



Data pre-post processing methods in AI-based modeling of seepage through earthen dams

Elnaz Sharghi ^{a,*}, Vahid Nourani ^{b,d}, Nazanin Behfar ^a, Gokmen Tayfur ^c

^a Faculty of Civil Engineering, University of Tabriz, Tabriz, Iran

^b Center of Excellence in Hydroinformatics, Faculty of Civil Engineering, University of Tabriz, Tabriz, Iran

^c Department of Civil Engineering, Faculty of Engineering, Izmir Institute of Technology, Gulbahcekoyu, Urla, Izmir, Turkey

^d Near East University, Faculty of Civil and Environmental Engineering, 99138, Nicosia, N. Cyprus, via Mersin 10, Turkey

ARTICLE INFO

Article history:

Received 26 January 2019

Received in revised form 17 June 2019

Accepted 13 July 2019

Available online 16 July 2019

Keywords:

Artificial intelligence

Seepage

Ensemble method

Jittering

Mutual information

ABSTRACT

In this paper, seepage of Sattarkhan earthen dam in northwest Iran was simulated using various artificial intelligence (AI) models (e.g., Feed forward neural network, Adaptive neural fuzzy inference system and Support vector regression) and linear ARIMA model based on different input combinations. Both jittering pre-processing and ensembling post-processing methods were also used in order to enhance the performance of the used AI-based data driven methods. For this purpose, various jittered datasets were produced by imposing noises (at different levels) to the original time series to enlarge the training data sample space. Further, three techniques of simple linear, weighted linear and nonlinear neural averaging were considered for pre-post processing purpose. The obtained results indicated that using both jittering and ensembling (especially neural ensemble) enhanced the modeling performance by almost 30% in the testing phase.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

Seepage analysis plays a principle role in different hydraulic, environmental and civil engineering problems. Particularly for earthen dams and embankments, the failure of structure is significantly affected by seepage and therefore, the dam safety draws increasing attention of engineers to accurate and reliable modeling of seepage [1].

Not long ago, Artificial Intelligence (AI) methods have shown significant potential in predicting and simulating nonlinear hydro-environmental time series. AI methods provide efficient approaches for dealing with large numbers of dynamic, nonlinear and noisy data, particularly when the essential physical relations are not exactly comprehended.

Due to the complexity and uncertainty involved in the seepage process, application of AI approaches to model this process may lead to appropriate outcomes. AI models such as Feed Forward Neural Network (FFNN), Adaptive Neural Fuzzy Inference System (ANFIS) and Support Vector Regression (SVR), are black box techniques that have been employed at different fields (e.g. see [2,3]). In the subject of analysis of seepage through earthen dams, Tayfur et al. [4] used the FFNN type of Artificial Neural Network (ANN) for simulating the water heads of piezometers of an earthen dam in Poland. An ANN based temporal modeling tool was connected to a spatial estimator by Nourani and Babakhani [5] to model piezometric heads of an earthen dam. Single, and multi output ANN models were developed by Nourani et al. [6] such that; in the sole case, for the each of piezometers a sole ANN was generated, whilst in the integrated case, just one ANN model should be developed for piezometers at different sections of the earthen dam. In modeling by ANFIS and SVR (nonlinear regression form of the Support Vector Machine-SVM) predictors, which are some other types of AI techniques, a few researches might be perceived when surveying the literature. Novaković et al. [7] employed ANFIS method for predicting of piezometric heads in a dam employing the downstream water heights as input and the piezometric heads as outputs. Finally, Yongbiao [8] used an SVR for analysis of water movement through an earthen dam.

Abbreviations: AI, artificial intelligence; ANN, artificial neural network; ANFIS, adaptive neural fuzzy inference system; ARIMA, autoregressive integrated moving average; BP, back propagation; CDF, cumulative distribution function; R^2 , coefficient of determination; FFNN, feed forward neural network; MI, mutual information; MF, membership function; PDF, probability density function; RMSE, root mean square error; SOM, self-organizing map; GA, genetic algorithm; SVM, support vector machine; SVR, support vector regression.

* Corresponding author at: University of Tabriz, 29 Bahman Ave., Tabriz 5166616471, Iran.

E-mail addresses: sharghi@tabrizu.ac.ir (E. Sharghi), nourani@tabrizu.ac.ir (V. Nourani), gokmentayfur@iyte.edu.tr (G. Tayfur).

For almost all practical issues with the time series, one observation is recorded at any given time. Thus, despite the fact that it may be feasible to enlarge the sample size by differing the length of the observed time series, due to the limited facilities, cost and time saving, there will only be a sole observation on the basic random variable at a time step [9]. For example, for a daily discharge time series, in one day a sample is taken or the average of a few taken samples is reported. In this case, the amount of discharge taken at a given time in one day may vary somewhat with the amount of discharge at other times of the same day. In the training process of the AI models, since each training dataset has a limited sample of all available data, a single dataset cannot represent all patterns of the phenomenon. Due to the limited possibilities and lack of access, there is no possibility to sample large amounts of data to cover different patterns of the phenomena. As such, size of training data can be increased by adding artificial data. Increasing size of dataset using artificial data, which has a pattern similar to the original data, can be considered as a pre-process method to improve training performance of the AI models. The proposed data pre-processing method in this paper is based on the idea that many possible observations may be made at each time. Therefore, for each time series, if sets of noises are created and added to the main time series with the overall pattern similar to the original dataset, samples would be increased. Then, these time series can be used to train AI models. It is possible that training via such data may create more constraints in the training process, which may prevent the network from overtraining [9,10]. The appropriate noise level which should be considered for imposing to the main dataset to generate the jittered time series has not been yet discussed in the previous studies. Jittered time series generated by high size noise could disturb the overall pattern of the time series. But then, low size noise might not lead to an efficient jittering process [11]. Some of scholars have already investigated impact of generating extra training dataset by appending noises to the input data in neural network modeling and its generalization efficiency [9,11–13].

On the other hand, data-driven methods (such as ANN, ANFIS, and SVR) solely or linked to a data pre-processing method may result comparatively dependable outcomes, it is obvious that for an issue at hand various models may result distinct outcomes. Hence, compounding various models employing an ensemble operation tool, various features of the fundamental patterns may be taken more exactly. In the proposed post-processing method in this study as an ensembling approach, classic linear time series modeling tool of ARIMA (autoregressive integrated moving average) was employed in addition to AI methods. The basic object of model combination can be summarized as: Firstly, it is generally complicated in real issues to recognize if the intended time series is following linear or nonlinear pattern or if one special technique is more accurate than other models. In other words, none of the models are the best predictors for all conditions and time intervals, so, by integrating different models, difficulty of opting proper predictor can be overcoming and the unique ability and feature of each model could be employed in more accurate predicting process [14,15]. However linear methods sometimes couldn't lead to accurate outcomes according to their restrictions to manipulate non-stationary and nonlinear behaviors, such methods are yet employed since, i) a linear method is cost-effective, uncomplicated, and the superposition principle holds true via these methods, ii) existing error in the employed dataset (or utilized computational method) rises in a linear manner in such linear method, however it is enlarged nonlinearly in the next steps in nonlinear methods. Therefore, it may be suggested to use a linear modeling tool for the linear sections of a natural phenomenon. Secondly, the natural phenomenon may contain patterns from each of linear and nonlinear behaviors. In this state, neither ARIMA nor AI models can be

proper for modeling and simulating the process because the ARIMA cannot handle the nonlinear behavior while an Als may enlarge the error of linear characteristics. Accordingly, by ensembling ARIMA and Als, intricate autocorrelation patterns in the time series can be identified more accurately. Thirdly, it has been confirmed in most of prior research that no specific model can probe the phenomenon excellently [14–16]. Since often a natural process is naturally complex and each specific method cannot detect the various patterns of the phenomenon. The concept of such model ensembling has been employed at various engineering subjects (e.g., [17,18]).

Although the effect of adding good noise (jittering) and using model ensembling on increasing the accuracy of AI-modeling at different engineering fields has been separately examined, as far as the authors are aware, they have not been discussed in the field of seepage analysis in general and specifically at studying of earthen dams. It should be noticed that utilizing model ensemble was performed by authors of this paper for analysis of seepage of earthen dams [19]; but employing of jittering method individually and linked to model combination did not examined. Thus, it is the main objective of this study to close this gap. Furthermore, it is clear that according to Darcy equation, obtaining piezometric heads, seepage can be achieved. So in this paper it is tried to predict the piezometric heads of 2 piezometers in the core of dam. In this regard it is tried to answer the research questions of (1) how the jittering pre-processing affects the models accuracy. (2) how much is the models' output ensembling effect on improving modeling performance. (3) what is the impact of applying the both of jittering and ensembling on modeling of piezometers' head. In this way; different black box methods including the FFNN (a commonly employed AI method), ANFIS (which can handle the uncertainty of phenomenon employing fuzzy theory) and SVR (more recently used AI model) as well as linear ARIMA models created and trained for modeling seepage in Sattarkhan earthen dam, via three scenarios with different input combinations that mutual information (MI) was employed for proper input selection of AI models. Next different artificially generated datasets patterned similar to the observed time series are generated, then generated time series are fed to the AI models. Then, the ensemble model is created using outcomes of the noted models for each scenario for evaluation of the predicting performance. For this purpose, three combination techniques as linear simple and weighted averaging and nonlinear neural averaging methods are applied to the both scenarios.

