

Estimation of Dam Failure Peak Outflow using Neural Network Approach

Ahmad Tahershamsi; Farhad Hooshyaripor^{*†}; Raza Sheikholeslami
Faculty of Civil and Environment Engineering, Amirkabir University of Technology, Tehran,
Iran

Abstract

This article presents an artificial neural network (ANN) for prediction of peak outflow from breached embankment dams based on considering height and volume of water behind the dam at the time of failure. Two different algorithms are used for training the ANN. They are Imperialist Competitive Algorithm (ICA) as a new evolutionary algorithm and Levenberg-Marquardt (LM) algorithm. The comparison of results between the proposed method and those conventional approaches which are based on regression analysis shows a better performance of the ANN based models. To evaluate the uncertainty of the two training algorithms, a Monte-Carlo simulation is used to sample 1000 sets from the database of historical dam failures for different sets of training and test in the ANN model. Three statistical measures (i.e. 95PPU, d -factor, and DDR) are used to compare the uncertainty analysis. The obtained results indicate a better performance of ICA compared to LM algorithm.

Key words: Dam failure; Neural network; Levenberg-Marquardt; Imperialist competitive algorithm; Uncertainty

1. Introduction

Prediction of peak outflow (Q_p) from breached embankment dams is an important procedure for emergency action plan preparation and risk assessment [1]. Therefore, developing a simple and friendly approach to deal with such problems has been the focus of a multitude of contributions to the literature in the last decades [2,3,4,5,6,7]. In spite of analyzing a wide range of reliable experimental and case studies, the problem of breach outflow prediction is still debatable. It seems that this is, partly, due to the complexity of the phenomenon and low accuracy of data driven from historical dam failures [8], and all the more reason because of the limitation of the analytical tool commonly used by most of the earlier investigators, namely traditional statistical regression. In the regression based method, historical datasets have usually been applied to develop relations between Q_p and height and/or volume of water behind the dam [2,3,4,5,6,7]. The uncertainty of these relations was specifically quantified by Wahl (2004). He declared that the uncertainties of peak outflow predictions were large for all considered relations, hence predictions of peak outflow had uncertainty of about ± 0.5 to ± 1 order of magnitude, except the Froehlich peak flow equation (Froehlich 1995), which had an uncertainty of about $\pm 1/3$ order of magnitude.

Froehlich (2008) evaluated parameters needed in the relations of breach formation that consider the breach to form in the shape of a trapezoid. Taking into account the uncertain

* Corresponding author: F. Hooshyaripor, Department of Civil and Environment Engineering, Amirkabir University of Technology, Tehran, Iran

† E-mail: Hooshyaripor@yahoo.com

nature of the breach model, expressions for expected values of breach parameters and their variances were used in a stochastic model of dam breach flooding that evaluates the variability of peak stages and peak outflow rates using a Monte-Carlo simulation approach. As an alternative to the traditional statistical approach, some researchers used artificial neural network (ANN) (Hooshyari and Tahershamsi, 2012; Nourani et al., 2012) due to its data-driven nature, model-free form of predictions, and tolerance to data errors [9,10,11,12,13,14]. The investigations have demonstrated the higher performance of the new approach compared to the other traditional methods. The objective of this study is to reanalyze the available datasets using ANN considering artificial intelligence approach in the network-training phase as well as evaluation of the model's uncertainty. The rest of the paper is organized as follows: Section 2 contains a description of the datasets; section 3 is a brief description of the methodology that was employed for data analysis; and sections 4 and 5 deal with summary of the results and conclusions, respectively.

2. Database collection

The proposed methodology uses a database containing 93 actual embankment dam failures collected by Hooshyari and Tahershamsi (2012) from numerous sources [1,5,7,15,16,17]. These datasets have previously been applied to develop equations for estimation of breach parameters and especially peak outflow. For example, Froehlich in 1995 and 2008 assembled data from 22 and 74 embankment dam failures respectively [5, 28]. In 1995 he developed a new empirical relation based on multiple regression analysis for rapid estimation of peak outflow from a breached embankment dam as

$$Q_p = 0.607 H_w^{1.24} V_w^{0.295} \quad (1)$$

where Q_p = predicted peak outflow (m³/s); H_w = height of water in the reservoir at the time of failure (m), and V_w = reservoir volume at the time of failure (m³).

3. Artificial Neural network (ANN) model

3.1. Architecture of ANN

Neural Networks are typically thought of as black boxes trained to a specific task on a large number of data samples. A typical network would consist of three layers of neurons namely, input, hidden, and output. Many theoretical and experimental works have shown that single hidden layer is sufficient for ANNs to approximate any complex nonlinear function [19,20].

3.2. Training of ANN

To achieve an efficient model, the ANN needs to be trained with sufficient data to minimize the error between the actual and network response. There are different training algorithms in the literature. Here, the algorithms used for ANN training are LM and ICA which are outlined below.

