Acta Geophysica (2018) 66:717–730
https://doi.org/10.1007/s11600-018-0168-4

RESEARCH ARTICLE - HYDROLOGY

Reservoir water level forecasting using group method of data handling

Amir Hossein Zaji[1] · Hossein Bonakdari[1] · Bahram Gharabaghi[2]

Received: 3 November 2017 / Accepted: 29 May 2018 / Published online: 2 June 2018

Abstract
Accurately forecasted reservoir water level is among the most vital data for efficient reservoir structure design and management. In this study, the group method of data handling is combined with the minimum description length method to develop a very practical and functional model for predicting reservoir water levels. The models' performance is evaluated using two groups of input combinations based on recent days and recent weeks. Four different input combinations are considered in total. The data collected from Chahnimeh#1 Reservoir in eastern Iran are used for model training and validation. To assess the models' applicability in practical situations, the models are made to predict a non-observed dataset for the nearby Chahnimeh#4 Reservoir. According to the results, input combinations (L, L_{-1}) and (L, L_{-1}, L_{-12}) for recent days with root-mean-squared error (RMSE) of 0.3478 and 0.3767, respectively, outperform input combinations (L, L_{-7}) and (L, L_{-7}, L_{-14}) for recent weeks with RMSE of 0.3866 and 0.4378, respectively, with the dataset from https://www.typingclub.com/st. Accordingly, (L, L_{-1}) is selected as the best input combination for making 7-day ahead predictions of reservoir water levels.

Keywords Daily reservoir water level forecasting · Group method of data handling · Input combination · Soft computing · Time-series prediction

Introduction

Among the most significant aspects of reservoir structures and reservoir management for industrial, agricultural and drinking water supply is the ability to predict reservoir water levels accurately. Various environmental parameters influence reservoir water levels, such as water consumption for agriculture, air and water temperature, wind speed, rainfall amount, etc. Two major approaches can be applied to predict reservoir water levels. First is to evaluate the models using the environmental variables that affect reservoir water levels. The second approach is to accept that the environmental variables affect previous reservoir water levels and use past data to predict future levels. Actually, using past reservoir water level data can significantly decrease the discrepancy between variables in a model.

Several studies have been conducted on the topic of simulating reservoir water levels in relation to the design and construction of reservoir hydraulic structures as well as the management of agricultural, industrial and drinking water supplies. Koppula (1980) used two univariate prediction methods, namely the Box–Jenkins (BJ) time-series technique and harmonic analysis, to model monthly reservoir water level prediction. Guganesharajah and Shaw (1984) developed a model for predicting low-level variations for different periods ahead. Gladkov et al. (1991) investigated the relation between seepage and reservoir water levels. Crapper et al. (1996) used water balance models to predict reservoir water level variations.

In recent years, researchers have successfully employed artificial intelligence (AI) techniques to model various multivariable complex problems (Zaji and Bonakdari 2014; Gholami et al. 2017; Zaji and Bonakdari 2018; Gholami et al. 2018). Due to the nonlinearity and complexity of hydrology problems, AI has become an accepted modeling and simulation method in this context. Khan and Coulibaly (2006) used support vector machines (SVMs) to predict

✉ Hossein Bonakdari
bonakdari@yahoo.com

[1] Department of Civil Engineering, Razi University, Kermanshah, Iran

[2] School of Engineering, University of Guelph, Guelph, ON NIG 2W1, Canada

mean monthly reservoir water levels and compared the results with the multilayer perceptron (MLP) and seasonal autoregressive (SAR). Altunkaynak (2007) developed an MLP model for reservoir water level prediction and compared the MLP results with autoregressive (AR), moving average (MA) and autoregressive moving average with eXogenous input (ARMAX) models. Ondimu and Murase (2007) developed an MLP method to forecast reservoir water levels by using rainfall, evaporation, discharge and water levels of nearby rivers. Güldal and Tongal (2010) used the recurrent neural network (RNN) and adaptive network-based fuzzy inference system (ANFIS) methods to predict reservoir water level changes and compared the results with the AR and autoregressive moving average (ARMA) methods. Kisi et al. (2012) used artificial neural networks (ANNs), ANFIS and gene expression programming (GEP) to predict the daily reservoir water levels up to 3 days ahead. Kakahaji et al. (2013) applied the MLP and local linear neuro-fuzzy (LLNF) methods to the reservoir water level forecasting problem and compared the results with the autoregressive with exogenous input (ARX) and BJ methods. Mahdi Hadi et al. (2013) compared the predictability of the genetic programming (GP), ANN, ANFIS and ARIMA methods to identify the most appropriate method of predicting reservoir water levels. Lan (2014) developed different SVM methods with various kernel functions (linear, polynomial and radial basis functions) for the complex problem of forecasting reservoir water levels. Kisi et al. (2015) predicted the daily reservoir water levels for three prediction horizons using a hybrid method of SVM with the firefly (FF) algorithm and compared the results of the proposed method with GP and MLP results. Shiri et al. (2016) forecasted the daily Urmia Reservoir water levels using the extreme learning machine (ELM) approach and compared the results with the GP and MLP methods. Yadav and Eliza (2017) utilized a hybrid method of wavelet transformation with SVM to forecast the daily Loktak Reservoir water fluctuations. The researchers utilized past reservoir water levels and other hydrometeorological data to predict 20-day ahead reservoir water levels.

The group method of data handling (GMDH) method is a robust subset of AI techniques, and it is used in various hydrology and hydraulics problems, such as sediment transport (Ebtehaj et al. 2016; Najafzadeh and Bonakdari 2017), rainfall–runoff modeling (Tsai et al. 2010; Zhang et al. 2013), scour modeling (Najafzadeh and Azamathulla 2013; Najafzadeh et al. 2014; Najafzadeh and Lim 2015), side weir discharge coefficient prediction (Ebtehaj et al. 2015, 2018), soil water characteristics (Neyshaburi et al. 2015) and simulating river characteristics (Najafzadeh and Tafarojnoruz 2016; Shaghaghi et al. 2017).

The goal of the present study is to develop MDL based on GMDH and identify the best model to predict reservoir water levels 1 week ahead. Two groups of input combinations are considered: reservoir water levels from the previous days [L, L_{-1}] and [L, L_{-1}, L_{-2}] and the previous weeks [L, L_{-7}] and [L, L_{-7}, L_{-14}]. To estimate the GMDH-MDL models' usability in practical situations, the developed models are applied to predict the daily water levels of another reservoir.

Materials and method

Data collection

The Chahnimeh reservoirs are found in Zabol and are among the largest cavities in the south of Sistan and Baluchestan Province in southeastern Iran. These reservoirs cover over 50 million m^2, and the water is used for drinking and irrigation in Zabol and Zahedan.

In this study, the performance of GMDH-MDL is evaluated in two stages. First, the dataset collected for Chahnimeh#1 Reservoir is used for model training and testing. Then to determine the usability of the proposed models for other reservoirs in the same geographical region, the most appropriate model found in the first stage is used to forecast the daily water level of another reservoir (Chahnimeh#4), which is near the base reservoir. The reservoirs under investigation are described subsequently. The locations of Chahnimeh#1 and Chahnimeh#4 are shown in Fig. 1.

Chahnimeh#1 Reservoir

Chahnimeh#1 is in eastern Iran at a longitude of 30°46′ to 30°51′ and latitude of 61°39′ to 61°43′. Information collected from Chahnimeh#1 was used to train and test the GMDH-MDL models. Measurements were taken over 4 years. In the present study, the first 2 years' data were used to train the models and the next 2 years' data were used for testing. The properties and water level fluctuations of Chahnimeh#1 are represented in Fig. 2 and Table 1, respectively.