2. Materials and methods

2.1. Sattarkhan earthen dam and data

Sattarkhan earthen dam is a reservoir dam impounded on the Ahar Chai River in 15 km southwest of Ahar city in East Azerbaijan Province of Iran. This 78 m high dam's crest is 340 m with reservoir capacity of about $131.5 \times 10^6 \text{ m}^3$. Fig. 1(a) presents a view of the dam. Various automatic piezometers were installed at 4 sections through dam. Water heads in the piezometers were recorded monthly since 20-Apr-99 to 19-Jan-13. As well, the water levels of dam's reservoir were monitored every day but water level in the downstream reservoir is generally negligible.

In current research, measured piezometric heads from piezometers # 207, 212, 216 and 217 at section # 2 (see Fig. 1 (b)) were used for the modeling purpose. Because most head losses occur through the dam's core, most instrument installations and monitoring programs were focused in core of the dams. Therefore, in this paper, the core of the dam was chosen as the study area. The statistics of the employed data tabulated in Table 1 and Fig. 2 as

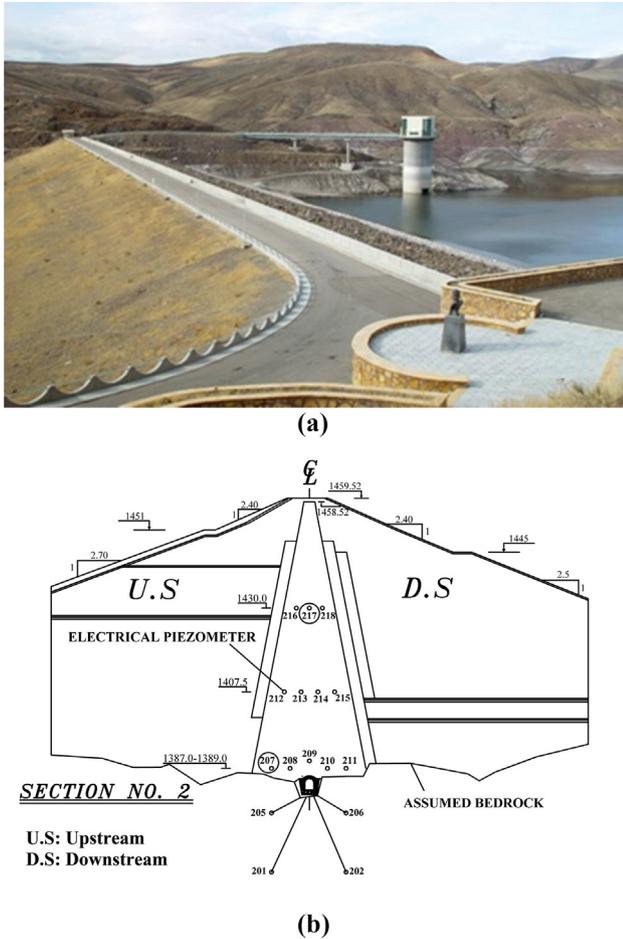


Fig. 1. (a) Sattarkhan earthen dam, (b) Piezometers' positions of cross section # 2.

example, demonstrates the observed heads in piezometers # 207, 217 and dam's reservoir for the considered period. For the training and testing purposes, dataset was spat up to two parts. The first portion of 75% were utilized as training dataset and the second portion of 25% were utilized as the testing dataset.

2.2. Proposed methodology

First, FFNN, SVR, ANFIS and ARIMA methods were individually developed, according to three scenarios. It should be noted that the dominant inputs of AI models were selected by employing MI, as a nonlinear correlation measure. Then, noise time series with various standard deviations were produced and added to the observed time series to generate jittered datasets. The original and jittered time series were used to train the AI models. Finally, the ensemble model was created employing outcomes of the sole models.

The objective of the developed method here was to estimate the heads of piezometers in an earthen dam employing the water level

in upstream reservoir as well as piezometric head measured in the piezometers as inputs of the predictors. To end this, 3 various input scenarios, each of which can be used in a particular operation condition, were considered in this study.

i) Scenario 1

For modeling via the first scenario, it was tried to simulate the piezometric heads of each piezometer employing the prior data in addition to water heads at upstream reservoir of the dam. Therefore, the prediction of *i*th piezometer's water levels can be formulated as:

$$P_t^i = f(P_{t-1}^i, P_{t-2}^i, \dots, P_{t-n}^i, h_t, h_{t-1}, \dots, h_{t-m}) \tag{1}$$

where *i* th piezometer's water level in time *t* supposes to be a function of *i* th piezometer's water level at prior time increments (*t*-1, *t*-2, . . . , *t*-*n*) and upstream's reservoir water head in *t* th time step and prior time increments (*t*-1, *t*-2, . . . , *t*-*m*). Predominant lag steps (*m*, *n*) are detected via trial and error process or a reliable input selection method (in this study MI).

ii) Scenario 2

In scenario 2, levels at the adjacent piezometers are used to predict water level of each piezometer by means that water level of *i* th piezometer depends on the adjacent piezometers data by:

$$P_t^i = f(P_t^j, P_{t-1}^j, \dots, P_{t-o}^j, P_t^k, P_{t-1}^k, \dots, P_{t-r}^k) \tag{2}$$

in which P_{t-o}^j, P_{t-r}^k show respectively time series of *j* th, *k* th piezometers corresponding to *o* and *r* months ago, respectively. MI could be used for recognizing the appropriate related piezometers. In scenario 2 similar to the first scenario, the lags of *o* and *r* could be detected by MI. by comparison of Eqs. (1) and (2) it is clear that in the second scenario to simulate the piezometric heads of piezometers, the piezometric heads from other piezometers with dependable relation to the pointed piezometer can be employed in the absence of prior recorded dataset of that piezometer. Hence, this scenario could be a beneficial strategy in the case of technical issues for the operation of the piezometers, so that the dataset from neighboring piezometers could be used to model and estimate the heads of the pointed piezometers.

iii) Scenario 3

In third scenario, it is tried to estimate the water heads in piezometers employing two another piezometers dataset and also upstream water head data. So, the overall function of scenario 3 was patterned as:

$$P_t^i = f(P_t^j, P_{t-1}^j, \dots, P_{t-o}^j, P_t^k, P_{t-1}^k, \dots, P_{t-r}^k, h_t, h_{t-1}, \dots, h_{t-m}) \tag{3}$$

Predominant piezometers and *m*, *n* (defined in scenarios 1 and 2) could be employed at scenario 3. Since in scenario 3, the model uses two adjacent piezometers dataset and also upstream reservoir water head time series, it is supposed to scenario 3 results more accurate outputs in comparison to the scenario 2.

Table 1 Statistical information of recorded water levels for piezometers of section # 2.

	Reservoir	Piezometer #			
		207	212	216	217
Maximum (m)	1447.21	1439.57	1439.35	1442.73	1438.25
Minimum (m)	1424.47	1424.83	1424.89	1429.63	1430.00
Average (m)	1432.40	1431.27	1431.51	1434.80	1432.74
Standard Deviation (m)	4.44	3.42	3.39	3.54	2.14

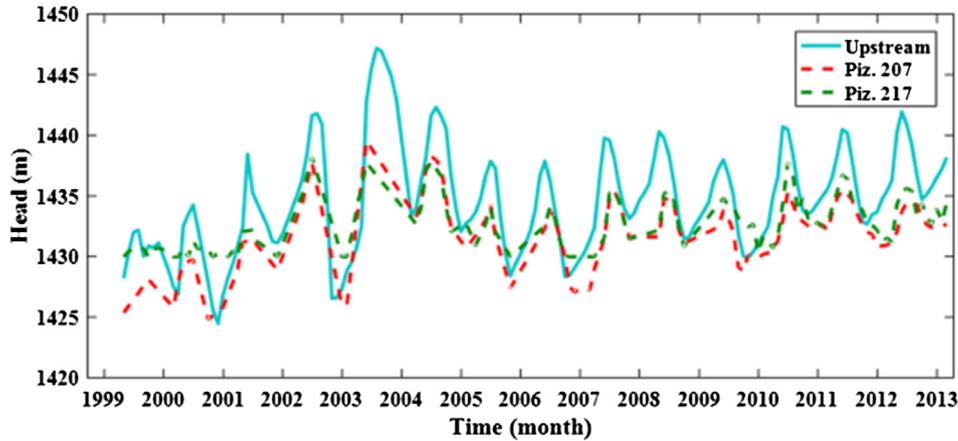


Fig. 2. The recorded water head in piezometer # 207 and piezometer # 217 and upstream water heads.

The purpose of scenarios 2 and 3 is to apply it when some piezometers get out of service or to stimulate the water head where there is no installed piezometer, and the other and more important purpose of scenario 3 is to enhance the modeling performance by utilizing the data of other piezometers and upstream reservoir.

In the following sub-sections, each component of the proposed framework (Fig. 3) is briefly described.

2.2.1. Mutual information (MI)

Information content (I) of a time series Y binned in M bins with probability of p_i for each bin ($i = 1, 2, \dots, M$) is defined as [20]:

$$I(Y) = I(p) = - \sum_{i=1}^M p_i \log p_i \tag{4}$$

The MI value between two-time series of A and B can be computed as [21]:

$$MI(A, B) = I(A) + I(B) - I(A, B) \tag{5}$$

where $I(A)$ and $I(B)$ respectively show the information contents of A and B , and $I(A, B)$ stands for their joint information content as:

$$I(A, B) = - \sum_A \sum_B p_{AB} \log p_{AB} \tag{6}$$

2.2.2. Feed forward neural network (FFNN)

ANN as an AI-based model is a mathematical model aims to handle nonlinear relationship of input-output dataset [22]. Among the different ANN algorithms, FFNN with Back propagation (BP) training is widely applied at different fields of water resources engineering and is the most common class of ANNs (For details see [3,23,24]).

