNN model was employed to determine the optimal input combinations of each station implementing evaluation criteria. In both training and testing stages in the three stations, the results of comparison indicated that the EPR technique would generally perform more efficiently than MLPNN and OP-ELM models. EPR model represented the best performance to simulate the peak flow compared to MLPNN and OP-ELM models while the MLPNN provided significantly under/overestimations. EPR models which include explicit mathematical formulations are recommended for daily streamflow forecasting which is necessary in watershed hydrology management.

Key words | evolutionary polynomial regression, optimally pruned extreme learning machine, river flow forecasting, Tajan catchment

INTRODUCTION
Accurate forecasting of daily river flow is considered essential for many hydrological management decisions such as assessment of risk and control of floods and droughts, hydroelectric energy production, planning of navigation, and sustainable management of water resources. To manage water levels/discharges and to operate water structures more efficiently, new approaches that forecast river flow with high precision and performance are needed (Chen et al. 2015). The data-driven modeling depends on distilling and reutilizing information which is incorporated in the hydrological data lacking in straightforwardly clarifying the physical principles that result in river flow processes (Samsudin et al. 2011; Ghorbani et al. 2016).

assessed the aggregation and disaggregation of rainfall using different data-driven models and wavelet.

Many AI approach-based data driven models, such as optimally pruned extreme learning machine (OP-ELM) and evolutionary polynomial regression (EPR), can obtain a robust correlation (between predicted and observed value) to forecast daily river flow. ANNs, first presented in 1940s, are the most widely used machine learning methods in various areas of water-related research, e.g., rainfall–runoff modeling, precipitation forecasting, groundwater modeling, and discharge prediction (Bhattacharya & Solomatine 2003; Kisi & Cigizoglu 2007; Nourani et al. 2009; Taormina et al. 2012; Tapoglou et al. 2014). Bhattacharya & Solomatine (2003) used ANNs and MT for modeling the water level–discharge relationship of an Indian river. Kisi & Cigizoglu (2007) compared various data-driven ANN methods in river flow forecasting. Tapoglou et al. (2014) forecasted groundwater-levels under climate change scenarios utilizing ANNs with particle swarm optimization.

In recent years, a novel learning approach named extreme learning machine (ELM), which significantly reduces the time required for training single-hidden layer feed-forward neural networks (SLFNs) has been presented (Huang et al. 2004). ELM is a nonlinear technique through which all weights and biases related to hidden node parameters can be generated and fixed randomly without tuning. Thus, the weights at the output layer are analytically determined (Schmidt et al. 1992). This tends to be regarded as the advantage of the ELM approach, which is able to solve simple linear least squares optimization against the nonlinear optimization. This technique is used to find solutions to various problems in different areas of the hydrologic engineering field, such as predicting reference evapotranspiration (Abdullah et al. 2015; Kumar et al. 2016), rainfall–runoff modeling (Taormina & Chau 2015), and daily streamflow forecasting (Lima et al. 2016). ELM is a nonlinear technique through which all weights and biases related to hidden node parameters can be generated and fixed randomly without tuning. Thus, the weights at the output layer are analytically determined (Schmidt et al. 1992). This tends to be regarded as the advantage of the ELM approach, which is able to solve simple linear least squares optimization against the nonlinear optimization. This technique is used to find solutions to various problems in different areas of the hydrologic engineering field, such as predicting reference evapotranspiration (Abdullah et al. 2015; Kumar et al. 2016), rainfall–runoff modeling (Taormina & Chau 2015), and daily streamflow forecasting (Lima et al. 2016).

EPR is a gray-box approach which is generally applied as a predictive model to forecast daily flow of the Tajan used in this research. Giustolisi & Savic (2009) were the first researchers to propose the EPR method in hydroinformatic fields and environmental issues. The methodology of EPR follows the objectives simultaneously as follows: (i) maximizing model accuracy and (ii) minimizing both model coefficients and the number of variables. Thus, these two criteria would be able to introduce the best model (Laucelli & Giustolisi 2011). Although this method has been used in civil engineering, including modeling failures in urban water systems (Berardi et al. 2005; Savic et al. 2006), predicting scour depth around piers (Najafzadeh et al. 2016a), and soil permeability modeling (Ahangar-Asr et al. 2011), it has rarely been used for hydrological issues (e.g., rainfall–runoff modeling, flood forecasting, reference evapotranspiration modeling, etc.). Successful applications of black-box data-driven models in water resources considerations have inspired the exploration of their ability to forecast daily river flow. Investigating preceding research showed that EPR and OP-ELM methods could be rarely employed in hydrological applications.