Chahnimeh#4 Reservoir

One of the most noteworthy characteristics of numerical models is their usability in practical situations. In this study, information from Chahnimeh#4 served as a non-observed dataset to evaluate the models' performance in forecasting the daily water levels of another reservoir located in the same geographical region. Chahnimeh#4 Reservoir is situated in eastern Iran, near Chahnimeh#1. The longitude and latitude intervals of Chahnimeh#4 are 30°44′ to 30°49′ and 61°31′ to 61°37′, respectively. The

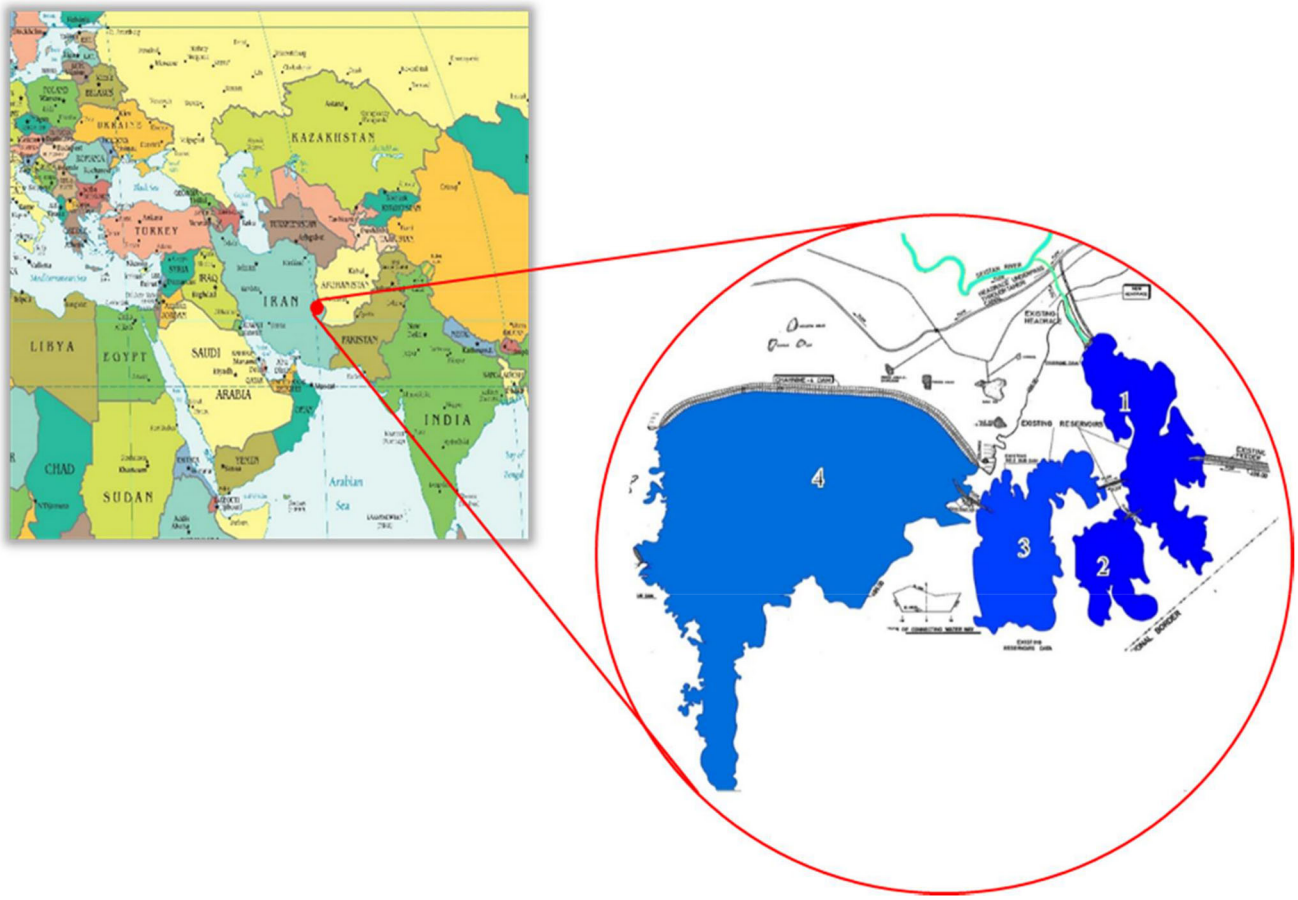


Fig. 1 Locations of Chahnimeh#1 and Chahnimeh#4 Reservoirs

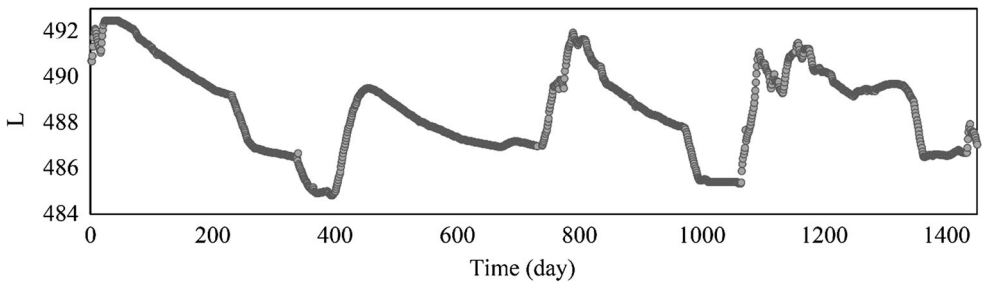


Fig. 2 Chahnimeh#1 water level fluctuations

Table 1 Chahnimeh#1 reservoir properties for the considered period

Reservoir name	Variable	Unit	Statistical parameters			
			Min	Max	Mean	SD
Chahnimeh#1	*L*	m	484.83	492.50	488.57	1.90

daily reservoir water levels of Chahnimeh#4 were measured for 3 years. The entire dataset for Chahnimeh#4 served as a non-observed dataset for numerical model testing. The properties and water level fluctuations of Chahnimeh#4 are represented in Fig. 3 and Table 2, respectively.

Group method of data handling (GMDH)

Artificial intelligence techniques have a significant impact on complex problem modeling. In order to simulate a system with an analytical or theoretical structure, it is necessary to simulate all system components. Therefore, regardless of model complexity or simplicity, the entire system set must be modeled. Numerical models have the advantage that only the input and output variables of the respective model are considered, and the complexity of

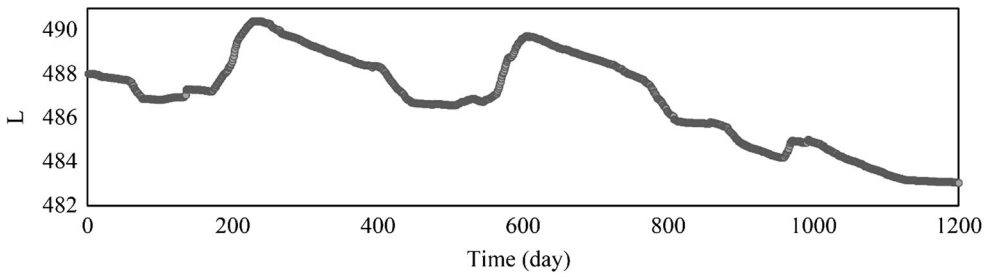


Fig. 3 Chahnimeh#4 water level fluctuations

Table 2 Chahnimeh#4 Reservoir properties for the considered period

Reservoir name	Variable	Unit	Statistical parameters			
			Min	Max	Mean	SD
Chahnimeh#4	*L*	m	483.06	490.43	486.90	2.06

other system variables does not affect the complexity of the numerical model. Reservoir water level is a multivariable system that cannot be modeled with analytical or theoretical models. In the present study, one of the most powerful numerical simulation methods is employed to model future reservoir water levels by using past information about a reservoir. The GMDH method (used in the current study) with its subsets is frequently applied in modeling complex engineering problems. However, this method has rarely been used to model the present study topic.

GMDH (Ivakhnenko 1970, 1971) is a mathematical-based self-organizing learning machine that attempts to find the most appropriate relation between the model inputs and outputs. GMDH can be used successfully in cases where there is no background on the theoretical relationship between the input variables and the results.

The GMDH method can be applied from two perspectives: (1) the mathematical basis of GMDH and (2) the theoretical aspect of GMDH with algorithm implementation. Volterra's function is used according to the mathematical basis of GMDH. A discrete form of the Volterra series is used to establish a connection between the input variables and the results. The Volterra series have been used extensively to develop nonlinear models (Ivakhnenko 1971; Iba et al. 1994; Nikolaev and Iba 2001). This function is defined as follows:

$$y = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} a_{ijk} x_i x_j x_k + \ldots \quad (1)$$

Equation (1) is a Kolmogorov–Gabor polynomial that is used in certain backward equations. In Eq. (1), x represents the model input variables and a represents coefficients that are calculated during model training.