2.2.3. Adaptive neural fuzzy inference system (ANFIS)

Neuro-fuzzy simulation points to the techniques of employing various learning algorithm to fuzzy modeling in the neural network or fuzzy inference system. Every fuzzy system is comprised of three main parts; fuzzy data base, fuzzifier, and defuzzifier [25]. Inference engine and fuzzy rule base are the two main parts of fuzzy data base. Fuzzy rule base involves rules that are related to fuzzy propositions as illustrated by Jang et al. [26]. Consequently, fuzzy inference applied operation analysis. Among many fuzzy inference engines that can be used to achieve this purpose, the Sugeno FIS utilized here [27].

2.2.4. Support vector regression (SVR)

Learning in the context of SVM was proposed and introduced by Cortes and Vapnik [28], which provides a satisfactory approach to the problems of prediction, classification, regression and pattern recognitions. SVM is based on the concept of machine learning

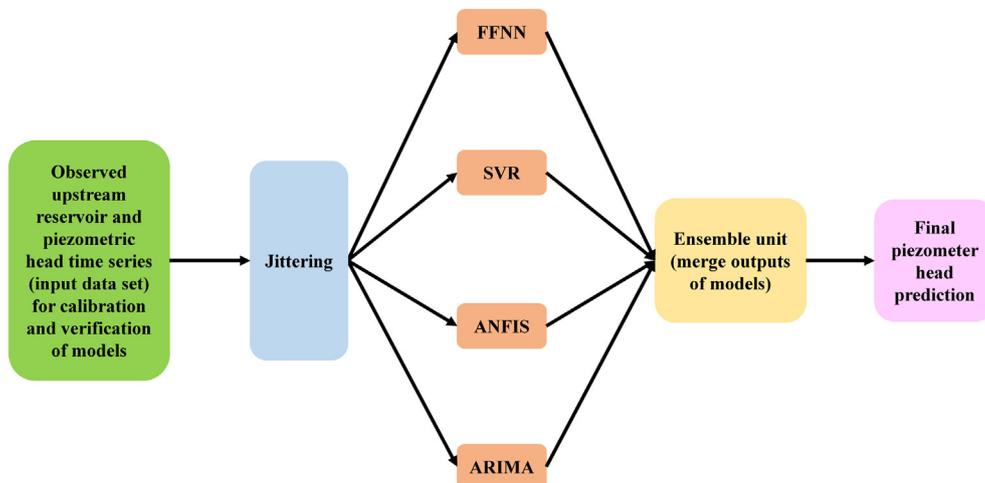


Fig. 3. Schematic of the proposed methodology.

comprises of data driven model. The structural risk minimization and statistical learning theory are two useful functions of SVM, this makes it different from ANN. SVR is a form of SVM in which first by a linear function fits to dataset then a nonlinear kernel is applied to the pervious results to obtain the nonlinear simulation of the target data. For more details, the readers are referred to Wang et al. [29] and Raghavendra and Deka [30].

2.2.5. Auto regressive integrated moving average (ARIMA)

ARIMA is a classic linear predicting and forecasting model, generally employed for experimental problems. The general formula for the of ARIMA can be presented by (p, d, q) that p refers to autoregressive degree, d shows differencing rank, q stands for the degree of moving average (readers are referred to [31]).

2.2.6. Jittered data generation

Typically, generation of a dataset which obeys a desired probability density function (PDF) involves 2 stages. Firstly, a random dataset with uniform PDF is produced. Secondly, the generated random dataset is employed to create the dataset which obeys the desired PDF (e.g. normal PDF). In the second stage, reverse conversion technique can be employed (among some others) for generating random dataset which obeys desired PDF. According to this method, if x obeys cumulative distribution function (CDF) of $F(x)$, then $u = F(x)$ will obey uniform PDF of $u(0,1)$. Inversely for $u \sim u(0,1)$, $x = F^{-1}(u)$ will obey CDF of F . Thus, for generating y as a random variable with desired CDF of G the following Eq. can be used [32]:

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

includes-G-cumulative-distribution

In this study, MATLAB software was employed for generating the random numbers of noises with a desired PDF (with mean of zero and a few small standard deviation values). Also ANN and ANFIS toolbox of MATLAB were used to develop single models of ANN and ANFIS but for SVR a code was developed in MATLAB software.

2.2.7. Ensembling unit

Model ensembling technique is a method to integrate the approach of predictors to improve the final performance [33]. Various studies at different fields suggested to ensemble outcomes of several methods as an effective approach to improve the performance of time series predictions [34].

In this paper, two techniques were utilized for ensembling of the employed models' outcomes to enhance modeling performance as (i) linear ensemble technique; which includes linear ensemble by simple averaging and linear ensemble by weighted averaging. (ii) neural ensemble technique; unlike linear ensemble techniques (Eqs. (8) and (9)), in nonlinear ensemble technique another FFNN is trained to obtain an ensemble output.

Simple averaging is done as:

$$\bar{f}(t) = \frac{1}{N} \sum_{i=1}^N f_i(t) \quad (8)$$

where $\bar{f}(t)$ is outcome of simple ensemble technique, N shows the number of single models (in this study, $N = 4$) and $f_i(t)$ stands for the outcome of the i th method (i.e. ANN, ANFIS, SVR and ARIMA).

The weighted averaging technique is formulated as:

$$\bar{f}(t) = \sum_{i=1}^N w_i f_i(t) \quad (9)$$

where w_i shows imposed weight on the output of i th method that may be computed on the basis of the performance measure of i th method as:

$$w_i = \frac{R_i^2}{\sum_{i=1}^N R_i^2} \quad (10)$$

where R_i^2 measures the i th model performance (such as determination coefficient).

In the nonlinear neural combination technique, the outputs obtained by sole models (FFNN, ANFIS, SVR and ARIMA) are integrated together as inputs to create a new model and train via FFNN technique to produce the ensemble output (see Fig. 4).

2.3. Performance criteria

To analyze and determine the performance of the models proposed, the Coefficient of Determination (R^2) and Root Mean Square Error (RMSE) utilized in this paper to check the performance of the methods. The Equations for R^2 and RMSE are given by [23]:

$$R^2 = 1 - \frac{\sum_{i=1}^n (o_{obs_i} - o_{com_i})^2}{\sum_{i=1}^n (o_{obs_i} - \bar{o}_{obs})^2} \quad (11)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (o_{obs_i} - o_{com_i})^2}{n}} \quad (12)$$

where n , o_{obs_i} , \bar{o}_{obs} and o_{com_i} are respectively the number of records, recorded data, mean of the recorded values, and predicted values. R^2 is among $-\infty$ and 1, with best grade of 1.

3. Results and discussion

In this section, first results of individual models without any data processing are presented. Then the results of single models employing jittered data following the results of ensembling models without pre-processing method are summarized. Finally results of employing both pre-post processing methods are described and evaluated.

3.1. Results of individual models without data processing

At first, FFNN, ANFIS, SVR and ARIMA models without any data processing were employed to model the heads of piezometers 207 and 217 according to scenario 1. In scenario 1, MI was used to detect the appropriate input data. Water levels of piezometers at any time (t) is related to the water levels at previous time step as well as the water levels in dam's reservoir at both t and $t-1$. Each FFNN was trained with scheme of scaled conjugate gradient of back propagation approach which is fast and avoids a time consuming search [35,36]. Furthermore, there are some kinds of popular transfer functions can be used in BP network. In real problems, the

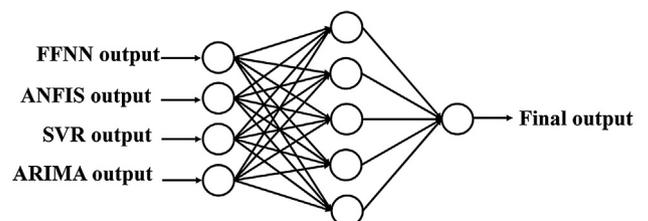


Fig. 4. Schematic of proposed neural ensemble model.

transfer function is usually selected to be a bounded, continuous, and non-constant function. In this research tan-sig transfer function was applied for both intermediate and target layers of FFNN models. Because their nonlinearity in combination with each other can be able to identify far more complex patterns [37,38]. The obtained results of the best structures (obtained by trial and error) for piezometer 207 and piezometer 217 are tabulated in Table 2.

For the ANFIS modeling Sugeno FIS engine used in the modeling framework. In this paper, Gaussian and Trapezoidal-shaped membership functions could lead to reliable outcomes in modeling of piezometric heads for piezometers # 207 and 217, respectively. Because PDF of piezometers # 207 and 217's dataset follows Gaussian or/and semi-Gaussian distribution thus, Gaussian and Trapezoidal MFs showed higher performance in modeling these piezometers. The number of training iteration epoch as well as the No. of MFs were examined to determine the best ANFIS architecture. (see Table 2).

Furthermore, SVR model was developed on the basis of RBF (Radial Basis Function) kernel function for both noted piezometers. Several studies have already reported more reliable results of SVR model using RBF kernel with regard to using other kernels maybe due to its smoothness assumption [39]. The SVR parameters were obtained by direct search method [40] (see Table 2).