The main aim of this paper is to investigate the ability of multilayer perceptron neural network (MLPNN), OP-ELM, and EPR in forecasting daily river flow. The paper presents some important points regarding EPR, a case study and datasets further describing development of the proposed models in daily flow forecasting, terminating with forecasting results and conclusions of the research.

STUDY AREA AND DATA

In order to investigate the application of proposed models, the Tajan River basin, located in Mazandaran, Iran was selected as the case study site. Daily river flow estimation was verified for the Tajan catchment, located in the northern part of Iran. The Tajan catchment (53° 56′–36° 17′ north latitude and 53° 7′–53° 42′ east longitude) is drained by the Tajan River, in Sari, the center of Mazandaran Province, with an area of 4,147.22 km². The Tajan watershed has experienced several floods with the most drastic in 1997. The catchment climate is between semi-humid and cold humid. The average slope of the area, discharge of the river, and annual rainfall are 85%, 20 m³/s, and 539 mm, respectively. The catchment elevation varies between 26 and 3,728 m. There exist nine active hydrometric stations in the Tajan watershed. In this study, datasets of three hydrometric stations, namely, Soleyman Tange, Perorich Abad, and Ali Abad were used for modeling purposes (see Figure 1). The characteristics of these stations are presented in Table 1.
The data were obtained from the Meteorological Organization of Mazandaran Province. For this study, daily discharge \( (m^3/s) \) data for 12 years (2002–2013) from three gauging stations (i.e., Soleyman Tange, Perorich Abad, and Ali Abad) were used for forecasting. The data were divided into two phases: approximately 75\% (1,920 data points) of the datasets were used for the training phase while the remaining datasets (25\% of the whole dataset, 637 data points) were kept for testing purposes. Training and test parts can be clearly seen in Figure 2 where time variation graph for the river flows of each station are shown. The other details of datasets and parameter statistics used for the proposed models are given in Tables 2 and 3. In these tables, \( Q_{\text{min}}, Q_{\text{max}}, Q_{\text{mean}}, \) and \( S_d \) indicate the minimum, maximum, mean, and standard deviation of the stream flow data. It is clear from Table 3 that Soleyman Tange station has the highest standard deviation followed by Perorich Abad.

**METHODS**

Hydrological process forecasting is one of the important issues providing reliable and accurate applications in water resources activities. Development of three different soft computing techniques, including traditional ANN, OP-ELM, and EPR approaches which were applied to forecast river daily flow are described in this section.

**MLPNN**

The ANN computational method is inspired by the biological nervous system, which is based on the human brain. The most noteworthy benefit of this method over conventional hydrological models could be its ability to successfully identify both the linear and the nonlinear hydrologic relationship between input and output. Furthermore, ANN can adapt itself to changing circumstances leading to improved performance of the model, reduced calculation times and
accelerated model enhancement (Birikundavyi et al. 2002). Even though there are different ANN types, MLPNN is the common type of neural system most widely used in hydrologic problems (McGarry et al. 1999). The MLPNN can comprise one to many neurons and layers. Generally, the structure of a network consists of three layers: the input layer, through which data enter the network; the hidden layer or layers, where data are processed; and the output layer, which is responsible for producing an appropriate response to the given input.

To establish the structure of a neural network, the number of its layers and neurons in each layer, and excitation function of each neuron should be determined to minimize the error. Different schemes and methods are available for the above mentioned process. Some methods proposed to minimize errors in the training process are the Levenberg–Marquardt algorithm, steepest descent, conjugate gradient algorithm, Bayesian approach, and momentum that are in ascending order of calculation speed, all of which follow a back propagation approach. First, some random values are assigned to the weight and bias of each neuron. Subsequently, the initial training sample vector is fed into the network and the output is calculated and compared to the available observed data. The process is followed by modifying weights or parameters using an iterative algorithm (methods mentioned above) in order to make the error become smaller than its present value (Farfani et al. 2015). More information on ANN structures can be found in Haykin (1998). Furthermore, the architecture of the MLPNN approach is shown in Figure 3.