LM method developed by Levenberg [22] and Marquardt [23], provides a numerical solution to the problem of minimizing a nonlinear function over a space of parameters of the function. These minimization problems arise, particularly, in least squares curve fitting and nonlinear programming. LM interpolates between the Gauss-Newton algorithm and the method of gradient descent. This method is a very popular curve-fitting algorithm used in many software applications for solving generic curve-fitting problems; however, it finds only a local minimum, not a global minimum. On the other hand, ICA is a new progressive meta-heuristic algorithm for optimization. This algorithm starts with an initial population, namely country. In ICA, some of the best countries in the population are selected to be the

imperialists and the rest form the colonies of these imperialists. When the competition starts, the imperialists attempt to achieve more colonies and the colonies start to move toward their imperialists. Thus, during the competition, the powerful imperialists will survive and the weak ones will collapse. The algorithm would terminate when only one imperialist is left. Moving colonies toward imperialists are continued and imperialistic competition and implementations are performed during the search process. More details about this algorithm are presented in Atashpaz-Gargari and Lucas [24], Atashpaz-Gargari et al. [25], Kaveh and Talatahari [26], Tahershamsi and Sheikholeslami [27], and Hooshyaripor and Tahershamsi (2012).

3.3. Uncertainty analysis of ANN

To compare the performance of the two algorithms of ANN models, an uncertainty analysis is carried out by using a Monte-Carlo simulation. To do so, input parameters are first characterized using a probability density function (PDF). Then a single set of input data for input layer of the ANN model is randomly selected from the PDFs of the parameters. The ANN model is then trained and tested with the selected set of input data and consequently the generated output for peak outflow is saved as the result of the ANN model. This procedure is repeated frequently for many times. Theoretically, the multiple runs must carry out as long as the results of a new run do not affect the probability distribution of the output variable. Here it is assumed that 1000 times are sufficient for multiple runs of the randomly generated ANN model. Finally, the resulting statistical performances (e.g. mean, median, variance, and percentiles) are collected, tabulated and their distributions are plotted. In the present work, the above method is adapted by using random samples instead of random data which is relatively similar to Bootstrap pairs re-sampling method (Tibshirani, 1995). In this method, the database is needed to be randomly re-sampled without replacement several times, maintaining the ratio between the training and validation sets.

3.4. Evaluation of the models

In order to compare the performance of the ANN models and regression techniques, a number of error quantification measures are used here as follows: (1) coefficient of determination (R^2) representing the association degree between predicted and observed values; (2) average error (AE) as a parameter commonly used in engineering applications equal to algebraic difference between predicted and observed values; (3) mean absolute error (MAE); and (4) root mean square error (RMSE).

To quantify the uncertainty of the two algorithms of ANN training, three techniques are used here: (1) the 95 percent prediction uncertainties (95PPU) and (2) d -factor; (3) developed discrepancy ratio (DDR). These techniques are outlined in the following.

95PPU is calculated as:

$$\text{Bracketed by 95PPU} = \frac{1}{n} \text{count} (X | X_L \leq X \leq X_U) \times 100 \quad (2)$$

where n is the number of observed data points, X is the predicted variable and X_L and X_U are, respectively, 2.5th and 97.5th percentiles of the cumulative distribution of every simulated point. Thus, the goodness of fit is assessed by the uncertainty measures calculated from percentage of measured data bracketed by the 95PPU band.

In the second method, the distance between upper and lower bound is used for uncertainty evaluation. Therefore, the average distance \bar{d}_X between the upper and the lower 95PPU (or the degree of uncertainty) calculated as follow (Abbaspour et al., 2007):

$$\bar{d}_x = \frac{1}{n} \sum_{i=1}^n (X_U - X_L) \quad (3)$$

The best result is that \bar{d}_x is equal to zero when 100% of the measurements are bracketed by the 95PPU. A dimensionless measure of the d -factor can be expressed as (Abbaspour et al., 2007):

$$d - \text{factor} = \frac{\bar{d}_x}{\sigma_x} \quad (4)$$

where σ_x is the standard deviation of the measured variable X . The values less than 1 are desirable measures for d -factor (Abbaspour et al., 2007), and the best value theoretically is zero. Obviously, the less d -factor value, the less the uncertainty is expected.

The statistical measures defined by Eqs. 2 and 4 only show the average accuracy in a model operation and they do not give any information about the predicts distribution. To overcome this disadvantage, another proper measure, developed discrepancy ratio (DDR) which was introduced by Noori et al. (2010), was used to be better checked the model's robustness. DDR can be calculated as follow (Noori et al. 2010):

$$\text{DDR} = \left(\frac{\text{Predicted Value}}{\text{Observed Value}} \right) - 1 \quad (5)$$

Here, according to 1000 predictions for an embankment dam from randomly generated sets of input data, 1000 DDR values were calculated. For a better judgment and visualization, the Gaussian function of DDR values could be calculated and illustrated in a standard normal distribution format. Therefore, DDR values are standardized and then, normalized values of DDR (Q_{DDR}) are calculated using Gaussian function. Then, Q_{DDR} is plotted versus standardized DDR (Z_{DDR}) to illustrate prediction distribution. The same procedure is repeated for all 93 datasets and related 68% confidence intervals are calculated and compared. In the obtained figures, more tendencies to the centerline as well as larger value of the maximum Q_{DDR} assess as more accuracy and less uncertainty of a model.