At last, ARIMA method developed for piezometers # 207 and 217. Trial and error procedure was employed to define the ARIMA models' parameters with best performance, for both piezometers # 207 and 217 (see Table 2).

Between different models created, AI models' results were more precise than the linear ARIMA model. As it is obvious, despite of utilizing 5 lag times in training of the ARIMA method, the AI methods using 3 input variables led to better performance with regard to the ARIMA method. The lower performance of the classic ARIMA method in comparison with AI-based models could be connected with the linear nature of the ARIMA and its shortcoming in modeling nonlinear phenomenon like seepage. Among AI models, the performance of FFNN and SVR models was a bit better than ANFIS model in testing step.

Piezometer 207 is the nearest piezometer to the reservoir of the dam and is effected mainly by the fluctuations of the upstream water heads, while piezometer # 217 located in topmost elevation of the core and mid-section and it is far from reservoir of the dam

to parallel with the variation of the dam's reservoir water head instantly. Furthermore, because of soil friction the variations might be disappeared within the dam, and oscillations of water levels of the dam's reservoir don't significantly impact this piezometer's water levels. Due to using reservoir water heads time series for training of AI models and because of the noted reasoning, the performance of AI models for piezometer 217 was less precise than piezometer # 207 (see Table 2).

At second scenario, MI employed to find the effective piezometers among piezometers of Sec. 2 and piezometers # 207, 217. Thus, piezometer # 212, piezometer # 216 utilized in modeling of piezometer # 207. For modeling piezometer # 217 the data of piezometer # 212 and piezometer # 216 were employed. Hence, data of two piezometers at time steps t and $t-1$ were utilized as 4 input data to calibrate AI models.

The modeling by third scenario is similar to Scenario 2, but a little more complicated. In scenario three for modeling the piezometers at time t , reservoir water head in time steps t and $t-1$ as well as two other piezometers' heads in step t utilized to develop AI methods. The outputs of FFNN, ANFIS, SVR and ARIMA models of piezometer 207, piezometer 217 obtained in scenario two and scenario three have been presented in Table 2. In this Table, the outputs of models with optimum performance have been tabulated. α - β - γ in architecture of an FFNN denotes to the number of input neurons (α), intermediate neurons (β) and of target neurons (γ). At ANFIS architecture, MF- α denotes to applied MFs and number of MFs (α). On the other hand, α, β, γ in SVR model indicate the kernel parameter (α), approximation accuracy (β) and constant parameter (γ), finally numbers of (α, β, γ) in ARIMA model refer to order of autoregressive component (α), the number of differencing operation (β) and order of moving average component (γ). It should be noticed that since in ARIMA model just the prior data of each piezometer were utilized in the modeling so, scenarios 2 and 3 couldn't be defined for ARIMA model.

Regarding results of three considered scenarios, for scenario two dataset of each piezometer was not utilized as input parameters, performance of scenario two because of using synchronic data with targets and employing the datasets of 2 other piezometers for modeling is a bit more accurate than scenario 1. In scenario 3 due to utilizing synchronous data with targets and using 2 other piezometers as well as dam's reservoir water head datasets, the performance is

Table 2
Sole models' results for scenarios 1, 2 and 3.

Scenario #	Piezometer #	Model	Model structure ^a	R ²		RMSE (normalized)	
				Training	Testing	Training	Testing
1	207	FFNN	3-8-1	0.8588	0.7166	0.0612	0.0357
		ANFIS	Gaussian-3	0.8859	0.6758	0.0534	0.0373
		SVR	0.333, 0.01, 15	0.8453	0.6935	0.0669	0.0382
		ARIMA	(5,2,4)	0.7205	0.5440	0.0741	0.0558
	217	FFNN	3-5-1	0.7566	0.6865	0.1098	0.1045
		ANFIS	Trapezoidal-2	0.7856	0.6101	0.1002	0.1180
		SVR	0.333, 0.1, 30	0.7229	0.6467	0.0944	0.1212
		ARIMA	(5,2,4)	0.6702	0.5115	0.1172	0.1363
2	207	FFNN	4-2-1	0.8597	0.7645	0.0460	0.0492
		ANFIS	Gaussian-2	0.8718	0.7319	0.0385	0.0549
		SVR	0.5, 0.01, 15	0.8500	0.7286	0.0422	0.0583
	217	FFNN	4-10-1	0.7852	0.6906	0.1003	0.1216
		ANFIS	Trapezoidal-2	0.7752	0.6533	0.0558	0.1351
		SVR	0.33, 0.01, 15	0.7890	0.6807	0.0654	0.1186
3	207	FFNN	4-5-1	0.8709	0.7800	0.0391	0.0286
		ANFIS	Gaussian-2	0.8657	0.7211	0.0425	0.047
		SVR	0.333, 0.01, 15	0.8571	0.7700	0.0476	0.038
	217	FFNN	4-2-1	0.7909	0.7100	0.0658	0.0967
		ANFIS	Trapezoidal-2	0.7700	0.6707	0.0787	0.1169
		SVR	0.25, 0.2, 5	0.8080	0.6901	0.0647	0.1077

^a The result has been presented for the best structure.

more accurate than both scenarios 1 and 2. Therefore generally, for modeling the seepage scenario 1 could be employed and whenever some piezometers get out of service, scenario 2 could be helpful. In addition, scenario 3 could be utilized to obtain better outcomes with more complicated modeling structure.

As it can be seen from the results presented in Table 2, the outcomes of the models are not quite so accurate, and the results are somehow overtrained that it can be due to lack of data. This issue may be solved by jittering method that generates artificial data and produces more data patterns artificially.

According to the outputs of sole models (for example see Fig. 5), it is obvious that in different parts of time series, some of models led to overestimates and others down estimations of the recorded values. Therefore, it is deducible that each method has some restrictions and superiority in predicting by various scenarios for various piezometers. Therefore, by ensembling various models, modeling performance could be enhanced over sole models. In the next step, jittering data pre-processing and three ensembling post-processing methods (described in Section 2.2.7) were used to train and combine the outcomes of sole models to enhance the prediction performance for each scenario.

Because of myriad number of parameters of ANFIS model and a bit less accuracy with regard to the other AI models, this model was not used in the data pre-processing step.

3.2. Results of models linked to jittering data pre-processing

To create jittered dataset with an overall pattern like to the main data, firstly the PDFs of data for the considered piezometers were investigated. Upstream reservoir water level and water levels in piezometers # 207 and 212 obey normal PDF but water levels in piezometers # 216 and 217 obey semi-normal PDF because they are located at the highest elevation of the dam and are not affected by low levels of upstream water levels. Hence, various noise time series were generated with zero mean and different standard deviations with normal distribution for upstream reservoir water level, piezometers # 207 and 212 and with semi-normal distribution for piezometers # 216 and 217. Then the generated noise series were added to their corresponding values in original time series so that more time series with the properties and patterns similar to the original time series were produced. It should be noted that noises with mean of zero and standard deviation values of 0.005, 0.01, 0.05 and 0.1 (normalized value) were produced. Then for each time series with specified standard deviation, 3 data series were produced. For instance, three jittered time series produced for piezometer 207 by noises with standard deviation value of 0.01

and three jittered time series produced for piezometer 217 by noises with standard deviation value of 0.1 are presented in Fig. 6(a) and (b), respectively.

Then the generated time series and original time series were employed as inputs to train the AI models. Given that the FFNN and SVR models had better results in the testing step in almost all cases of the first step, and because of myriad number of parameters of ANFIS model and its low performance with the large number of inputs, in this step the generated and main time series were utilized as inputs of the FFNN and SVR models. Thus firstly FFNN and SVR models designed based on scenario 1. Since scenario 1 has 3 original input time series, so in this step AI models have 12 inputs. In FFNN model like first step, applying tan-sig transfer function for both intermediate and target layers, the framework was calibrated employing scaled conjugate gradient method of the BP approach to find the optimum architecture and training iteration epoch. Also in SVR model like first step, the RBF kernel was used and by tuning the corresponding parameters, a model with optimal performance was obtained. Results of this step for both piezometers 207 and 217 via scenario 1 for FFNN and SVR models are presented in Table 3.

By comparing the results of ANN and SVR models using jittered data and the results of single models of first step, it is clear that the jittered data produced by noise series with standard deviation value of 0.005 had no effect on the performance of the models. This result indicates that noises with lower standard deviation values could not lead to an effective modeling. But by increasing the noise levels up to 0.01 and 0.05, the performance of models improved and in the standard deviation of 0.05 the optimal models were obtained. However, increasing the noise levels up to 0.1 reduced the performance of the models, which indicates that when the noise series with high standard deviation are selected, it can disrupt the patterns between the data.

In the following, according to results of scenario 1, FFNN and SVR models created and trained based on scenarios 2 and 3 using the jittered data produced by noise series with standard deviation value of 0.05 which had the best results. The results obtained from the FFNN and SVR models for the best standard deviation for scenarios 2 and 3 are presented in Table 4.

According to Table 4, in scenarios 2 and 3 similar results to scenario 1 have been obtained. In other words, in scenarios 2 and 3, the jittered data produced by noise with standard deviation value of 0.05 increased the accuracy and performance of modeling because of the mentioned reasons.