### ELM

Huang et al. (2006) proposed an original algorithm to determine the weights of hidden neurons (see Figure 4) named ELM. It could decrease the computational time required in training and selecting a model structure by hundreds. Moreover, the algorithm is rather simplistic, which makes the implementation easy to process (Miche et al. 2008).

For a network with $p$ input units, $q$ hidden neurons, and $c$ outputs, the $i$-th output at time step $t$, is given by:

$$o_i(t) = m_i^T h(t)$$  \hspace{1cm} (1)
Table 3 | Data statistics for calibration, validation, and whole of dataset for each station

<table>
<thead>
<tr>
<th>Stations</th>
<th>Calibration dataset</th>
<th>Validation dataset</th>
<th>Whole dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Q&lt;sub&gt;max&lt;/sub&gt;</td>
<td>Q&lt;sub&gt;min&lt;/sub&gt;</td>
<td>Q&lt;sub&gt;mean&lt;/sub&gt;</td>
</tr>
<tr>
<td>Soleyman Tange</td>
<td>18.9</td>
<td>0.07</td>
<td>4.35</td>
</tr>
<tr>
<td>Perorich Abad</td>
<td>18.2</td>
<td>0.08</td>
<td>2.69</td>
</tr>
<tr>
<td>Ali Abad</td>
<td>8.12</td>
<td>0.02</td>
<td>1.07</td>
</tr>
</tbody>
</table>

Unit of discharge is cubic meters per second.

where \( m_i \in \mathbb{R}^q \), \( \forall i \in \{1, \ldots, c\} \), is weight vector which links hidden neurons to the \( i \)-th output neuron, and \( h(t) \in \mathbb{R}^q \) is hidden neuron vector output for a given input pattern \( X(t) \in \mathbb{R}^q \) from a dataset \( \{X(t)\}_{t=1}^N \). Vector \( h(t) \) itself is

\[
h(t) = \left[ f(W_1^T X(t) + b_1), f(W_2^T X(t) + b_2), \ldots, f(W_q^T X(t) + b_q) \right]^T
\]

(2)

where \( b_l \) is the bias of the \( l \)-th hidden neuron, \( W_l \in \mathbb{R}^{q \times d} \) is the weight vector of the \( l \)-th hidden neuron and \( f(\cdot) \) is a sigmoidal activation function. Weight vectors \( w_l \) are randomly sampled from a normal or even from a uniform distribution. On the other hand, \( H = [h(1) \ h(2) \ \ldots \ h(n)] \) is a \( q \times n \) matrix whose \( t \)-th column is the hidden-layer output vector \( h(t) \in \mathbb{R}^q \). Similarly, \( D = [d(1) \ d(2) \ \ldots \ d(n)] \) is a \( c \times n \) matrix whose \( t \)-th column is the desired (target) vector \( d(t) \in \mathbb{R}^c \) associated with the input pattern \( x(t), t = 1, \ldots, N \). Finally, \( M = [m(1) \ m(2) \ \ldots \ m(c)] \) is a \( q \times c \) matrix, whose \( i \)-th column is the weight vector \( m_i \in \mathbb{R}^q, i = 1, \ldots, c \). These three matrices are related by linear mapping:

\[
D = M^T H
\]

(3)

where matrices \( D \) and \( H \) are known and built by the data, while the weight matrix \( M \) is unknown. Nonetheless, the weight matrix \( M \) can be easily computed by means of the Moore–Penrose pseudo-inverse method as follows:

\[
M = (HH^T)^{-1}HD^T
\]

(4)

Assuming that the number of output neurons is equal to the number of classes, class index \( i^* \) for a new input pattern can be calculated by the following decision rule:

\[
i^* = \arg \max_{i=1,\ldots,c} \{O_i\}
\]

(5)

where \( o_i \) is computed by Equation (1).
The OP-ELM contains three main stages, which are shown in Figure 5.