4. Results and discussion

4.1. development of ANN models

To develop the ANN models, 85% of the total observations of embankment dam failure are assigned for ANN training and the rest 15% is kept for test which are used for the performance evaluation of the ANN model and the Froehlich relation. Note that these test data are not used for derivation of neither ANN model nor the empirical relations. According to Cybenko (1989), here a three-layer network is employed in which mean square error (MSE) is used as the performance function. In the first model, to check the over-fitting problem in the training step, stop training algorithm method [21] is utilized, while in the second model when the number of iterations reached a pre-defined value, the search process was stopped. Furthermore, the tangent-sigmoid and linear functions are chosen as the activation function respectively in the hidden and output layers. Table 1 shows a quantitative comparison between the models.

Table 1: Comparison of the models performance

| Training algorithm/ relation | Step | Statistical indices | | | |
|---------------------------------|-------|---------------------|--------|-------------------------|--------------------------|
| | | R ² | AE (%) | MAE (m ³ /s) | RMSE (m ³ /s) |
| LM | train | 0.90 | 621.3 | 1756 | 3276 |
| | test | 0.93 | 149.1 | 3449 | 7331 |
| ICA | train | 0.89 | 503.1 | 1763 | 3497 |
| | test | 0.96 | 100 | 2032 | 3839 |
| Froehlich | train | 0.54 | 417.8 | 2473 | 7974 |
| | test | 0.85 | 21.5 | 3938 | 8236 |

As it can be seen in the Table, although the peak-outflow prediction for Froehlich relation outperform the ANN models with respect to the average error (AE), the performance of the ANN models is better for the coefficient of determination (R²) and absolute deviation and RMSE indicating that the prediction made using the statistical technique may be viewed with skepticism. Overall, the performance of the ANN model with ICA training with respect to all statistical indices indicates a higher performance compared to other models.

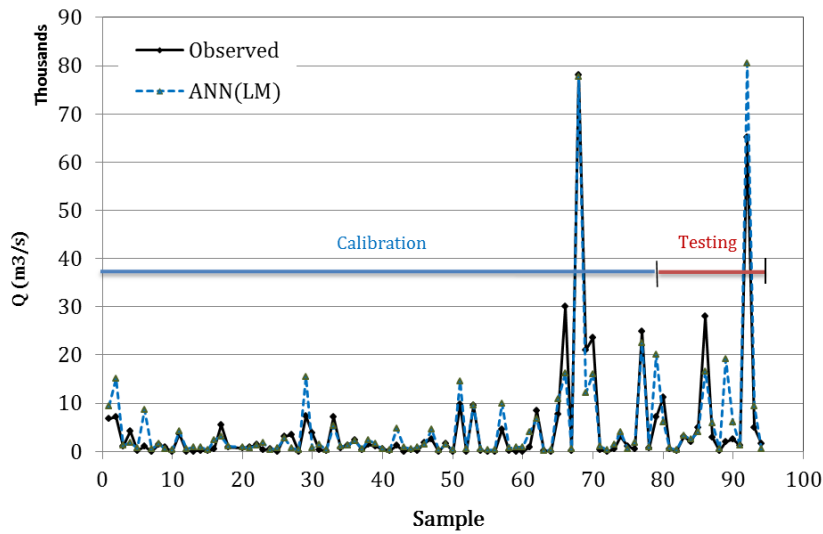


Fig. 1 shows LM results in both training and test steps. It is noticed that most of prediction values are approximately coincided with their corresponding observed values while there are

discrepancies among others which are marked in

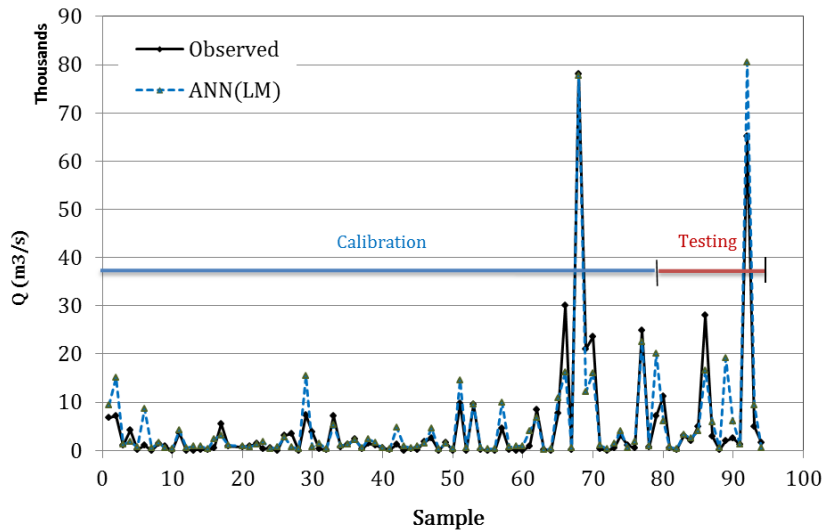


Fig. 1 with circles and squares for calibration and testing steps, alternatively. As it is illustrated, there are 10 points in calibration step (12.6% of calibration datasets) and 3 points in the testing step (21.4% of testing datasets) which are not predicted satisfactorily. Considering the percentage of the badly predicted points, one can clarify the reason of high error measures of LM in the testing step. In addition, most of the badly predicted points are related to extrem values (large dams). Accordingly, it can strongly be concluded that insufficient data and/or lack of historical data of large breached embankment dams could be the main reason of weak performance of ANN.