Overall, employing jittered data that produced by adding noises with standard deviation value of 0.05 improved the performance of

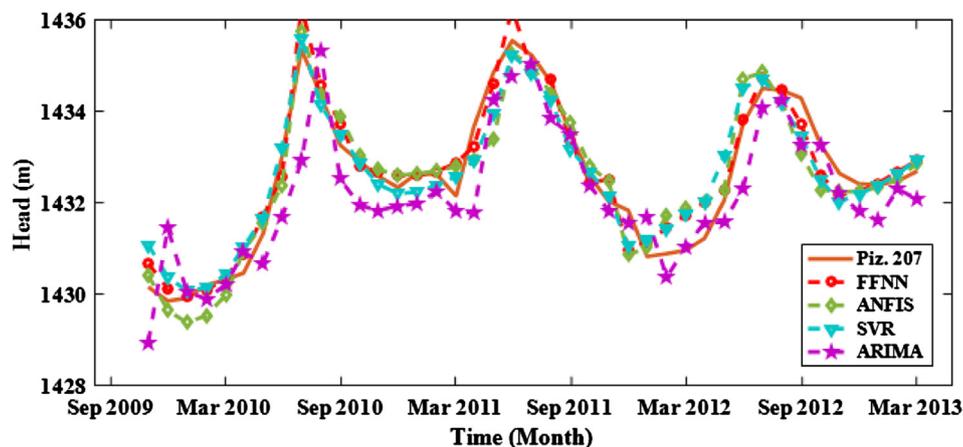


Fig. 5. Sole models' results without any data processing for testing phase for piezometer 207 using scenario 3.

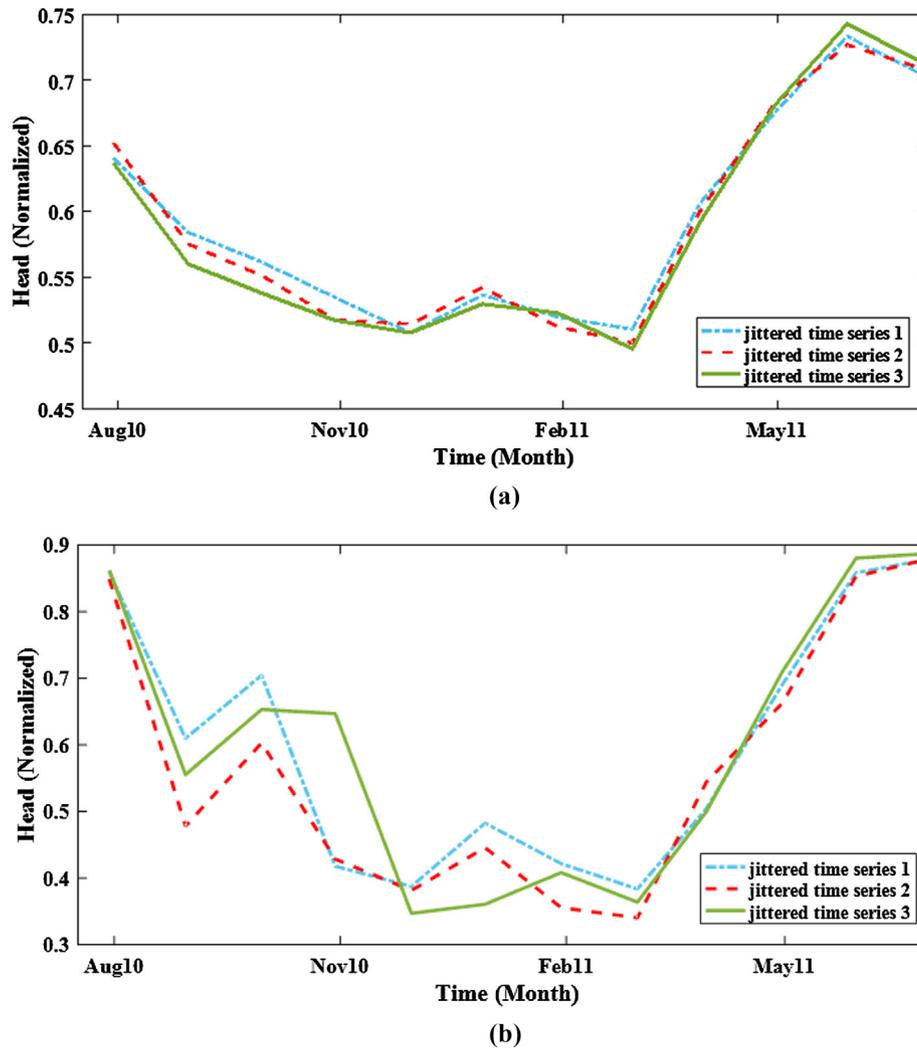


Fig. 6. 3 samples of generated jittered time series for (a) piezometer 207 with noise standard deviation of 0.01, (b) piezometer 217 with noise standard deviation of 0.1.

Table 3
Sole FFNN and SVR models' results using jittered data for scenario 1.

Model	Piezometer #	Standard deviation	R^2		RMSE (normalized)		
			Training	Testing	Training	Testing	
FFNN	207	0.005	0.9123	0.7186	0.0597	0.0356	
		0.01	0.9125	0.7794	0.0596	0.0336	
		0.05	0.9144	0.8169	0.0588	0.0327	
		0.1	0.8303	0.7000	0.0826	0.0414	
	217	0.005	0.8166	0.7221	0.1129	0.1117	
		0.01	0.8763	0.7528	0.0927	0.1135	
		0.05	0.8371	0.8235	0.1064	0.1058	
		0.1	0.8001	0.6347	0.0850	0.1280	
	SVR	207	0.005	0.9110	0.7104	0.0646	0.0370
			0.01	0.9180	0.7721	0.0617	0.0367
			0.05	0.9264	0.7960	0.0629	0.0356
			0.1	0.8173	0.6654	0.0661	0.0410
217		0.005	0.8104	0.6526	0.0953	0.1237	
		0.01	0.8739	0.6630	0.0940	0.1219	
		0.05	0.8332	0.7015	0.0824	0.1147	
		0.1	0.8040	0.5546	0.0901	0.1401	

AI modeling for piezometers 207 and 217 respectively by about 10%, 15% for training step and 15%, 20% for the testing step. In other words, by comparing the results of first step and results of modeling employing jittering, it's clear that this method could improve

the modeling performance. Furthermore, according to Tables 2–4, employing jittering in this study could reduce the overtraining issue in some cases such as FFNN models for piezometers # 207 and 217 via scenario 1 and SVR model for piezometer # 207 via

Table 4
Sole AI models' results using jittered data for scenarios 2 and 3 for standard deviation 0.05.

Scenario #	Piezometer #	Model	R^2		RMSE (normalized)	
			Training	Testing	Training	Testing
2	207	FFNN	0.9164	0.8354	0.0479	0.0396
		SVR	0.9364	0.8107	0.0362	0.0446
	217	FFNN	0.8638	0.7262	0.0501	0.0979
		SVR	0.8541	0.7078	0.0566	0.1147
3	207	FFNN	0.9205	0.8400	0.0320	0.0392
		SVR	0.9004	0.8307	0.0427	0.0423
	217	FFNN	0.9080	0.7400	0.0510	0.1121
		SVR	0.9050	0.6922	0.0585	0.1175

scenario 3 and in some other cases it could not decrease overtraining problem like SVR model for piezometer # 217 via scenario 1 and FFNN model for piezometer # 207 via scenario 3. By comparing the outcomes of two piezometers, it is clear that employing the jittering pre-processing method had a higher impact on piezometer 217 and improved its performance more than piezometer 207. In other words, pre-processing method improved the performance of the piezometer with lower performance more than other and the reason of this can be due to following statements. As mentioned in Section 3.1 piezometer # 217 situated in topmost elevation of the core and mid-section so it is not affected by the variation of the dam's reservoir water head as much as piezometer #207. So modeling piezometric head of piezometer #217 using dam's reservoir water head and other piezometers' water heads cannot discover much patterns of data and is less efficient than piezometer # 207. But, since that jittering pre-processing method generates more artificial data patterns, so more uncovered data patterns of piezometer #217 can be handled. Hence pre-process method of jittering is more efficient, when the quality of original dataset is not suitable due to some problems. Employing jittering data pre-processing method improved the performance of the models, because models are trained by various patterns during the training step, so the accuracy of modeling increases in the testing step.

3.3. Results of models linked to ensemble data post-processing

In the third step, outputs of single models that obtained in the first step were combined employing three ensembling post-processing methods. In this step just the training dataset was employed to determine the parameters of both weighted linear

and nonlinear averaging techniques. In nonlinear averaging technique similar to sole FFNN, applying tan-sig transfer function for both intermediate and target layers, the model was calibrated utilizing scaled conjugate gradient method of the BP approach. Also trial-error procedure employed to find the optimum architecture and training iteration epoch.

The outputs of the ensemble techniques for piezometers 207 and 217 via scenarios 1, 2 and 3 have been presented in Table 5. It should be noticed that numbers of $\alpha, \beta, \gamma, \lambda$ in the weighted ensemble refer to the constants applied to the outputs of the FFNN (α), ANFIS (β), SVR (γ) and ARIMA (λ) approaches. For example, Fig. 7 depicts the recorded and obtained water level time series of piezometers # 207 and 217 obtained by the neural ensemble method for the testing phase. Furthermore, scatter plot of testing phase for scenarios 1, 2 and 3 for piezometer 207 and piezometer 217 have been respectively illustrated by Figs. 8 and 9.