The GMDH algorithm is deemed a neural network with a neural network structure. Therefore, the GMDH neural network consists of one input layer, one or more hidden layers and one output layer. Each layer consists of neurons. The input layer contains the input variables, and the output layer contains the output variable. In the present study, the input variables are previous reservoir water levels, and the output variable is the 7-day ahead reservoir water level. The hidden layers consist of neurons that serve to establish an interpolation between the input and output layers. The GMDH neural network relies on the feedforward approach with the quadratic form of Eq. (1) (Nariman-zadeh et al. 2002, 2005; Najafzadeh and Barani 2011). In the GMDH neural network, different pairs of neurons from each layer are selected to compute a new neuron in the next layer. In case of m observations and for given inputs (x_1, x_2, …, x_n), the results are represented as follows:

$$y_i = f(x_{i1}, x_{i2}, \ldots, x_{in}) \quad \text{where } (i = 1, 2, \ldots, m) \quad (2)$$

The principal purpose of GMDH is to find a predicted output (Y_{pred}) according to Eq. (3) that has the least square difference from the observed output (y_i) according to Eq. (4).

$$y_{\text{Pred}\,i} = f_{\text{pred}}(x_{i1}, x_{i2}, \ldots, x_{in}) \quad \text{where } (i = 1, 2, \ldots, m) \quad (3)$$

$$\text{Difference}^2 = \sum_{i=1}^{m} [y_{\text{Pred}\,i} - y_i]^2 \quad (4)$$

The full form of the Volterra series needs to be simplified before use with the GMDH algorithm; therefore, the two-variable quadratic form of Eq. (1) is used as follows

$$G(x_i, x_j) = a_0 + a_1 x_i + a_2 x_j + a_3 x_i^2 + a_4 x_j^2 + a_5 x_i x_j \quad (5)$$

Equation (5) indicates that the GMDH neural network considers two neurons together to build a new neuron. In calculating the a_0 to a_5 coefficients, the regression method is applied to minimize e according to Eq. (6). Here, least squares minimization is used to determine the unknown coefficients.

$$e = \frac{\sum_i^m (y_i - G_i)^2}{m} \tag{6}$$

where i represents the sample indices.

A flowchart of the GMDH neural network algorithm used in this study is presented in Fig. 4. According to this figure, the GMDH algorithm changes the number of layers in the biggest loop. Subsequently, the number of neurons represents the changes in the middle loop, and finally, the number of possible combinations represents the changes in the smallest loop. In order to establish a new combination in the current GMDH, each neuron is permitted to input two independent variables. According to the given flowchart, the independent variables are selected from the first layer neurons (input variables) and the previous layer neurons. The number of possible choices is calculated with Eq. (7).

$$\binom{n}{2} = n(n-1)/2 \tag{7}$$

After evaluating each neuron and adding it to the present layer, all combinations are considered again and the combinations not used by any preceding layers are deleted.

Figure 4 indicates that GMDH requires a defined criterion for the process to stop or add a new layer, neuron or combination to the model structure. In this study, the minimum description length (MDL) method is used together with the GMDH algorithm as the criterion function. MDL was described by Grünwald et al. (2005), who used MDL as a genetic programming (GP) criterion. MDL is added to a numerical procedure for two primary reasons: it increases model accuracy and prevents excessive model expansion. Model performance with numerical methods can be improved by increasing the model size. However, greater model size has two significant disadvantages. First, increasing the number of layers and neurons in the model reduces the possibility of using the results in practical situations due to model complexity. Second, increasing the number of layers and neurons may trap the model in overtraining. Overtraining occurs when a model performs very well with the training dataset and very badly with the testing and validation datasets. The MDL method establishes a trade-off between model performance and complexity. The MDL criterion is calculated with the following equation:

$$\text{Criterion} = n \times \log(y_{\text{Pred}} - y)^2 + \text{Complexity} \times \log(n) \tag{8}$$

where n is the number of input samples. The first term in this equation controls the model's prediction accuracy, and the second term controls the model size. The complexity is calculated as follows:

$$\text{Complexity} = \text{Number of layers} + \text{Number of total neurons} \tag{9}$$

According to Eqs. (8) and (9), the criterion estimated by GMDH-MDL is a combination of model error and complexity, because increasing each raises the criterion coefficient and vice versa.

Performance evaluation statistics

Several statistical methods can be applied to evaluate forecasting ability. In the present study, the statistical indices RMSE, mean absolute error (MAE), standard error prediction percent (SEP%) and absolute deviation percent (δ%) are used. In addition, the residual, standard deviation (SD) and coefficient of determination (R^2) concepts are also utilized. The residual represents the difference between forecasted and observed samples (Eq. 10). RMSE calculates the standard deviation of the residuals (Eq. 11). MAE is the average of absolute residuals and represents the closeness between forecasted and actual values (Eq. 12). The primary advantage of SEP% and δ% (Eqs. 13 and 14) is the non-dimensionality. Therefore, these indices can help compare the different variables, and the models are not scale dependent. SD represents the deviation from the average. If the sample tends to be close to the mean, the SD is small; otherwise, if the sample disperses from the average, the SD is high. R^2 denotes how well the model replicated the actual samples.

$$\text{Residual}_i = (e_i - o_i) \tag{10}$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N (e_i - o_i)^2}{N}} \tag{11}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^N |e_i - o_i| \tag{12}$$

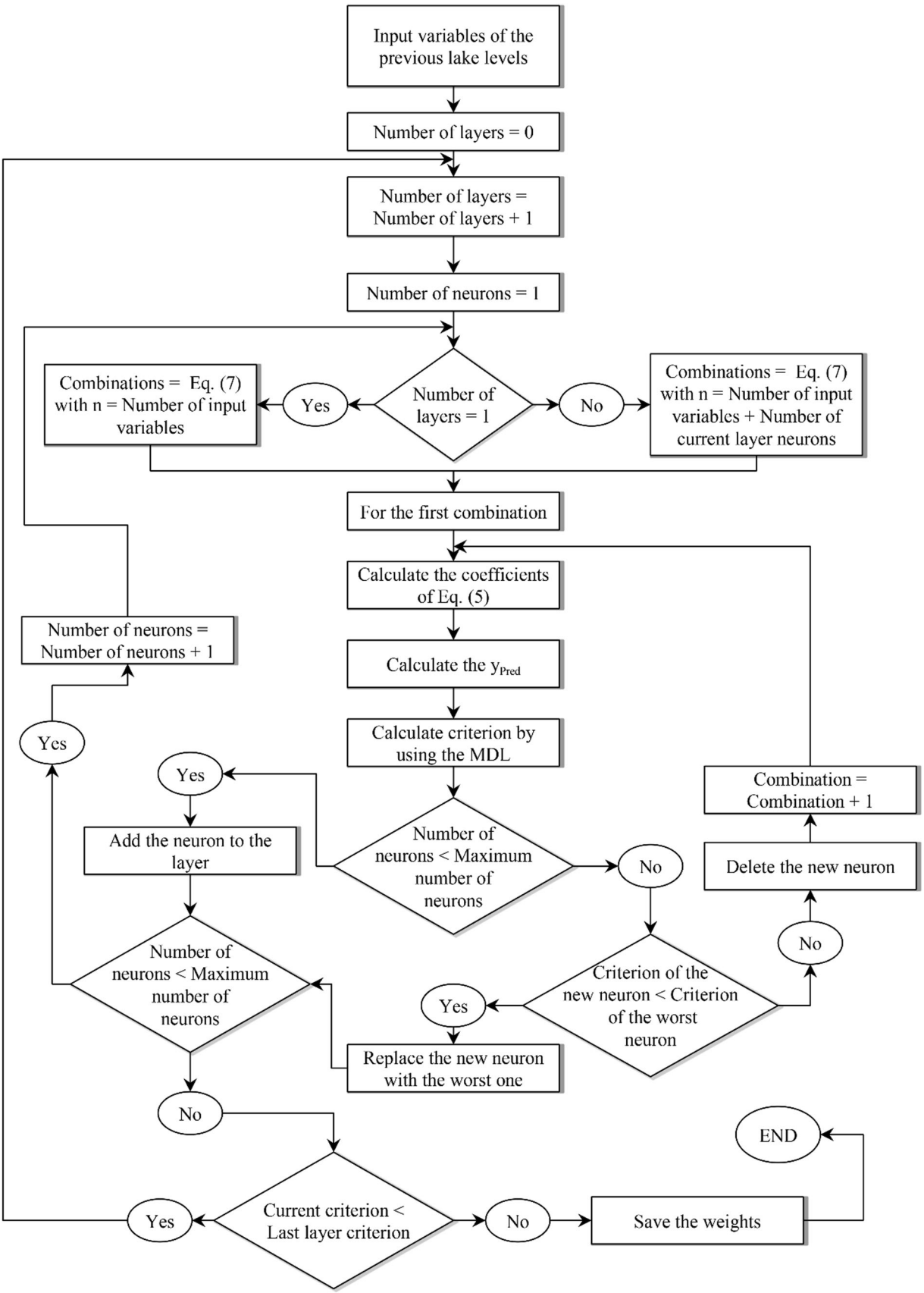
$$\text{SEP\%} = \frac{100}{\bar{o}} \times \text{RMSE} \tag{13}$$

$$\delta\% = \frac{\sum_{i=1}^N |(e_i - o_i)|}{\sum_{i=1}^N e_i} \times 100 \tag{14}$$

$$\text{SD} = \sqrt{\frac{1}{N-1}\sum_{i=1}^N (s_i - \bar{s})^2} \tag{15}$$

$$R^2 = \left[\frac{\sum_{i=1}^n (o_i - \bar{o})(e_i - \bar{e})}{\sqrt{\sum_{i=1}^n (o_i - \bar{o})^2 \sum_{i=1}^n (e_i - \bar{e})^2}}\right]^2 \tag{16}$$

where o_i is the ith observed sample, e_i is the ith estimated sample, $\bar{o}$ is the average of observed samples, $\bar{e}$ is the average of estimated samples, N is the number of samples