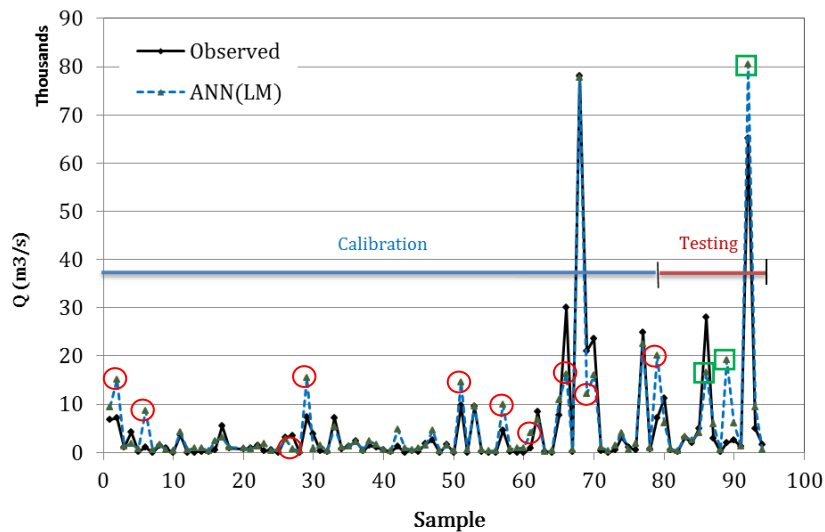


Fig. 1. Results of ANN model trained with LM algorithm

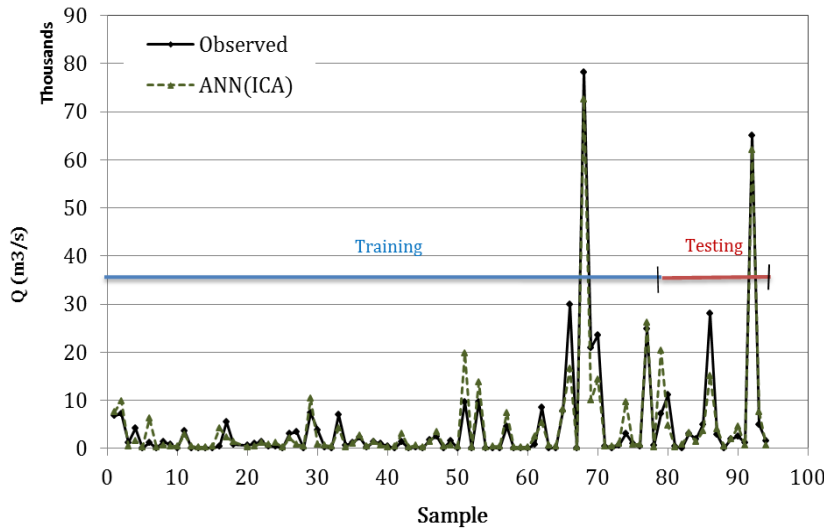


Fig. 2 illustrates ICA performance in training and testing steps. As it is illustrated, there are some imprecise predicted values in the training (marked with circles) and testing (marked with square) steps. They consist of 8 points in the training step (10.1% of training datasets) and one point in the testing step (7.1% of testing datasets). A comparison of the percentage of imprecisions in ANN models shows that ICA has more efficiency than LM. This is due to the fact that LM algorithm finds local minimum while ICA, as an evolutionary algorithm, does not trap in local optimum and will be stopped when a certain number of iterations is achieved.

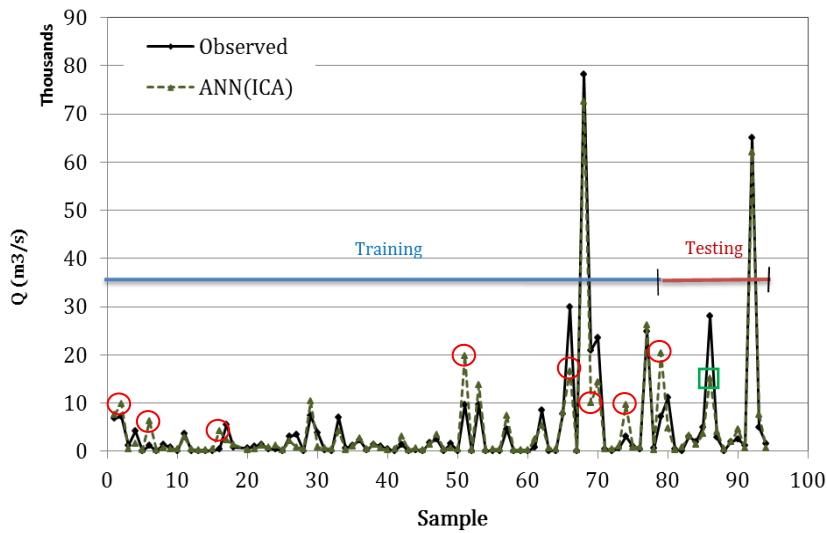


Fig. 2. Results of ANN model trained with ICA

The information on number of nodes, number of epochs required to achieve the error goal, and the CPU time taken in the case of each training scheme are presented in Table 2. A PC with characteristics of Pentium IV processor (CPU: Core i5, 2.53 GHz), was utilized in this study. It is remarkable that the information is the average of 50 iterations for each model.