The obtained outcomes of ensemble techniques evidenced that pretty all of the ensemble models provide more accurate results regarding to the sole models. Therefore, simple, weighted and nonlinear averaging techniques improved the performance of AI models in piezometer 207 and piezometer 217 respectively by about 7%, 7%, 10% and 15%, 15%, 22% for training step and 16%, 16%, 18% and 17%, 17%, 24% in the testing step. It is clear that increment of R^2 s in the training step was not considerable almost for all methods, but the performance betterments are significant in the testing step that is the essential objective of this study. As discoursed before, some of the methods resulted over and some others resulted lower estimates because each method has its own perks and drawbacks. However, ensemble models, due to utilizing each method's unique potency, predict the issue more accurate than sole models. It should be noticed that since the outputs of the sole

Table 5
Ensemble methods' results for scenarios 1, 2 and 3.

Scenario #	Piezometer #	Ensemble method	Model structure	R^2		RMSE (normalized)	
				Training	Testing	Training	Testing
1	207	Simple averaging	-	0.9009	0.7704	0.0550	0.0348
		Weighted averaging	0.2594, 0.2676, 0.2553, 0.2176	0.9050	0.7810	0.0549	0.0348
		Neural averaging	4-7-1	0.9102	0.8000	0.0543	0.0348
	217	Simple averaging	-	0.8108	0.7157	0.0907	0.1130
		Weighted averaging	0.2578, 0.2676, 0.2463, 0.2283	0.8252	0.7146	0.0905	0.1132
		Neural averaging	4-5-1	0.8809	0.7565	0.1052	0.1045
2	207	Simple averaging	-	0.9108	0.7810	0.0390	0.0366
		Weighted averaging	0.2510, 0.2541, 0.2527, 0.2422	0.9108	0.7806	0.0390	0.0366
		Neural averaging	4-6-1	0.9387	0.8220	0.0333	0.0296
	217	Simple averaging	-	0.8504	0.7284	0.0587	0.1099
		Weighted averaging	0.2363, 0.2639, 0.2594, 0.2404	0.8515	0.7310	0.0581	0.1104
		Neural averaging	4-1-1	0.9081	0.7851	0.0882	0.0982
3	207	Simple averaging	-	0.911	0.8360	0.0374	0.0271
		Weighted averaging	0.2628, 0.2612, 0.2586, 0.2174	0.9150	0.8366	0.0374	0.0270
		Neural averaging	4-2-1	0.9401	0.8430	0.0364	0.0240
	217	Simple averaging	-	0.8850	0.7884	0.0556	0.0975
		Weighted averaging	0.2602, 0.2534, 0.2659, 0.2205	0.8870	0.7878	0.0557	0.0976
		Neural averaging	4-7-1	0.9201	0.8087	0.0598	0.0927

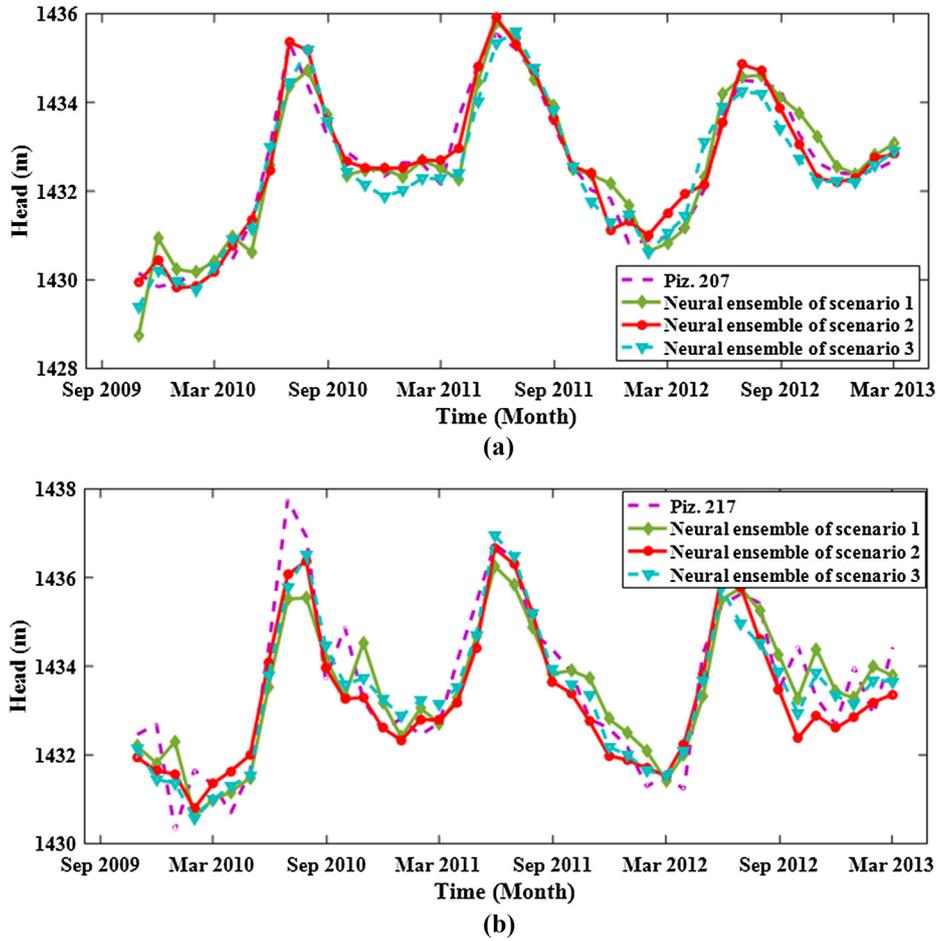


Fig. 7. Neural ensemble method's results for testing step (a) piezometer 207, (b) piezometer 217.

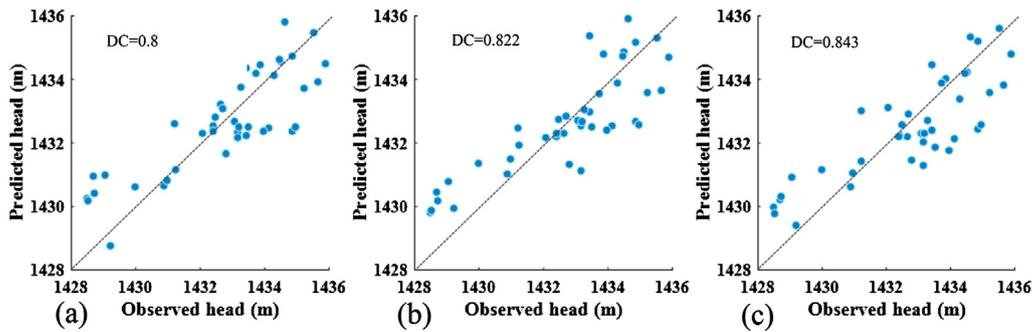


Fig. 8. Scatter plots of neural ensemble methods of piezometer 207 for scenarios (a) one, (b) two, (c) three for the testing phase.

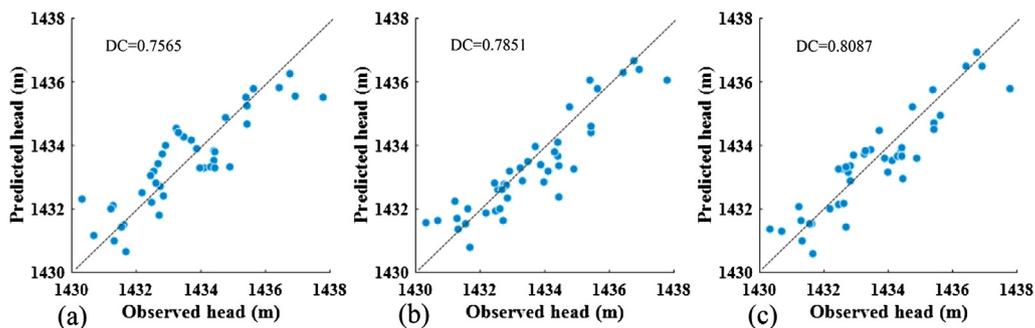


Fig. 9. Scatter plots of neural ensemble methods of piezometer 217 for scenarios (a) one, (b) two, (c) three for the testing phase.

methods are close together (see Table 2), and because the performance of simple and weighted averaging ensemble techniques are in the same directs with the sole methods, outcomes of simple and weighted ensemble techniques are quite same.

The performance of the nonlinear averaging method was higher than two linear averaging techniques. In nonlinear averaging because of using FFNN, modeling of the nonlinear pattern of the problem can be more precise than linear averaging models. On the other hand, because the outputs of linear averaging methods are in the same directs with the sole methods, if the performance of an AI model is weak, the achieved outcomes of averaging will be weak as well; in such condition the nonlinear neural averaging using will be much beneficial.