Input variables of the previous lake levels
Number of layers = 0
Number of layers = Number of layers + 1
Number of neurons = 1
Number of layers = 1
Yes
Combinations = Eq. (7) with n = Number of input variables
No
Combinations = Eq. (7) with n = Number of input variables + Number of current layer neurons
For the first combination
Calculate the coefficients of Eq. (5)
Calculate the y_{Pred}
Calculate criterion by using the MDL
Number of neurons < Maximum number of neurons
Yes
Add the neuron to the layer
No
Delete the new neuron
Combination = Combination + 1
Criterion of the new neuron < Criterion of the worst neuron
Yes
Replace the new neuron with the worst one
No
Number of neurons < Maximum number of neurons
Yes
Number of neurons = Number of neurons + 1
No
Current criterion < Last layer criterion
Yes
No
Save the weights
END

◀ **Fig. 4** GMDH neural network flowchart

in the dataset, s_i is the ith sample and $\bar{s}$ is the average of the samples studied.

Results

In this study, GMDH-MDL was used to model daily reservoir water levels. The first part of this section explains the input combinations considered and the goal of the numerical model. The second and third parts investigate the efficiency of the first and second input combination groups. The fourth part comprises the study results. Finally, the last part examines the accuracy of the best input combinations determined previously with the non-observed dataset in predicting reservoir water levels.

GMDH-MDL model input combinations

The purpose of the numerical models in the current study is to forecast 7-day ahead reservoir water levels (L_{+7}). It should be mentioned that the entire dataset for Chahnimeh#1 contains 1392 samples. The first 730 samples from this dataset were considered for model training, and the remaining 662 samples were considered for model testing. Two suitable input combination groups entail recent-days input combinations (i.e., L, L_{-1}, L_{-2}) and recent-weeks input combinations (i.e., L, L_{-7}, L_{-14}). Four input combinations were used in this study: two in the first group and two in the second group. The GMDH-MDL models developed with each input combination are given in Table 3.

Figure 5 illustrates the GMDH-MDL Model#1 structure. This model has a two-neuron input layer, one hidden layer and one output layer. Equation (17) is used by Model#1 to predict 7-day ahead reservoir water levels. This model converges rapidly and has a simple structure.

Table 3 Input combinations considered

Model name	Group name	Input combinations	Forecasting target
Model#1	Group#1	L, L_{-1}	L_{+7}
Model#2	Group#1	L, L_{-1}, L_{-2}	L_{+7}
Model#3	Group#2	L, L_{-7}	L_{+7}
Model#4	Group#2	L, L_{-7}, L_{-14}	L_{+7}

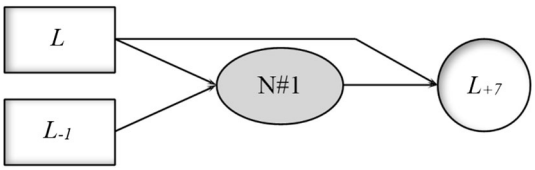


Fig. 5 GMDH-MDL Model#1 structure

$$L_{+7} = 921.94 - 39.18 \times N\#1 + 36.39 \times L + 1.51 \times L \times N\#1 - 0.71 \times N\#1 \times N\#1 - 0.79 \times L \times L \quad (17)$$

$$N\#1 = -302.09 + 441.50 \times L - 439.23 \times L_{-1} - 17.73 \times L_{-1} \times L + 8.42 \times L \times L + 9.31 \times L_{-1} \times L_{-1} \quad (17-1)$$

Increasing the number input variables raises the model complexity. Thus, according to Fig. 6, the structure size of Model#2 is greater than Model#1, and the model has three hidden layers and six hidden neurons in total. The equations for predicting 7-day ahead reservoir water levels based on the six hidden neurons in the model are denoted by Eqs. (18).

$$L_{+7} = 66.44 + 11.89 \times N\#6 - 11.16 \times L_{-1} - 0.21 \times L_{-1} \times N\#6 + 0.09 \times N\#6 \times N\#6 + 0.11 \times L_{-1} \times L_{-1} \quad (18)$$

$$N\#1 = -316.73 + 570.89 \times L_{-1} - 568.56 \times L_{-2} - 27.41 \times L_{-2} \times L_{-1} + 13.12 \times L_{-1} \times L_{-1} + 14.28 \times L_{-2} \times L_{-2} \quad (18-1)$$

$$N\#2 = -75.32 + 254.10 \times L - 252.76 \times L_{-2} - 3.55 \times L_{-2} \times L + 1.51 \times L \times L + 2.03 \times L_{-2} \times L_{-2} \quad (18-2)$$

$$N\#3 = -302.09 + 441.50 \times L - 439.23 \times L_{-1} - 17.73 \times L_{-1} \times L + 8.42 \times L \times L + 9.31 \times L_{-1} \times L_{-1} \quad (18-3)$$

$$N\#4 = 921.94 - 39.18 \times N\#3 + 36.39 \times L + 1.51 \times L \times N\#3 - 0.71 \times N\#3 \times N\#3 - 0.79 \times L \times L \quad (18-4)$$

$$N\#5 = -67.35 - 71.99 \times N\#2 + 73.27 \times N\#1 + 0.08 \times N\#1 \times N\#2 + 0.03 \times N\#2 \times N\#2 - 0.11 \times N\#1 \times N\#1 \quad (18-5)$$

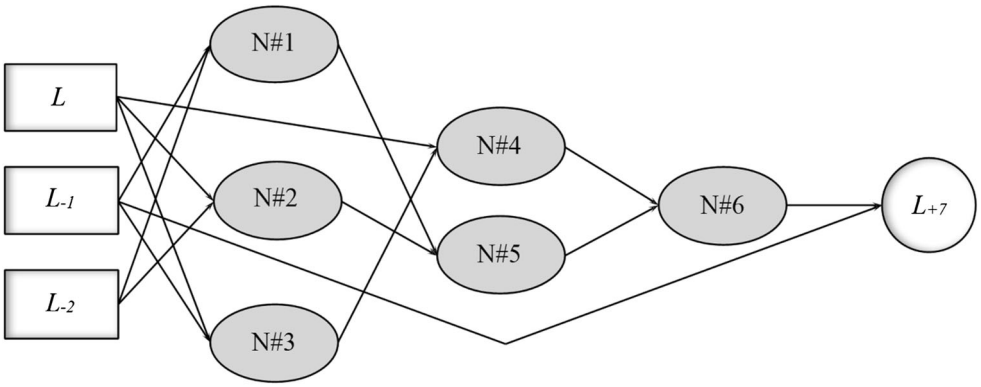


Fig. 6 GMDH-MDL Model#2 structure

$$N\#6 = 68.92 + 5.08 \times N\#5 - 4.36 \times N\#4 + 0.75 \times N\#4 \times N\#5 - 0.38 \times N\#5 \times N\#5 - 0.37 \times N\#4 \times N\#4 \quad (18-6)$$

Similar to Model#1, due to the simple input layer of Model#3 that uses L and L_{-7} water levels as input neurons, the GMDH-MDL structure of this model is very simple. According to Fig. 7, the current GMDH-MDL has no hidden layers, and therefore, the equation that represents the model is very simple (Eq. 19).