Table 2. Network architecture.

| Algorithm | |
|-----------|-----|
| LM | ICA |

| | | | |
|----------------|--------------|-----|----|
| | Input | 2 | 2 |
| node | Hidden layer | 4 | 3 |
| | output | 1 | 1 |
| Epoch No. | | 215 | 70 |
| CPU time (sec) | | 3 | 55 |

As a matter of fact, it is noticeable that ICA, in a larger time scale, trains the network with fewer epochs compared to LM algorithm.

Fig. 3 presents a comparison between performance of ANN models and Froehlich (Table 2). Froehlich formula has a very poor performance for testing data ($R^2=0.54$) and relatively good performance for training data ($R^2=0.85$). On the other hand, high prediction error (RMSE=8236) implies that this formula might not be reliable in engineering application. Moreover, as it has been illustrated Fig. 3, the Froehlich formula underestimates the observed values, this is typically shown in Fig. 3a; the observed peak outflow of Banqiao dam failure is $78100 \text{ m}^3/\text{s}$ while Froehlich formula gives it as $16735 \text{ m}^3/\text{s}$ which is 78% less than that of observed value.

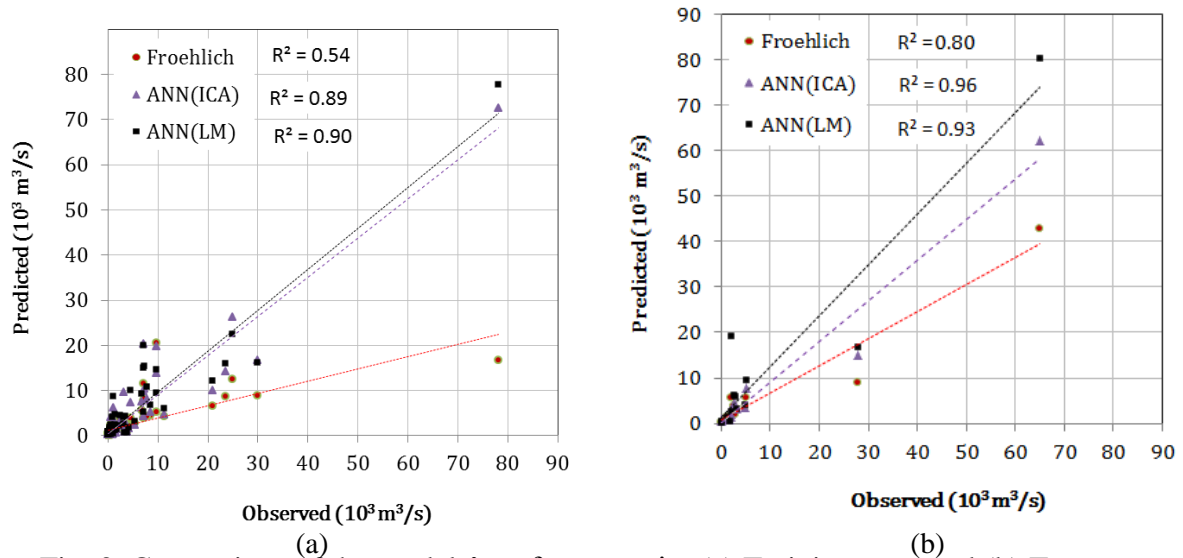


Fig. 3. Comparison of the models' performance in; (a) Training step; and (b) Test step

In the case of Liujiatai dam failure (Fig. 3b) the observed value is $28000 \text{ m}^3/\text{s}$, while the estimated value by Froehlich Formula is $9032 \text{ m}^3/\text{s}$ (about 67% less than the observed value). The main cause of this event is referred to the low impact of V_w on Q_p in Froehlich formula. Thus, Froehlich formula doesn't cover all ranges of effective parameters (V_w and H_w) and it could only be applicable to a limited range of the parameters variations.

4.2. Results of uncertainty analysis

As described earlier, the database was randomly sampled without replacement for 1000 times, assigning 90% for training and the remaining 10% for test. Hence, an uncertainty analysis of 1000 developed ANN models were carried out by the aforementioned indicators (i.e 95 PPU and d -factor) presented in Table 3. average values of R^2 , MAE, and RMSE are also provided in the table for the two ANN models. The result of LM algorithm outperforms those of ICA (R^2 for LM algorithm is 26% better and also the MAE and RMSE values of this algorithm are 15% and 24% respectively worse than those of ICA. The LM algorithm

shows to outperform the ICA algorithm with respect to uncertainty analysis of multiple runs although the best ANN model could be developed using ICA algorithm as shown previously.