First, a SLFN is created by plenty of hidden layer neurons. To enhance robustness, OP-ELM methodology tends to use a combination of three different types of kernels instead of sigmoid kernel. Linear, sigmoid, and Gaussian are regarded as possible kernel types. The second stage consists of constructing a multi-response sparse regression (MRSR) responsible for mapping outputs of the hidden layer to targets. MRSR can be considered as an extension of least angle regression algorithm, hence, actually a variable ranking technique rather than a selected one (Alencar et al. 2016). In the third stage, the decision on the optimal number of neurons is made by a leave-one-out (LOO) validation method after ranking the neurons. However, the dataset includes a high number of samples, in which case the LOO method can be really time-consuming. To overcome this disadvantage, the prediction sum of squares statistics, can be used giving a direct and exact formula to calculate LOO error in linear models. More details in this respect can be found in Miche et al. (2008).

EPR

EPR is a method founded on evolutionary computation programmed to seek mathematical expressions of polynomial structures which illustrate a true physical system. In EPR, at first, symbolic structures made by a genetic algorithm (GA) are searched and the next stage starts by estimating the constant values by solving a least square (LS) linear problem (Giustolisi & Savic 2006, 2009). This technique is a non-linear global stepwise regression, providing symbolic formulas of models.

EPR combines evolutionary algorithm power with numerical regression in order to improve its efficiency. Polynomial models are capable of combining the independent variables simultaneously with user-defined function (Laucelli et al. 2005):

\[
y = \sum_{j=1}^{m} F(X, f(X), a_j) + a_0
\]

where \( F \) is a function produced by the process, \( y \) is the estimated output of the system, \( X \) is the matrix of input variables, \( a_j \) is a constant value, \( f \) is a function defined by the user, and \( m \) is the number of terms of the expression excluding the term \( a_0 \). To enhance the performance of the response (dependent) variable, user-defined function can be included. To construct symbolic models in EPR there are two basic stages: (i) identification of structure and (ii) estimation of parameters, in which a GA simple search method is used to search in the space of the model structure. Similar to other numerical regression, EPR applies the LS method to estimate the parameters of the selected model structure based on the performed GA search. Crossover and mutation are two standard operators of GA where search progresses with them. This is not an exhaustive search since it is impractical to conduct such a search in an infinite search space (Laucelli et al. 2005). EPR permits pseudo-polynomial
expressions which belong to Equation (6):

\[
\hat{y} = a_0 + \sum_{j=1}^{m} a_j (X_1)^{ES(j,1)} \cdots (X_K)^{ES(j,K)} f((X_1)^{ES(j,K+1)}) \\
\cdots f((X_K)^{ES(j,2K)}) \text{ case 1} \\
\hat{y} = a_0 + \sum_{j=1}^{m} a_j f((X_1)^{ES(j,1)} \cdots (X_K)^{ES(j,K)}) \text{ case 2} \\
\hat{y} = a_0 + \sum_{j=1}^{m} a_j (X_1)^{ES(j,1)} \cdots (X_K)^{ES(j,K)} f((X_1)^{ES(j,K+1)}) \\
\cdots (X_K)^{ES(j,2K)} \text{ case 3} \\
\hat{y} = g a_0 + \sum_{j=1}^{m} a_j (X_1)^{ES(j,1)} \cdots (X_K)^{ES(j,K)} \text{ case 4}
\]

where \( \hat{y} \) is model prediction vector, ES is the matrix of exponents (elements of matrix can assume values within user-defined bounds), and \( K \) is the number of variables (inputs) which are independent predictors, whereas each element is the candidate exponent used for each single input. User-specified functions \( f \) shown in Equation (7) might be tangent hyperbolic, natural logarithmic, exponential, and so on. It should be noted that the last structure in Equation (7) needs to be assumed as an invertible function \( g \), due to the following stage of estimating parameters. Using the term ‘pseudo-polynomial expressions’ here is due to the fact that the parameters in any expressions of Equation (7) can be computed as a linear problem and/or true polynomial expression. A typical flow diagram for the EPR procedure is presented in Figure 6. For more detailed information on the EPR procedure the reader can refer to Giustolisi & Savic (2006) and Laucelli et al. (2005).