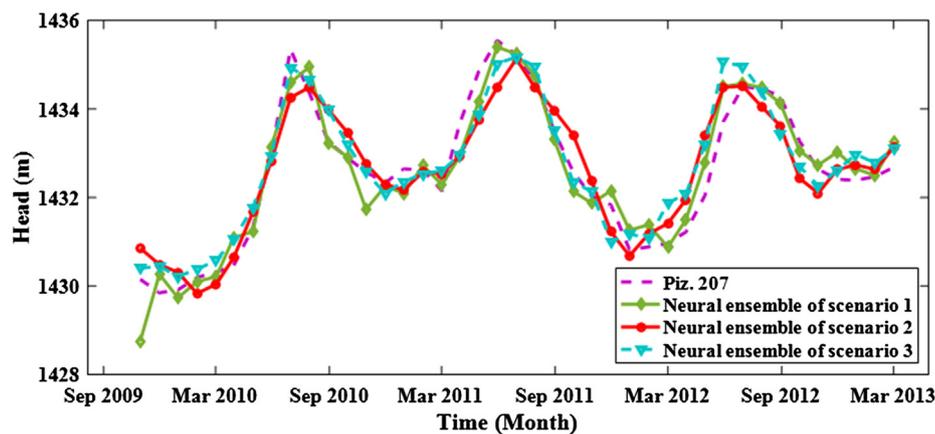
3.4. Results of models linked to both pre-post processing methods

Based on the previous step, the performance of the neural ensemble method was more accurate than two linear ensemble models. Therefore in the fourth step, the outputs of AI models using jittered data generated by the best standard deviation ($\sigma = 0.05$) were combined using neural ensemble method. Same as the third step, employing tan-sig transfer function for both intermediate and target layers, the neural ensemble model was calibrated employing scaled conjugate gradient method of the BP approach to find the optimum architecture and training iteration epoch and only the training dataset was employed to determine the parameters of the neural ensemble method.

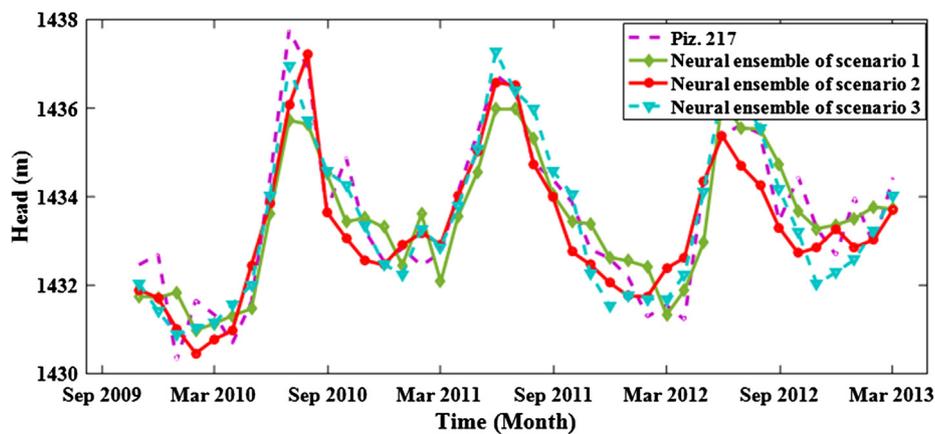
Table 6

Results of ensemble methods linked to jittering data pre-processing for scenarios 1, 2, 3.

Scenario #	Piezometer #	Ensemble method	Model structure	R^2		RMSE (normalized)	
				Training	Testing	Training	Testing
1	207	Neural averaging	2-7-1	0.9397	0.9153	0.0564	0.0308
	217	Neural averaging	2-6-1	0.8553	0.800	0.1003	0.0997
2	207	Neural averaging	2-5-1	0.9462	0.8661	0.0422	0.0388
	217	Neural averaging	2-6-1	0.8614	0.8038	0.0518	0.0938
3	207	Neural averaging	2-9-1	0.939	0.8811	0.0290	0.0365
	217	Neural averaging	2-10-1	0.9117	0.8100	0.0867	0.0854



(a)



(b)

Fig. 10. The results of neural ensemble method linked to jittering data pre-processing for testing step (a) piezometer 207, (b) piezometer 217.

The obtained results for the piezometers 207 and 217 via scenarios 1, 2 and 3 are presented in Table 6. Fig. 10 depicts the observed and estimated water level time series by the neural ensemble method for the piezometer 207 and piezometer 217 in the testing phase. Moreover, the scatter plots of the testing phase for all scenarios and piezometer 207 and piezometer 217 have been respectively presented in Figs. 11 and 12. Besides, the best results (R^2) of testing phase for FFNN model for different stages of modeling of piezometers # 207 and # 217 for scenarios 1,2,3 are tabulated in Table 7 for better comparison of modeling performance in 4 different stages of modeling.

According to Table 6, at this step of modeling, in almost all cases, the neural ensemble model increases the accuracy and performance of the modeling. In addition, employing both jittering and neural ensemble techniques could reduce overtraining problem slightly in all cases except the neural averaging method for piezometer # 217 via scenario 3 (see Tables 2 and 6).

Overall, employing both pre-processing (jittering) and post-processing (neural ensemble) techniques improved the performance of AI modeling for piezometers # 207 and 217 by 18% for training step and 32% for the testing step. It should be noted that using the jittering data pre-processing and ensemble post-processing methods simultaneously improved the modeling per-

formance more than using of each pre-processing or post-processing alone.

4. Conclusions

This paper deals with the performance of data pre and post processing techniques for seepage modeling. The available historic data of piezometric heads from Sattarkhan earthen dam, situated in northwest of Iran were employed to develop AI models. For this purpose, as the first step of modeling, the original dataset without any processing was employed to create AI models. In the second step of modeling, jittering pre-processing method was used. In this way, the noise time series with mean of zero and different standard deviation values were produced and added to the observed data. In the last stage, to improve the performance of modeling, the ensemble post-processing method was employed using linear simple and weighted averaging as well as nonlinear neural ensemble approaches to integrate outputs from single methods of first step (outputs of single models without any data pre-processing). Finally, in the fourth step of modeling, the obtained outputs of second step (outputs of single models linked to data pre-processing) were combined using ensemble post-processing method.

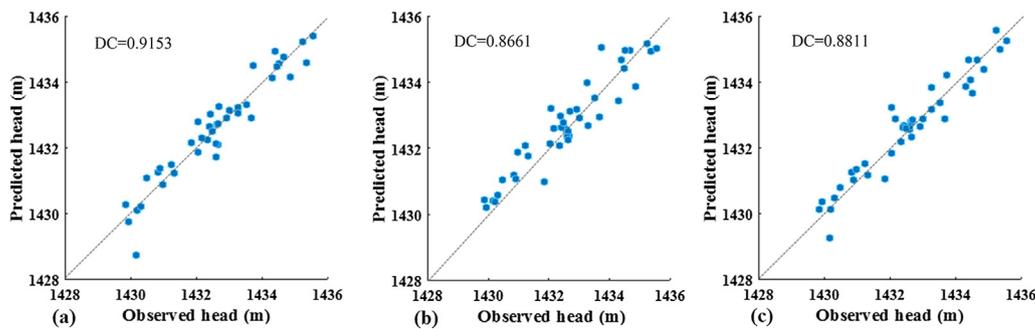


Fig. 11. Scatter plots of neural ensemble methods linked to jittering data pre-processing of piezometer 207 for scenarios (a) one, (b) two, (c) three for the testing phase.

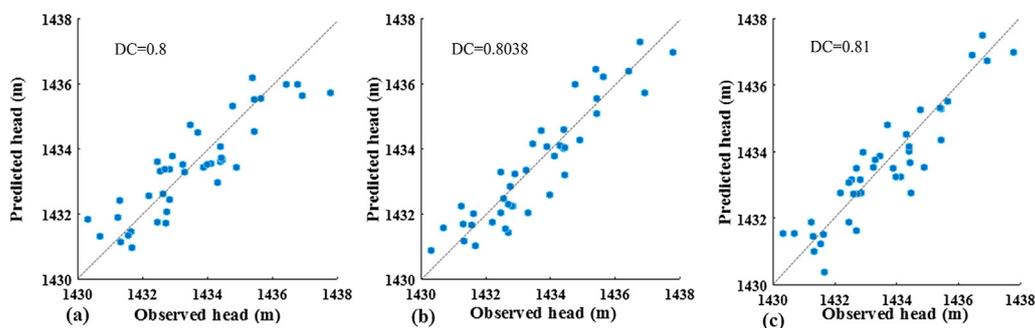


Fig. 12. Scatter plots of neural ensemble methods linked to jittering data pre-processing of piezometer 217 for scenarios (a) one, (b) two, (c) three for the testing phase.

Table 7

The best results (R^2) of testing phase for FFNN model for different stages of modeling of piezometers # 207 and # 217 for scenarios 1, 2, 3.

Scenario #	Piezometer #	Without pre-post processing	With pre-processing	With post-processing	With pre-post processing
1	207	0.7166	0.8169	0.8000	0.9153
	217	0.6865	0.8235	0.7565	0.800
2	207	0.7645	0.8354	0.8220	0.8661
	217	0.6906	0.7262	0.7851	0.8038
3	207	0.7800	0.8400	0.8430	0.8811
	217	0.7100	0.7400	0.8087	0.8100

The comparison of proposed scenarios indicated that using synchonic data with outputs can enhance modeling performance by 10%. In addition, it was examined and confirmed that if one of piezometers gets out of service, other piezometers could be utilized in simulating via scenarios 2 or 3.

Comparing the obtained results by employing jittering based pre-process method with regard to results of the modeling by original time series shows that the proposed method is efficient because of identifying the intermediate patterns in the data. Jittering could enhance the AI modeling performance for the testing phase by about 20% compared to the modeling with unprocessed data.

Ensemble models provided more accurate estimations than the sole methods and model ensembling enhanced the predicting performance by 25%. Expectedly, the neural ensemble method was found to be more potent and effective technique of model ensembling and this technique could enhance the performance of the AI based modeling in the testing stage by 25%.