$$L_{+7} = 716.10 + 69.36 \times L - 71.26 \times L_{-7} + 0.03 \times L_{-7} \times L - 0.08 \times L \times L + 0.05 \times L_{-7} \times L_{-7} \quad (19)$$

The GMDH-MDL structure of Model#4 is shown in Fig. 8. This model has two hidden layers and three hidden neurons in total. Comparing Figs. 5, 6, 7 and 8 signifies that increasing the number of input variables can significantly affect GMDH-MDL model complexity, i.e., Model#1 and Model#3 with two input variables are much simpler than Model#2 and Model#4 with three input variables. The performance of each model is addressed in the subsequent sections.

$$L_{+7} = -59.90 - 1.18 \times N\#3 + 2.42 \times L_{-14} + 0.057 \times L_{-14} \times N\#3 - 0.02 \times N\#3 \times N\#3 - 0.03 \times L_{-14} \times L_{-14} \quad (20)$$

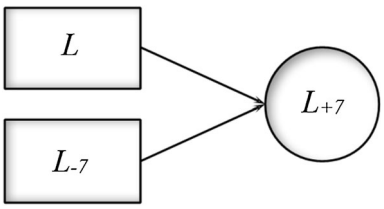


Fig. 7 GMDH-MDL Model#3 structure

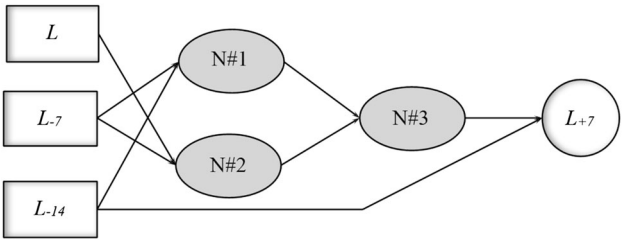


Fig. 8 GMDH-MDL Model#4 structure

$$N\#1 = 3124.45 + 117.13 \times L_{-7} - 128.84 \times L_{-14} + 0.71 \times L_{-14} \times L_{-7} - 0.47 \times L_{-7} \times L_{-7} - 0.22 \times L_{-14} \times L_{-14} \quad (20-1)$$

$$N\#2 = 716.10 + 69.36 \times L - 71.26 \times L_{-7} + 0.031 \times L_{-7} \times L - 0.08 \times L \times L + 0.05 \times L_{-7} \times L_{-7} \quad (20-2)$$

$$N\#3 = -246.64 + 30.67 \times N\#2 - 28.66 \times N\#1 - 0.27 \times N\#1 \times N\#2 + 0.10 \times N\#2 \times N\#2 + 0.16 \times N\#1 \times N\#1 \quad (20-3)$$

Group#1 input combinations

In this section, the input combinations in Group#1 are used to evaluate the GMDH-MDL models. Table 4 lists the statistic errors of Model#1 and Model#2. It can be concluded that both Model#1 and Model#2 with $\%\delta$ of 0.0421 and 0.0424, respectively, performed very well with the testing dataset. However, Model#1 with *RMSE* of 0.3478 for the testing dataset slightly outperformed Model#2 with RMSE of 0.3767. The lower RMSE (0.1845) for Model#2 in training compared with Model#1 (0.1971) and the higher RMSE (0.3767) for Model#2 in testing compared with Model#1 (0.3478) signify that Model#2 exhibited slight overtraining.

Table 4 Statistics for Group#1 input combinations

Model name	Dataset	Input variables	RMSE	MAE	SEP	%δ
Model#1	Training	L, L_{-1}	0.1971	0.0944	0.0404	0.0193
	Testing	L, L_{-1}	0.3478	0.2058	0.0712	0.0421
Model#2	Training	L, L_{-1}, L_{-2}	0.1845	0.0869	0.0378	0.0178
	Testing	L, L_{-1}, L_{-2}	0.3767	0.2069	0.0771	0.0424

The residual scatterplots of the input combinations for Model#1 and Model#2 during training and testing are shown in Fig. 9. The horizontal axis indicates the sample numbers, and the vertical axis denotes the residual of each sample predicted by GMDH compared with each sample observed with Eq. (10). The residuals are analyzed by using SD (Eq. 15). By definition, 95% of all samples are located between 2 × SD and − 2 × SD. Therefore, 95% of samples in Fig. 9 are delimited by two gray lines. According to Fig. 9, both Model#1 and Model#2 were reasonably accurate during testing and training. However, Model#1 seemed to perform better in testing. As mentioned before, the closeness between testing and training datasets is an advantage for a model. Thus, although Model#2 performed better in training, Model#1 seemed to predict 7-day ahead reservoir water levels better.

Scatterplots of Model#1 and Model#2 during testing are illustrated in Fig. 10. Here, the $y = ax + b$ trendline is denoted by a black line. If a is closer to 1 and b is closer to 0, the scatter is closer to the exact line with equation $y = x$. In Fig. 10, it is clear that the trend lines of Model#1 and Model#2 fit almost completely to the exact lines; thus, the models performed similarly. Nonetheless, the greater R^2 indicates that Model#1 performed slightly better.

Group#2 input combinations

This section evaluates the performance of the input combinations in Group#2. The performance of Model#3 and Model#4 is presented in Table 5. According to this table, Model#3 with %δ of 0.0464 in testing outperformed Model#4 with %δ of 0.0519.

The residual scatterplots for Model#3 and Model#4 are represented in Fig. 11. Similar to Fig. 9, the 2 × SD appears here as a gray line. The results demonstrate that 95% of Model#3 residuals were limited between 0.774 and