Performance evaluation criteria

To compare the rainfall–runoff simulation performance of the developed models, different statistical indices (Equations (8)–(10)) were used. The indices include correlation coefficient \( R \), root mean squared error \( \text{RMSE} \), and relative absolute error \( \text{RAE} \):

\[
R = \frac{\sum_{i=1}^{M} (O_i - \overline{O}) \cdot (P_i - \overline{P})}{\sqrt{\sum_{i=1}^{M} (O_i - \overline{O})^2 \sum_{i=1}^{M} (P_i - \overline{P})^2}} 
\]  

\[
\text{RMSE} = \frac{\sum_{i=1}^{M} (P_i - O_i)^2}{M} 
\]  

\[
\text{RAE} = \frac{\sum_{i=1}^{M} |P_i - O_i|}{\sum_{i=1}^{M} |O_i - \overline{O}|} 
\]

where \( O \) and \( P \) signifies the observed runoff and predicted runoff by the model, respectively. \( \overline{O} \) is the mean of the observed values and \( \overline{P} \) represents the mean of the predicted values. \( M \) shows the total number of dataset samples. The \( R \) measures how well considered independent variables account for the measured dependent variable. The \( \text{RMSE} \) is used to measure estimating accuracy, which produces a positive value by squaring the errors. \( \text{RMSE} \) would increase from zero for perfect estimates to large positive values as the differences between simulations and observations grow. A small value of \( \text{RMSE} \) and high value of \( R \) (up to 1) show high efficiency of the model. \( \text{RAE} \) is the ratio of the absolute error of the measurement to the accepted measurement. Lower value of the \( \text{RAE} \) illustrates high performance of the proposed model (Kisi et al. 2013; Najafzadeh et al. 2014; Kisi & Parmar 2016).

RESULTS AND DISCUSSION

Analysis of results for proposed models

In this study, the MLPNN, OP-ELM, and EPR approaches were investigated for present daily discharge forecasting in Tajan River. As there is no procedure to select the relevant inputs to forecast daily river flow, several input combinations were applied using MLPNN in the three stations studied. The inputs of the MLPNN models present the previously recorded daily discharge values (t-1 to t-6), whereas the output corresponds to the daily flow value at the current time (t). Thus, the combinations of input data of daily flow values are defined in six scenarios that are illustrated in Table 4. Thereafter, with calculating the statistical measures that are presented in Table 5, the optimal input combinations were selected for forecasting the daily flow of Tajan River.
The performance evaluation of the MLPNN technique through the R, RMSE, and RAE statistical indexes is made in Table 5. As can be seen from the table, six different MLPNN models with different configurations were applied for three stations. 1-2-1 in the third column of the table indicates an ANN model having 1 input, 2 hidden, and 1 output nodes, respectively. As seen from Table 5, the RMSE of the MLPNN model's range is 2.743–11.32, 3.874–7.414 and 0.720–2.107 for the Soleyman Tange, Perorich Abad and Ali Abad stations, respectively. It should be noted that these ranges are parallel to the data ranges given in Table 3, in which the streamflow data of Soleyman Tange has the highest range (0.071–34.3 m³/s) and standard deviation (5.86 m³/s). Based on the performance evaluation
testing stage. Figure 7 shows the scatterplots and time series plots of the daily streamflow forecasting results for Soleyman Tange station. In the testing stage (Table 6), it can be seen that the EPR method estimated the daily streamflow at Soleyman Tange station with a higher correlation ($R^2 = 0.991$) and lower error statistics of RMSE ($0.263$) and RAE ($0.219$) compared to the OP-ELM ($R^2 = 0.977$, RMSE $= 0.263$, and RAE $= 0.219$) and MLPNN ($R^2 = 0.977$, RMSE $= 0.181$, and RAE $= 0.414$) models.

It should be noted that one of the key aspects in using EPR is the capability of this method in giving mathematical expression between the input–output variables, which is not the case in MLPNN and OP-ELM models. The corresponding explicit expression of the present EPR model is:

$$Q_{S(t)} = 1.1507Q_{S(t-1)}^{0.936} \exp(-0.5Q_{S(t-3)}^{0.5} - 2Q_{S(t-2)}^{0.5} - 0.5Q_{S(t-1)}) + 2.478Q_{S(t-3)}^{1.5}Q_{S(t-1)}^{0.5} \exp(-0.5Q_{S(t-3)}^{0.5} - 0.5Q_{S(t-1)})$$

(11)

where $Q_{S(t)}$ is daily discharge at present time ($t$) at Soleyman Tange station. In the testing stage (Table 6), it can be noted that Equation (11), given by the EPR model, estimated the daily flow with a better accuracy than those using OP-ELM and MLPNN techniques similar to the training stage. Figure 7 shows the scatterplots and time series plots of the daily streamflow forecasting results for Soleyman Tange station among the other scenarios (i.e., S1, S2, S4, S5 and S6) has the highest correlation ($R^2 = 0.936$) and the lowest errors in terms of RMSE ($2.742$) and RAE ($1.666$). Also, for Perorich Abad and Ali Abad stations, S3 and S2 were selected as optimal structures for daily streamflow forecasting, respectively (Table 5).