The best outcomes were achieved by employing both of the pre-processing and post-processing methods together, so that the used pre-post processing methods improved the performance of AI modeling by about 30% in the testing phase.

Generally, the results of this paper proved the performance and effectiveness of jittering and ensembling data processing.

Two methods of pre-post processing examined in this research, but other methods and techniques can be applied to overcome the modeling deficiency such as Wavelet-Entropy and Self-Organizing Map (SOM) pre-processing methods [41–43]. Furthermore, it is suggested to utilize Genetic Algorithm (GA) to select the main inputs of AI models in future studies [44,45].

References

- [1] V. Nourani, M.H. Aminfar, M.T. Alami, E. Sharghi, V.P. Singh, Unsteady 2-D seepage simulation using physical analog, case of Sattarkhan embankment dam, *J. Hydrol.* 519 (2014) 177–189.
- [2] M.A. Shahin, M.B. Jaksa, H.R. Maier, Artificial neural network applications in geotechnical engineering, *Austr. Geomech.* 36 (1) (2001) 49–62.
- [3] ASCE Task Committee on Application of Artificial Neural Networks in Hydrology, Artificial neural networks in hydrology. 2: hydrology applications, *J. Hydrol. Eng.* 5 (2) (2000) 124–137.
- [4] G. Tayfur, D. Swiatek, A. Wita, V.P. Singh, Case study: finite element method and artificial neural network models for flow through Jeziorsko earthfill dam in Poland, *J. Hydraul. Eng.* 131 (6) (2005) 431–440.
- [5] V. Nourani, A. Babakhani, Integration of artificial neural networks with radial basis function interpolation in earthfill dam seepage modeling, *J. Comput. Civil Eng.* 27 (2) (2012) 183–195.
- [6] V. Nourani, E. Sharghi, M.H. Aminfar, Integrated ANN model for earthfill dams seepage analysis: Sattarkhan Dam in Iran, *Artif. Intell. Res.* 1 (2) (2012) 22–37.
- [7] A. Novaković, V. Ranković, N. Grujović, D. Divac, N. Milivojević, Development of neuro-fuzzy model for dam seepage analysis, *Ann. Faculty Eng. Hunedoara* 12 (2) (2014) 133–136.
- [8] L. Yongbiao, Prediction methods to determine stability of dam if there is piping, *IERI Procedia* 1 (2012) 131–137.
- [9] G.P. Zhang, A neural network ensemble method with jittered training data for time series forecasting, *Inf. Sci.* 177 (2007) 5329–5346.
- [10] R. Reed, R.J. Marks, S. Oh, Similarities of error regularization, sigmoid gain scaling, target smoothing, and training with jitter, *IEEE Trans. Neural Netw.* 6 (3) (1995) 529–538.
- [11] V. Nourani, A. Partoviyani, Hybrid denoising-jittering data pre-processing approach to enhance multi-step-ahead rainfall-runoff modeling, *Stochastic Environ. Res. Risk Assess.* (2017) 1–18.
- [12] K. Matsuoka, Noise injection into inputs in back-propagation learning, *IEEE Trans. Syst. Man Cybern.* 22 (3) (1992) 436–440.
- [13] S. Singh, Noise impact on time-series forecasting using an intelligent pattern matching technique, *Pattern Recognit.* 32 (8) (1999) 1389–1398.
- [14] G.P. Zhang, Time series forecasting using a hybrid ARIMA and neural network model, *Neurocomputing* 50 (2003) 159–175.
- [15] A.Y. Shamseldin, K.M. O'Connor, G.C. Liang, Methods for combining the outputs of different rainfall-runoff models, *J. Hydrol.* 197 (1997) 203–229.
- [16] A. Krogh, J. Vedelsby, Neural network ensembles, cross validation, and active learning, *Adv. Neural Inf. Process. Syst.* 7 (1995) 231–238.
- [17] J.M. Bates, C.W.J. Granger, The combination of forecasts, *Oper. Res. Quart.* 20 (1969) 451–468.
- [18] R.T. Clemen, Combining forecasts: a review and annotated bibliography, *Int. J. Forecast.* 5 (1989) 559–583.
- [19] E. Sharghi, V. Nourani, N. Behfar, Earthfill dam seepage analysis using ensemble artificial intelligence based modeling, *J. Hydroinf.* 20 (5) (2018) 1071–1084.
- [20] C.E. Shannon, A mathematical theory of communications I and II, *Bell Syst. Tech. J.* 27 (1948) 379–443.
- [21] H.H. Yang, S.V. Vuuren, S. Sharma, H. Hermansky, Relevance of time-frequency features for phonetic and speaker-channel classification, *Speech Commun.* 31 (2000) 35–50.
- [22] B. Najafi, S. Faizollahzadeh Ardabili, S. Shamshirband, K.W. Chau, T. Rabczuk, Application of ANNs, ANFIS and RSM to estimating and optimizing the parameters that affect the yield and cost of biodiesel production, *Eng. Appl. Comput. Fluid Mech.* 12 (1) (2018) 611–624.
- [23] V. Nourani, An Emotional ANN (EANN) approach to modeling rainfall-runoff process, *J. Hydrol.* 544 (2017) 267–277.
- [24] G. Tayfur, *Soft Computing Methods in Water Resources Engineering: Artificial Neural Networks, Fuzzy Logic, and Genetic Algorithm*, WIT Press, Southampton, UK, 2012.
- [25] V. Nourani, T.R. Khanghah, A.H. Baghanam, Application of entropy concept for input selection of wavelet-ann based rainfall-runoff, *Modeling, J. Environ. Inform.* 26 (1) (2015).
- [26] J.S.R. Jang, C.T. Sun, E. Mizutani, *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, Prentice-Hall, New Jersey, 1997.
- [27] J.S.R. Jang, ANFIS: Adaptive-network-based fuzzy inference system, *IEEE Trans. Syst. Man Cybern.* 23 (3) (1993) 665–685.
- [28] C. Cortes, V. Vapnik, Support-vector networks, *Mach. Learn.* 20 (1995) 273–297.
- [29] W.C. Wang, D.M. Xu, K.W. Chau, S. Chen, Improved annual rainfall-runoff forecasting using PSO-SVM model based on EEMD, *J. Hydroinf.* 15 (2013) 1377–1390.
- [30] N.S. Raghavendra, P.C. Deka, Support vector machine applications in the field of hydrology: a review, *Appl. Soft Comp.* 19 (2014) 372–386.
- [31] J.D. Salas, J.W. Delleur, V. Yevjevich, W.L. Lane, *Applied Modeling of Hydrological Time Series*, Water Resources Publications, Denver, 1990.
- [32] A.H. Bowker, G.J. Lieberman, *Engineering Statistics*, Prentice-Hall, New Jersey, 1972.
- [33] S. Makridakis, R.L. Winkler, Average of forecasts: some empirical results, *Manage. Sci.* 29 (9) (1983) 987–996.
- [34] P. Kazienko, E. Lughofer, B. Trawiński, Hybrid and ensemble methods in machine learning, *J. Comput. Sci.* 19 (4) (2013) 457–461.
- [35] S. Haykin, *Neural Networks: A Comprehensive Foundation*, McMillan, New York, 1994.
- [36] M.F. Møller, A scaled conjugate gradient algorithm for fast supervised learning, *Neural Netw.* 6 (4) (1993) 525–533.
- [37] R. Prasad, A. Pandey, K.P. Singh, V.P. Singh, R.K. Mishra, D. Singh, Retrieval of spinach crop parameters by microwave remote sensing with back propagation artificial neural networks: a comparison of different transfer functions, *Adv. Space Res.* 50 (3) (2012) 363–370.
- [38] J.E. Schmitz, R.J. Zemp, M.J. Mendes, Artificial neural networks for the solution of the phase stability problem, *Fluid Phase Equilib.* 245 (1) (2006) 83–87.
- [39] R. Noori, A.R. Karbassi, A. Moghaddamnia, Assessment of input variables determination on the SVM model performance using PCA, gamma test and forward selection techniques for monthly stream flow prediction, *J. Hydrol.* 401 (2011) 177–189.
- [40] R. Luus, T.H. Jaakola, Optimization by direct search and systematic reduction of the size of search region, *AIChE J.* 19 (4) (1973) 760–766.
- [41] V. Nourani, M. Komasi, A. Mano, A multivariate ANN-wavelet approach for rainfall-runoff modeling, *Water Resour. Manage.* 23 (14) (2009) 2877–2894.
- [42] V. Nourani, M.T. Alami, F.D. Vousoughi, Wavelet-entropy data pre-processing approach for ANN-based groundwater level modeling, *J. Hydrol.* 524 (2015) 255–269.
- [43] V. Nourani, M. Parhizkar, Conjunction of SOM-based feature extraction method and hybrid wavelet-ANN approach for rainfall-runoff modeling, *J. Hydroinf.* 15 (3) (2013) 829–848.
- [44] C.L. Chang, S.L. Lo, S.L. Yu, Applying fuzzy theory and genetic algorithm to interpolate precipitation, *J. Hydrol.* 314 (1–4) (2005) 92–104, <https://doi.org/10.1016/j.jhydrol.2005.03.034>.
- [45] Z. Şen, A. Öztopal, Genetic algorithms for the classification and prediction of precipitation occurrence, *Hydrol. Sci. J.* 46 (2) (2001) 255–267.