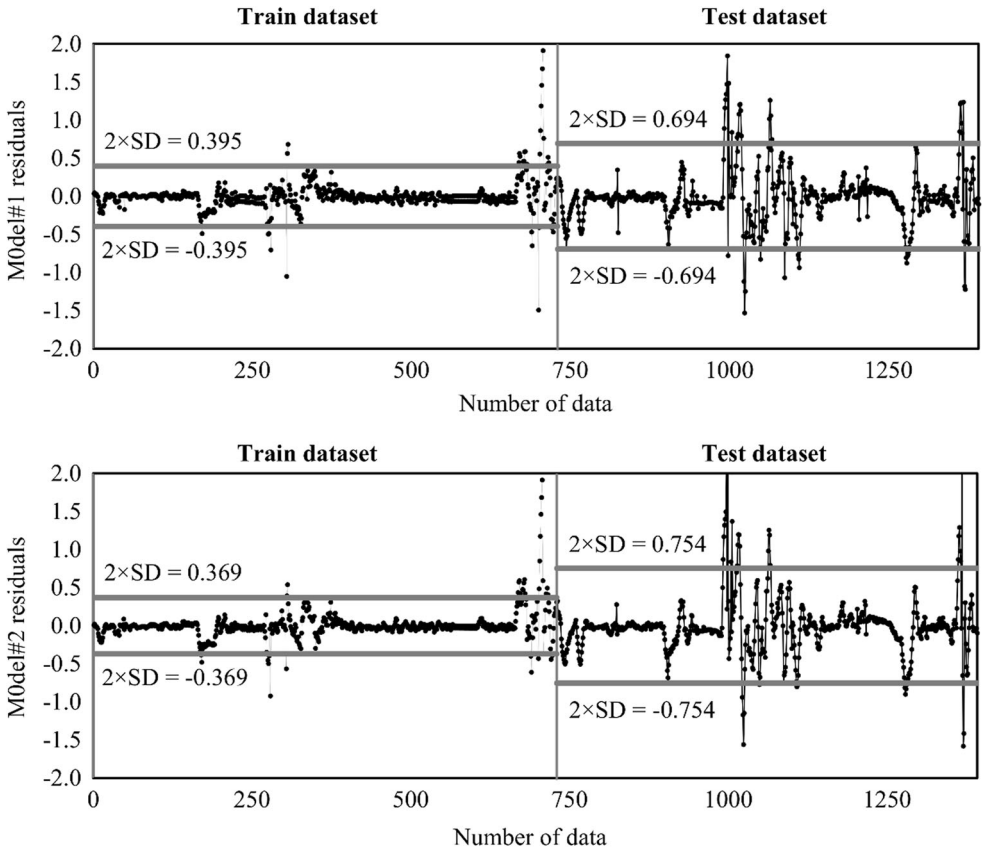


Fig. 9 Residual scatterplots of Model#1 and Model#2 during training and testing

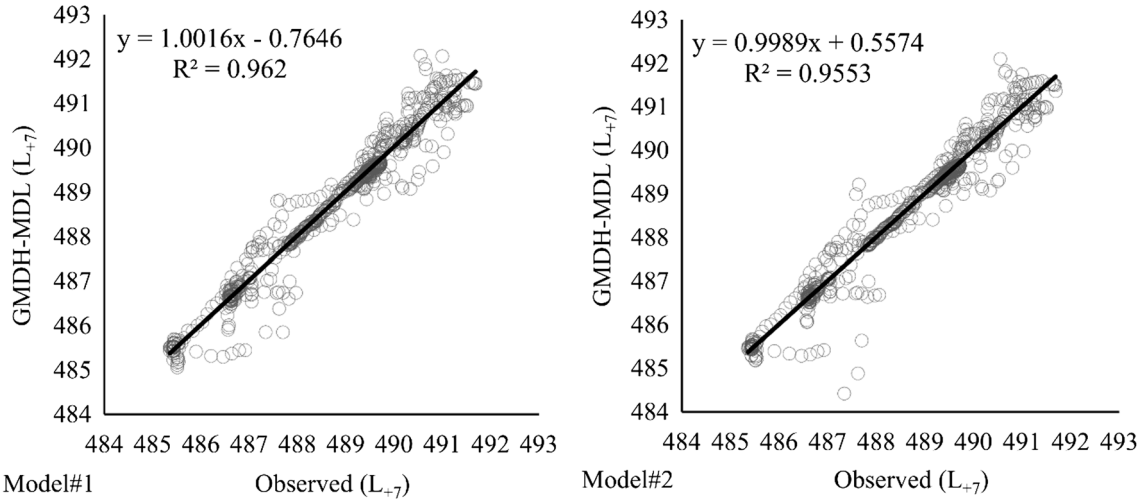


Fig. 10 Scatterplots of Model#1 and Model#2 during testing

Table 5 Statistics for Group#2 input combinations

Model name	Dataset	Input variables	RMSE	MAE	SEP	%δ
Model#3	Training	L, L_{-7}	0.2011	0.0971	0.0412	0.0199
	Testing	L, L_{-7}	0.3866	0.2269	0.0791	0.0464
Model#4	Training	L, L_{-7}, L_{-14}	0.1908	0.0933	0.0391	0.0191
	Testing	L, L_{-7}, L_{-14}	0.4378	0.2534	0.0896	0.0519

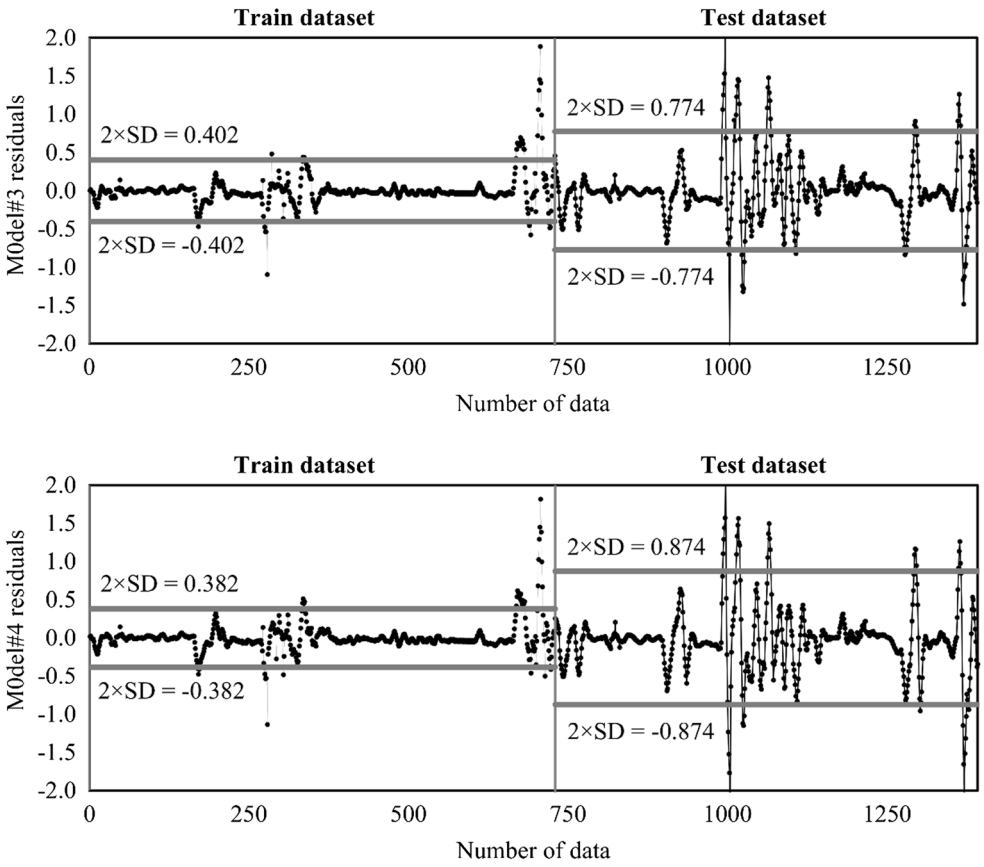


Fig. 11 Residual scatterplots of Model#3 and Model#4 during training and testing

− 0.774. Hence, this model outperformed Model#4 where 95% of residuals were limited between 0.874 and − 0.874. In addition, the similar training and testing values for Model#3 signify it was not trapped in overtraining.

The scatterplots of Model#3 and Model#4 during testing are illustrated in Fig. 12. In terms of the trendline equation $y = ax + b$, Model#4 with a close to 1 (0.9885) and b close to 0 (5.6097) exhibited scattering near the exact line.

Comparison of the input combinations

The results of the input combinations in Group#1 are compared with those in Group#2. Evidently, the model that used previous days' reservoir water levels (Group#1) as an input combination performed significantly better than with previous weeks' water levels (Group#2). An overview of the RMSE statistics for Model#1 to Model#4 is shown in Fig. 13. During testing, Model#1 performed significantly better than the other models. During training, Model#2 performed the best. Overall, it can be concluded that the models in Group#1 outperformed the models in Group#2, and it is better to use recent days in the input combination than recent weeks to forecast 7-day ahead reservoir water levels.

Evaluation of model performance with the non-observed dataset

One of the most common ways to examine the trustworthiness of numerical models in practical situations is to model a non-observed dataset and calculate the models' accuracy. The performance of the best group of GMDH-MDL models developed is evaluated in this section using the Chahnimeh#4 dataset. Model#1 and Model#2 were employed to forecast Chahnimeh#4 reservoir water levels

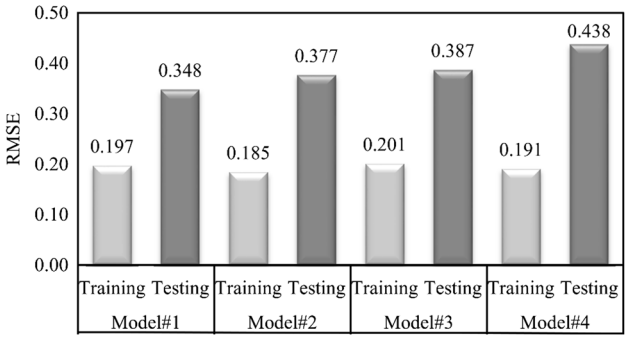


Fig. 13 Overview of model RMSE during testing and training

7 days ahead. The statistics of these two models are given in Table 6, which indicates that Model#2 with RMSE of 0.1239 performed much better than Model#1 with RMSE of 0.1309. Scatterplots of Model#1 and Model#2 for Chahnimeh#4 Reservoir water level forecasting are represented in Fig. 14. The trendline equation in this figure ($y = ax + b$) signifies that Model#2 with a of 0.9761 and b of 11.662 performed better than Model#1 with a of 0.9717 and b of 13.804.