### Table 4 | Model structures for forecasting daily flow of case study stations

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Model Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>$Q_{S(t)} = f(Q_{S(t-1)})$</td>
</tr>
<tr>
<td>S2</td>
<td>$Q_{S(t)} = f(Q_{S(t-1)}, Q_{S(t-2)})$</td>
</tr>
<tr>
<td>S3</td>
<td>$Q_{S(t)} = f(Q_{S(t-1)}, Q_{S(t-2)}, Q_{S(t-3)})$</td>
</tr>
<tr>
<td>S4</td>
<td>$Q_{S(t)} = f(Q_{S(t-1)}, Q_{S(t-2)}, Q_{S(t-3)}, Q_{S(t-4)})$</td>
</tr>
<tr>
<td>S5</td>
<td>$Q_{S(t)} = f(Q_{S(t-1)}, Q_{S(t-2)}, Q_{S(t-3)}, Q_{S(t-4)}, Q_{S(t-5)})$</td>
</tr>
<tr>
<td>S6</td>
<td>$Q_{S(t)} = f(Q_{S(t-1)}, Q_{S(t-2)}, Q_{S(t-3)}, Q_{S(t-4)}, Q_{S(t-5)}, Q_{S(t-6)})$</td>
</tr>
</tbody>
</table>

### Table 5 | Test performance measures of the MLPNN model in daily flow forecasting

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Stations</th>
<th>Best structure of the ANN model</th>
<th>RMSE</th>
<th>R</th>
<th>RAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>Soleyman Tange</td>
<td>1-2-1-1</td>
<td>6.362</td>
<td>0.931</td>
<td>68.01</td>
</tr>
<tr>
<td></td>
<td>Perorich Abad</td>
<td>1-1-1-1</td>
<td>3.874</td>
<td>0.863</td>
<td>6.365</td>
</tr>
<tr>
<td></td>
<td>Ali Abad</td>
<td>1-2-1-1</td>
<td>0.720</td>
<td>0.850</td>
<td>2.556</td>
</tr>
<tr>
<td>S2</td>
<td>Soleyman Tange</td>
<td>2-3-1-1</td>
<td>7.632</td>
<td>0.934</td>
<td>21.00</td>
</tr>
<tr>
<td></td>
<td>Perorich Abad</td>
<td>2-2-1-1</td>
<td>4.060</td>
<td>0.865</td>
<td>6.867</td>
</tr>
<tr>
<td></td>
<td>Ali Abad</td>
<td>2-2-1-1</td>
<td>0.547</td>
<td>0.865</td>
<td>2.250</td>
</tr>
<tr>
<td>S3</td>
<td>Soleyman Tange</td>
<td>3-2-1-1</td>
<td>2.742</td>
<td>0.936</td>
<td>1.666</td>
</tr>
<tr>
<td></td>
<td>Perorich Abad</td>
<td>3-2-1-1</td>
<td>5.009</td>
<td>0.938</td>
<td>12.98</td>
</tr>
<tr>
<td></td>
<td>Ali Abad</td>
<td>3-2-1-1</td>
<td>1.469</td>
<td>0.862</td>
<td>2.077</td>
</tr>
<tr>
<td>S4</td>
<td>Soleyman Tange</td>
<td>4-2-1-1</td>
<td>9.944</td>
<td>0.921</td>
<td>15.11</td>
</tr>
<tr>
<td></td>
<td>Perorich Abad</td>
<td>4-2-1-1</td>
<td>5.416</td>
<td>0.932</td>
<td>13.70</td>
</tr>
<tr>
<td></td>
<td>Ali Abad</td>
<td>4-3-1-1</td>
<td>1.678</td>
<td>0.847</td>
<td>2.126</td>
</tr>
<tr>
<td>S5</td>
<td>Soleyman Tange</td>
<td>5-3-1-1</td>
<td>6.528</td>
<td>0.910</td>
<td>26.32</td>
</tr>
<tr>
<td></td>
<td>Perorich Abad</td>
<td>5-3-1-1</td>
<td>7.414</td>
<td>0.924</td>
<td>10.62</td>
</tr>
<tr>
<td></td>
<td>Ali Abad</td>
<td>5-3-1-1</td>
<td>2.107</td>
<td>0.806</td>
<td>2.427</td>
</tr>
<tr>
<td>S6</td>
<td>Soleyman Tange</td>
<td>6-3-1-1</td>
<td>11.32</td>
<td>0.820</td>
<td>14.56</td>
</tr>
<tr>
<td></td>
<td>Perorich Abad</td>
<td>6-3-1-1</td>
<td>7.201</td>
<td>0.891</td>
<td>9.441</td>
</tr>
<tr>
<td></td>
<td>Ali Abad</td>
<td>6-4-1-1</td>
<td>2.099</td>
<td>0.783</td>
<td>2.409</td>
</tr>
</tbody>
</table>