Table 3. The results of uncertainty analysis of the two ANN models

| Training algorithm | $\overline{R^2}$ | \overline{MAE} (m ³ /s) | \overline{RMSE} (m ³ /s) | <i>d</i> -factor | 95PPU |
|--------------------|------------------|---|--|------------------|-------|
| LM | 0.85 | 2586 | 5118 | 0.53 | 68.8 |
| ICA | 0.67 | 3023 | 6716 | 0.75 | 66.7 |

Additionally, assessment of 95PPU and *d*-factor indices in Table 3 reflect relatively lower uncertainty of LM algorithm. *d*-factor of LM algorithm is 29% less than that of ICA, which demonstrates more limited non-dimensional uncertainty bounds of this algorithm in comparison with ICA.

As the amount of uncertainty indices are almost close together, another index, DDR, is applied to get a better comparison. So, DDR for all predictions were calculated, standardized, and normalized. Fig. 4 illustrates normal distributions fitted to the obtained results for 9 embankment dams. The obtained distributions for the other embankment dams were presented in appendix I. According to Noori et al. 2010 the more maximum Q_{DDR} the more bell-shaped distribution would happen and the less prediction error is expected accordingly. It can be seen that the distributions related to ICA are often higher than those of LM algorithm.

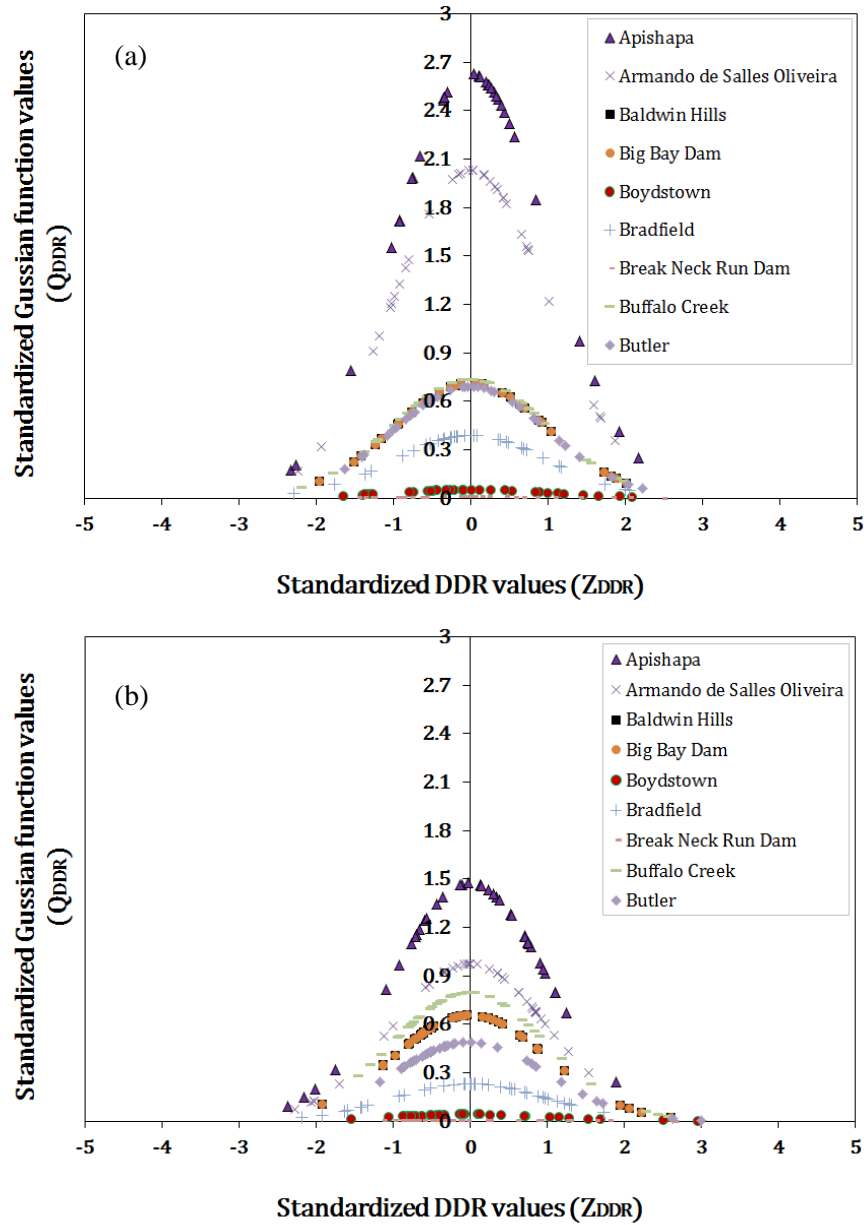


Fig. 4. normal distributions fitted to the standardized DDR values for the results of; (a) ICA; and (b) LM algorithm

Maximum Q_{DDR} values of 9 considered embankment dams are presented in Table 4. The maximum Q_{DDR} values related to the other embankment dams are available in appendix II. According to Table 4 the average of maximum Q_{DDR} related to ICA is 38% more than that of LM algorithm, which demonstrates less uncertainty of ICA in comparison with LM algorithm. Just in the case of Buffalo Creek dam the maximum Q_{DDR} value of LM algorithm is 8% higher than ICA and out of 84 other embankment dams only 8 cases are confronted with similar situation.