Proposed method compared with other well-known prediction methods

As mentioned before, GMDH is a powerful regression-based method that is utilized to predict complex problems in a vast range of fields. However, there are additional powerful regression methods that can be used to solve the problem in the present study. Thus, the current section compares the GMDH results with three well-known prediction methods, namely MLP (Haykin and Network 2004), ELM (Huang et al. 2006, 2012) and RBF (Poggio and Girosi 1990). MLP, ELM and RBF are very popular

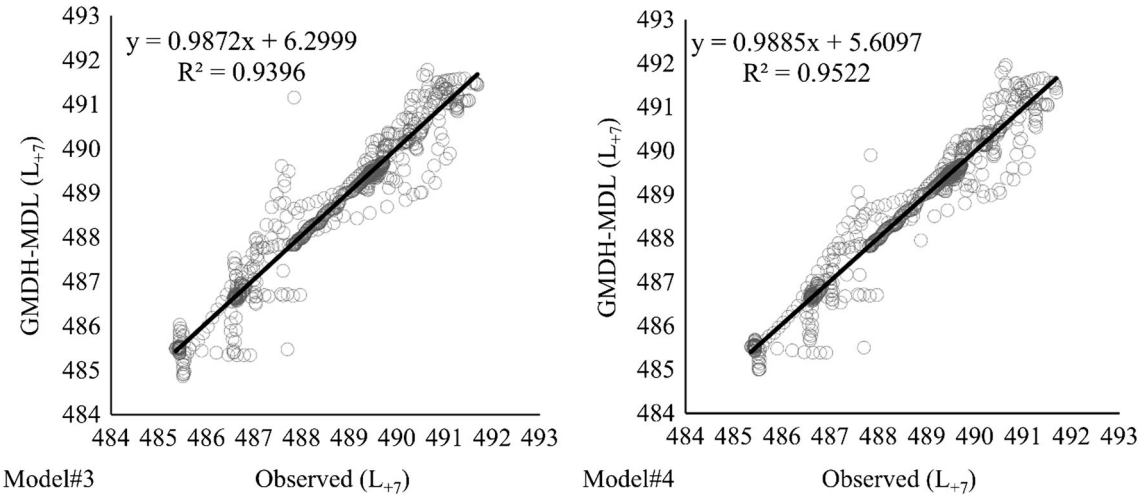


Fig. 12 Scatterplots of Model#3 and Model#4 during testing

Table 6 Statistics for Group#1 input combinations in predicting Chahnimeh#4 Reservoir water levels

Model name	Dataset	Input variables	RMSE	MAE	SEP	%δ
Model#1	Chahnimeh#4	L, L_{-1}	0.1309	0.0898	0.0269	0.0184
Model#2	Chahnimeh#4	L, L_{-1}, L_{-2}	0.1239	0.0787	0.0255	0.0162

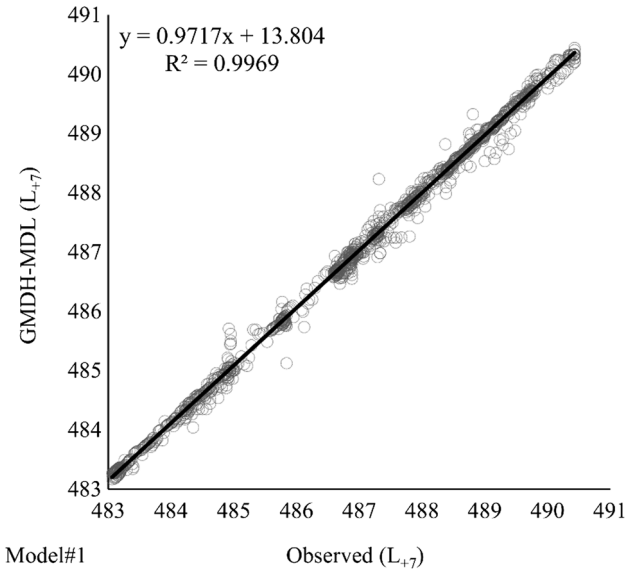


Fig. 14 Scatterplots of Group#1 input combinations in predicting Chahnimeh#4 Reservoir water levels

and have been used extensively in different areas of hydrology (Barzegar et al. 2018; Liu et al. 2018; Roushangar et al. 2018).

In the modeling procedure, all prediction models are run several times with adjustments in the number of hidden layer neurons and other modeling variables. The best modeling result denotes the best performance in predicting with both the testing and training datasets. The modeling input variables were selected according to Model#2 (Table 3), which exhibited superior performance in modeling with the non-observed dataset.

The modeling results are presented in Table 7 for both training and testing datasets. The scatterplots for the testing dataset modeling results are presented in Fig. 15. According to Table 7, it is obvious that all three models performed very similarly in training and testing. Overtraining did not occur in the modeling procedures. In addition, the testing dataset statistical indices for ELM, MLP and RBF (RMSE of 1.011, 0.3767 and 0.4588, respectively) in Table 7 are compared with the testing dataset statistical index of Model#2 (RMSE of 0.3767) in Table 4. Evidently, the proposed GMDH method outperformed ELM and RBF. Comparing Table 4 with 7 indicates that GMDH performed almost the same as MLP. However, modeling in the field of hydrology should be applicable in practical situations. Hence, although MLP and GMDH were very close in terms of performance, GMDH has priority for use in practical situations owing to the explicit solution and equation this method offers. The results of GMDH Model#2 can be calculated easily with Eq. (18), but the MLP method lacks this feature.

Comparing the results in Fig. 15 with the Model#2 plot results in Fig. 10 shows that according to R^2, GMDH outperformed the ELM, MLP and RBF models. Moreover, Fig. 15 shows that MLP and RBF performed much better than ELM in the case study considered.

Table 7 Statistics for the ELM, MLP and RBF models

Method	Dataset	Input variables	RMSE	MAE	SEP	%δ
ELM	Training	L, L_{-1}, L_{-2}	0.7862	0.5001	0.1610	0.1024
	Testing	L, L_{-1}, L_{-2}	1.0115	0.6249	0.2070	0.1279
MLP	Training	L, L_{-1}, L_{-2}	0.2157	0.1077	0.0442	0.0221
	Testing	L, L_{-1}, L_{-2}	0.3767	0.2145	0.0771	0.0439
RBF	Training	L, L_{-1}, L_{-2}	0.1906	0.0970	0.0390	0.0199
	Testing	L, L_{-1}, L_{-2}	0.4588	0.2317	0.0939	0.0474

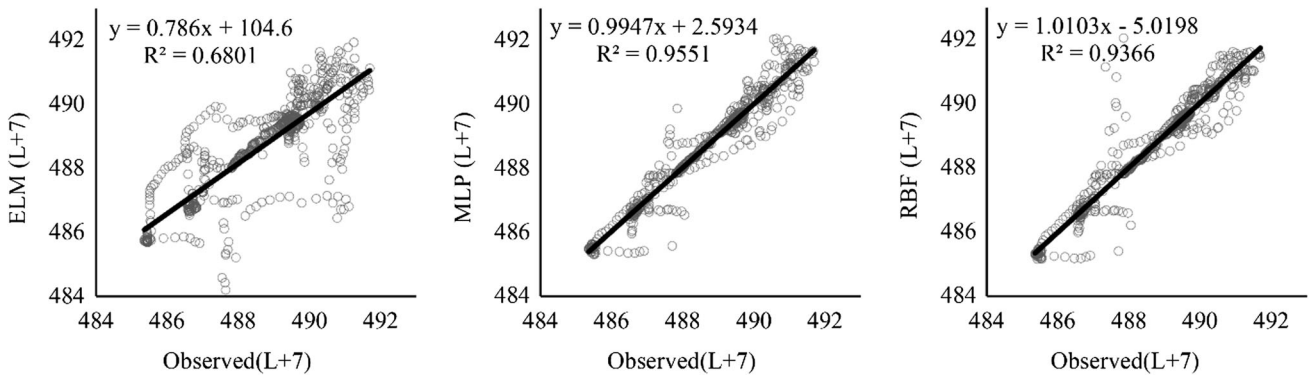


Fig. 15 Scatterplots of ELM, MLP and RBF for the testing dataset

Conclusion

Optimizing multi-purpose reservoir management operations for hydropower generation, and agricultural, industrial and drinking water supplies necessitates the ability to predict reservoir water levels accurately. In this study, the GMDH-MDL method was used to forecast reservoir water levels 7 days ahead. Data from the Chahnimeh#1 Reservoir was used to test and train the models. Four input combinations were considered in two major groups. The first group included reservoir water levels from recent days, while the second group comprised water levels from recent weeks. The results demonstrated that GMDH-MDL can predict daily reservoir water levels very well, and the first group of input combinations outperformed the second group. Subsequently, to identify the model reliability, the first input group was employed with the non-observed dataset from Chahnimeh#4 Reservoir. The most accurate model contained the (L, L_{-1}, L_{-2}) input combination and had RMSE of 0.12 in forecasting the non-observed dataset for Chahnimeh#4 Reservoir.

Compliance with ethical standards

Conflict of interest The authors have no conflict of interest to declare.