### Table 6 | Results of performances for proposed models – Soleyman Tange station

<table>
<thead>
<tr>
<th></th>
<th>MLPNN</th>
<th>OP-ELM</th>
<th>EPR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMSE</td>
<td>R</td>
<td>RAE</td>
</tr>
<tr>
<td>Training stage</td>
<td>1.181</td>
<td>0.977</td>
<td>0.414</td>
</tr>
<tr>
<td>Testing stage</td>
<td>2.741</td>
<td>0.931</td>
<td>1.666</td>
</tr>
</tbody>
</table>
series graphs of the observed and predicted river flow forecasts using MLPNN, OP-ELM, and EPR models for testing datasets.

On the basis of this figure, the observed and predicted values of daily discharge from EPR showed better agreement than the MLPNN and OP-ELM, especially for the peak
flows. Significantly over- and underestimations are seen for the MLPNN model.

**Comparison of MLPNN, OP-ELM, and EPR methods for the data of Perorich Abad station**

Similar to the methodology used for Soleyman Tange station, three proposed models (i.e., MLPNN, OP-ELM, and EPR) were constructed for the purpose of daily flow estimation in the Perorich Abad station. The evaluation criteria RMSE, R, and RAE during training and testing stages for proposed techniques are illustrated in Table 7. Statistics given for the training stage clearly show that the EPR model whose inputs are the discharge of 3 previous days better approximates the corresponding observed flow values than the other models with a higher R (0.913) and lower RMSE (0.384) and RAE (1.125). The proposed equation for daily flow of Tajan River at Perorich Abad from the EPR model is:

\[
QP(t) = 1.2155 QP(t-1) \times \exp(-2QP(t-3) - 2QP(t-2) - 2QP(t-1)) \\
+ 4.7655 QP^{0.5}(t-3) QP^{1.5}(t-1) \\
\times \exp(-0.5QP(t-3) + 0.5QP(t-2) - 2QP(t-1))
\]  

where \(QP(t)\) is daily discharge at present time (t) at Perorich Abad station. Also, it can be observed that Equation (12) from EPR is able to attain acceptable forecasting results as the R, RMSE, and RAE values during the testing period are 0.932, 1.301, and 0.883, respectively. Figure 8 indicates the observed and modeled runoff values with the optimal MLPNN, OP-ELM, and EPR models during the test datasets through the scatterplots and time series graphs. MLPNN gives significant overestimates for the daily river flows of Perorich-Abad station. The peak discharge forecast by the MLPNN model is about 4.68 m³/s higher when compared with the observed value. This over-forecasting for peak discharge is intolerable for flood warning. According to the scatterplots, the OP-ELM significantly overestimates high discharges (>10 m³/s) while the EPR significantly underestimates them. The EPR technique provides a better forecast of the peak discharge, and performs better than the MLPNN and OP-ELM techniques.

**Comparison of MLPNN, OP-ELM, and EPR methods for the data of Ali Abad station**

Similar to the previous two stations (i.e., Soleyman Tange and Perorich Abad), the MLPNN, OP-ELM, and EPR were applied to forecast daily river flow at Tajan basin. The training and testing performance statistics of the proposed techniques are given in Table 8. The EPR model again was the best model in terms of RMSE (0.384), RAE (1.125), and R (0.913) in the training stage compared with the MLPNN and OP-ELM models. The RAE (2.250) and RMSE (0.865) of the MLPNN approach in the testing stage indicate poor performance of estimation of daily flow at Ali Abad station. Figure 9 illustrates the scatterplots and time series graphs of the predicted and observed daily flow forecasts using applied techniques in training and testing stages at Ali Abad station.