Table 4. Maximum Q_{DDR} value of the evaluated dams

| Dam Name | Location | LM | ICA |
|----------------------------|----------|------|------|
| Apishapa | USA | 1.47 | 2.63 |
| Armando de Salles Oliveira | Brazil | 0.97 | 2.03 |

| | | | |
|----------------|---------|------|------|
| Baldwin Hills | USA | 0.66 | 0.71 |
| Big Bay | USA | 2.05 | 2.44 |
| Boystown | Unknown | 0.04 | 0.05 |
| Bradfield | England | 0.23 | 0.39 |
| Break Neck Run | USA | 0.01 | 0.01 |
| Buffalo Creek | USA | 0.80 | 0.74 |
| Butler | USA | 0.49 | 0.69 |

The value of upper bound, lower bound, and average of the thousand predictions are illustrated in Fig. 5a and b. The best fitted linear curve on the data are also provided in the figures. It can be observed that data dispersion is more in the case of LM algorithm which confirms that uncertainty of LM algorithm is greater than that of ICA. Furthermore, the figures show that ICA is somewhat underestimating the real values. Fig. 5 confirms the results of DDR parameter.

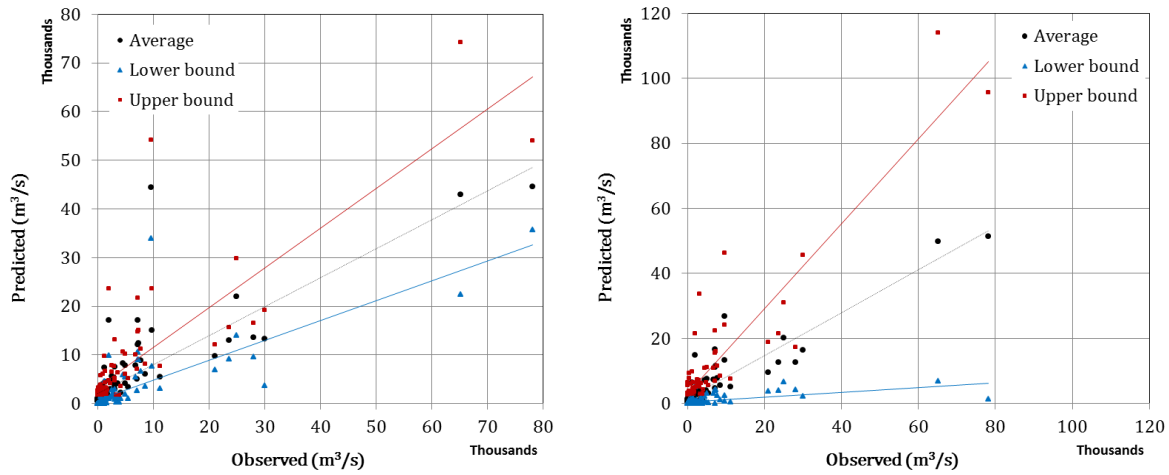


Fig. 5. dispersion^(a) of the predictions in the case of (a) ICA; and ^(b) LM algorithm

5. Summary and conclusion

This investigation focused on evaluating the ICA and LM performance on ANN modeling. ICA is a new evolutionary algorithm, in the evolutionary computation field, based on the human's socio-political evolution and the LM algorithm is a common method of training ANNs called multi-stage dynamic system optimization method. The problem of peak outflow prediction from breached embankment dams was undertaken and 93 case study data were collected from the literatures. According to the results, a three-layer ANN model with three and four neurons in the hidden layer, respectively, for ICA and LM was appropriate for modeling the phenomenon. By considering the statistic indices as well as CPU time taken in the case of each training algorithm, ICA was recognized as the best training algorithm. Although LM usually resulted in satisfactory R^2 values, its error indices were high to some extent that made it not quite reliable. Besides, the ANN performance was compared to an empirical formula (i.e. Froehlich formula). The results indicated that although the Froehlich formula was simply applicable, it led to unsatisfactory results, particularly, for large embankment dams.