References

Altunkaynak A (2007) Forecasting surface water level fluctuations of lake van by artificial neural networks. Water Resour Manag 21(2):399–408

Barzegar R, Moghaddam AA, Adamowski J, Ozga-Zielinski B (2018) Multi-step water quality forecasting using a boosting ensemble multi-wavelet extreme learning machine model. Stoch Environ Res Risk Assess 32(3):799–813

Crapper PF, Fleming PM, Kalma JD (1996) Prediction of lake levels using water balance models. Environ Softw 11(4):251–258

Ebtehaj I, Bonakdari H, Khoshbin F, Azimi H (2015) Pareto genetic design of group method of data handling type neural network for prediction discharge coefficient in rectangular side orifices. Flow Meas Instrum 41:67–74

Ebtehaj I, Bonakdari H, Khoshbin F (2016) Evolutionary design of a generalized polynomial neural network for modeling sediment transport in clean pipes. Eng Optim 48(10):1793–1810

Ebtehaj I, Bonakdari H, Gharabaghi B (2018) Development of more accurate discharge coefficient prediction equations for rectangular side weirs using adaptive neuro-fuzzy inference system and generalized group method of data handling. Measurement 116:473–482

Gholami A, Bonakdari H, Ebtehaj I, Shaghaghi S, Khoshbin F (2017) Developing an expert group method of data handling system for predicting the geometry of a stable channel with a gravel bed. Earth Surf Process 42(10):1460–1471

Gholami A, Bonakdari H, Ebtehaj I et al (2018) A methodological approach of predicting threshold channel bank profile by multi-objective evolutionary optimization of ANFIS. Eng Geol 239:298–309

Gladkov EG, Eletskii VS, Zhabin VF (1991) Prediction of the change in the water level of Lake Sarez and characteristics of seepage through the Usoi barrier. Plenum Publishing Corporation, New York

Grünwald PD, Myung IJ, Pitt MA (2005) Advances in minimum description length: theory and applications. MIT Press, Massachusetts

Guganesharajah K, Shaw EM (1984) Forecasting water levels for Lake Chad. Water Resour Res 20(8):1053–1065

Güldal V, Tongal H (2010) Comparison of recurrent neural network, adaptive neuro-fuzzy inference system and stochastic models in Eğirdir lake level forecasting. Water Resour Manag 24(1):105–128

Haykin S, Network N (2004) Neural networks: a comprehensive foundation. Prentice Hall, Upper Saddle River

Huang GB, Zhu QY, Siew CK (2006) Extreme learning machine: theory and applications. Neurocomputing 70(1–3):489–501

Huang GB, Zhou H, Ding X, Zhang R (2012) Extreme learning machine for regression and multiclass classification. IEEE Trans Syst Man Cybern 42(2):513–529

Iba H, Sato T, de Garis H (1994) System identification approach to genetic programming. In: IEEE world congress on computational intelligence, Orlando, Florida, USA

Ivakhnenko A (1970) Heuristic self-organization in problems of engineering cybernetics. Automatica 6(2):207–219

Ivakhnenko A (1971) Polynomial theory of complex systems. IEEE Trans Syst Man Cybern SMC-1(4):364–378

Kakahaji H, Banadaki HD, Kakahaji A, Kakahaji A (2013) Prediction of Urmia Lake water-level fluctuations by using analytical, linear

statistic and intelligent methods. Water Resour Manag 27(13):4469–4492
Khan MS, Coulibaly P (2006) Application of support vector machine in lake water level prediction. J Hydrol Eng 11(3):199–205
Kisi O, Shiri J, Nikoofar B (2012) Forecasting daily lake levels using artificial intelligence approaches. Comput Geosci 41:169–180
Kisi O, Shiri J, Karimi S et al (2015) A survey of water level fluctuation predicting in Urmia Lake using support vector machine with firefly algorithm. Appl Math Comput 270:731–743
Koppula SD (1980) Forecasting lake levels—a case study. In: National symposium on urban stormwater management in coastal areas, Va Tech, ASCE, New York, USA
Lan Y (2014) Forecasting performance of support vector machine for the Poyang Lake's water level. Water Sci Technol 70(9):1488–1495
Liu H, Sun S, Zheng T, Li G (2018) Prediction of water temperature regulation for spawning sites at downstream of hydropower station by artificial neural network method. Trans Chin Soc Agric Eng 34(4):185–191
Mahdi Hadi R, Shokri S, Ayubi P (2013) Urmia Lake level forecasting using Brain Emotional Learning (BEL). In: 3rd International conference on computer and knowledge engineering, ICCKE 2013, Mashhad, Iran
Najafzadeh M, Azamathulla HM (2013) Group method of data handling to predict scour depth around bridge piers. Neural Comput Appl 23(7–8):2107–2112
Najafzadeh M, Barani GA (2011) Comparison of group method of data handling based genetic programming and back propagation systems to predict scour depth around bridge piers. Sci Iran 18(6):1207–1213
Najafzadeh M, Bonakdari H (2017) Application of neuro-fuzzy GMDH model for predicting the velocity at limit of deposition in storm sewers without deposited beds and under non-cohesive bed load sediment transport conditions. J Pipeline Syst Eng 8(1):06016003-1:8
Najafzadeh M, Lim SY (2015) Application of improved neuro-fuzzy GMDH to predict scour depth at sluice gates. Earth Sci Inform 8(1):187–196
Najafzadeh M, Tafarojnoruz A (2016) Evaluation of neuro-fuzzy GMDH-based particle swarm optimization to predict longitudinal dispersion coefficient in rivers. Environ Earth Sci 75(2):157
Najafzadeh M, Barani GA, Azamathulla HM (2014) Prediction of pipeline scour depth in clear-water and live-bed conditions using group method of data handling. Neural Comput Appl 24(3–4):629–635
Nariman-zadeh N, Darvizeh A, Darvizeh M, Gharababaei H (2002) Modelling of explosive cutting process of plates using GMDH-type neural network and singular value decomposition. J Mater Process Technol 128(1–3):80–87
Nariman-zadeh N, Darvizeh A, Jamali A, Moeini A (2005) Evolutionary design of generalized polynomial neural networks for modelling and prediction of explosive forming process. J Mater Process Technol 164–165:1561–1571
Neyshaburi MR, Bayat H, Mohammadi K, Nariman-zadeh N, Irannejad M (2015) Improvement in estimation of soil water retention using fractal parameters and multiobjective group method of data handling. Arch Agron Soil Sci 61:257–273
Nikolaev NI, Iba H (2001) Accelerated genetic programming of polynomials. Genet Program Evol Mach 2(3):231–257
Ondimu S, Murase H (2007) Reservoir level forecasting using neural networks: Lake Naivasha. Biosyst Eng 96(1):135–138
Poggio T, Girosi F (1990) Regularization algorithms for learning that are equivalent to multilayer networks. Science 247(4945):978–982
Roushangar K, Alizadeh F, Nourani V (2018) Improving capability of conceptual modeling of watershed rainfall–runoff using hybrid wavelet-extreme learning machine approach. J Hydroinform 20(1):69–87
Shaghaghi S, Bonakdari H, Gholami A, Ebtehaj I, Zeinolabedini M (2017) Comparative analysis of GMDH neural network based on genetic algorithm and particle swarm optimization in stable channel design. Appl Math Comput 313:271–286
Shiri J, Shamshirband S, Kisi O et al (2016) Prediction of water-level in the Urmia Lake using the extreme learning machine approach. Water Resour Manag 30(14):5217–5229
Tsai TM, Yen PH, Jiang MQ, Shieh YL (2010) Stream level forecasting in storm period by using self-organization algorithm coupled with distance level relation model. J Chin Inst Civ Hydraul Eng 22(4):363–374
Yadav B, Eliza K (2017) A hybrid wavelet-support vector machine model for prediction of Lake water level fluctuations using hydro-meteorological data. Measurement 103:294–301
Zaji AH, Bonakdari H (2014) Performance evaluation of two different neural network and particle swarm optimization methods for prediction of discharge capacity of modified triangular side weirs. Flow Meas Instrum 40:149–156
Zaji AH, Bonakdari H (2018) Robustness lake water level prediction using the search heuristic-based artificial intelligence methods. ISH J Hydraul Eng. https://doi.org/10.1080/09715010.2018.1424568
Zhang H, Liu X, Cai E, Huang G, Ding C (2013) Integration of dynamic rainfall data with environmental factors to forecast debris flow using an improved GMDH model. Comput Geosci 56:23–31