As can be seen from Figure 9, the developed OP-ELM model at Ali Abad station is not as good as that constructed for Soleyman Tange and Perorich Abad stations. The OP-ELM and EPR methods have approximately similar performances in low flow forecasting, but in peak flow forecasting the EPR is the more robust method with a higher accuracy than the MLPNN and OP-ELM methods. Also, the peak discharge forecasts by the MLPNN model are underestimated when compared with the observed value. After all these comparisons, the selected equation

| Table 7 | Results of performances for proposed models – Perorich Abad station |
|---------|-----------------|-----------------|--------------|
|         | MLPNN           | OP-ELM          | EPR           |
|         | RMSE       | R     | RAE       | RMSE       | R     | RAE       | RMSE       | R     | RAE       |
| Training stage | 0.715 | 0.902  | 1.373     | 0.724       | 0.903  | 1.713     | 0.691       | 0.912  | 1.627     |
| Testing  stage  | 5.009 | 0.957  | 12.98     | 1.650       | 0.924  | 1.39      | 1.301       | 0.932  | 0.883     |
for daily discharge as a trade-off between accuracy and parsimony at Ali Abad station:

\[ QA(t) = 1.025QA(t-1) \times \exp\left(-2QA(t-3) - 1.5QA(t-1)\right) \]
\[ + 3.837QA^2(t-1) \times \exp\left(-2QA(t-1)\right) \]
\[ + 0.0114QA^2(t-2)QA^2(t-1) \times \exp(1.5QA(t-2) + 1.5QA(t-1)) \]

(13)

where \( QA(t) \) is daily discharge at present time (t) at Ali Abad station.

Overall, the EPR seems to more adequate for forecasting daily streamflow when compared to MLPNN and OP-ELM methods. The flexible structure of EPR may have provided it with a better ability to learn investigated phenomena. MLPNN generally underestimated the peak discharge.
Table 8  | Results of performances for proposed models – Ali Abad station

<table>
<thead>
<tr>
<th></th>
<th>MLPNN</th>
<th>OP-ELM</th>
<th>EPR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMSE</td>
<td>R</td>
<td>RAE</td>
</tr>
<tr>
<td>Training</td>
<td>0.392</td>
<td>0.905</td>
<td>2.469</td>
</tr>
<tr>
<td>Testing</td>
<td>0.547</td>
<td>0.865</td>
<td>2.250</td>
</tr>
</tbody>
</table>

Figure 9  | Hydrographs and scatterplots of observed and predicted river flows for proposed models at Ali Abad station: (a) MLPNN model, (b) OP-ELM model, and (c) EPR model.
values at all three stations. The main reason for this may be the lesser number of data of extreme values. The main disadvantage of the MLPNN method is that it has a high number of data for adequately learning.

**CONCLUSION**

In this study, MLPNN, OP-ELM, and EPR models were employed for forecasting of daily river flow using inputs of previous daily flows of Soleyman Tangi, Perorich Abad and Ali Abad stations that are located along the Tajan River, Mazandaran region of Iran. The MLPNN model was tested by applying it to different input combinations of daily flow data. After selecting the optimal input combination by MLPNN, the performances of the three methods were evaluated based on three different criteria, R, RMSE, and RAE. The obtained results indicated that the EPR model was superior to the MLPNN and OP-ELM models for forecasting daily flows of the studied river. MLPNN model could not simulate the daily discharge values well and the accuracy of this predictive model was generally found to be low. On the other hand, the EPR approach provided a better forecast of the peak flow than the MLPNN and OP-ELM techniques. The main advantage of the EPR model is its explicit mathematical formulations. It can be simply used in practical applications. Instead, the MLPNN and OP-ELM models are black-box models whose formulations are closed.

The proposed techniques may also be compared in other hydrological applications (e.g., short-term wind speed predictions, sea water level forecasting, prediction of daily evapotranspiration). The efficiency and accuracy of the proposed EPR models may be compared with other AI methods in future studies.

**REFERENCES**


First received 14 December 2016; accepted in revised form 17 February 2017. Available online 27 March 2017.