Afterward the uncertainty of the ANN models in the developing procedure was evaluated. Monte-Carlo simulation was applied and considering each training algorithm so many models were developed. For each model, initially two sets of data were randomly selected for

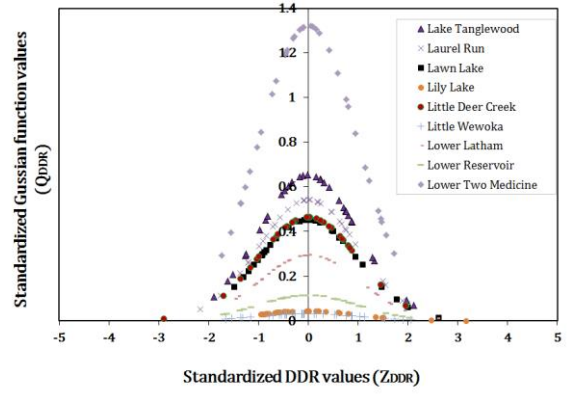
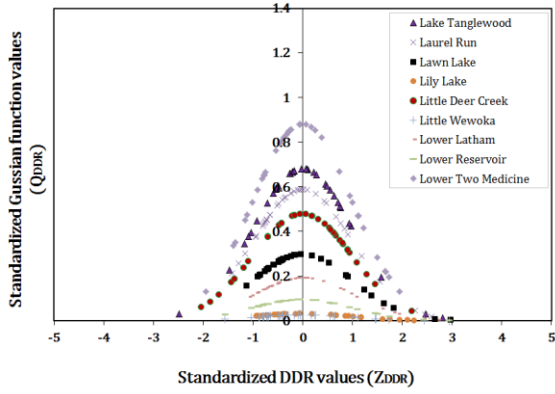
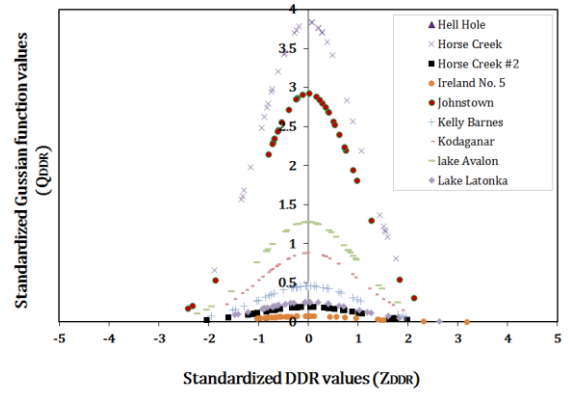
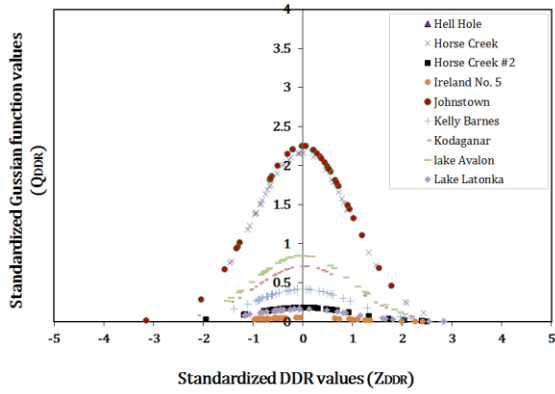
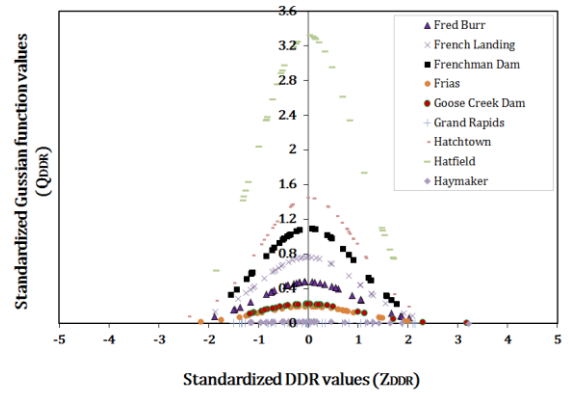
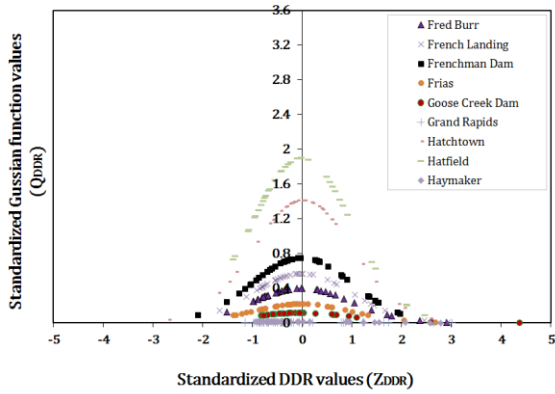
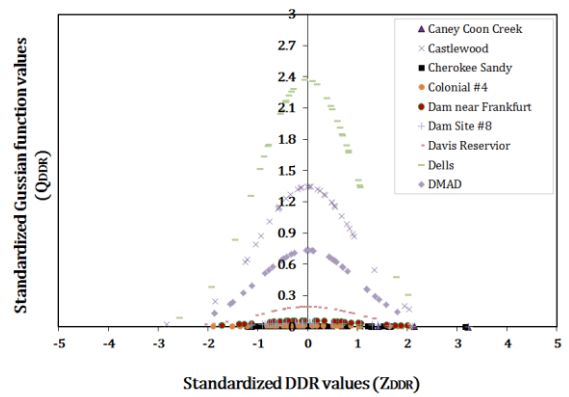
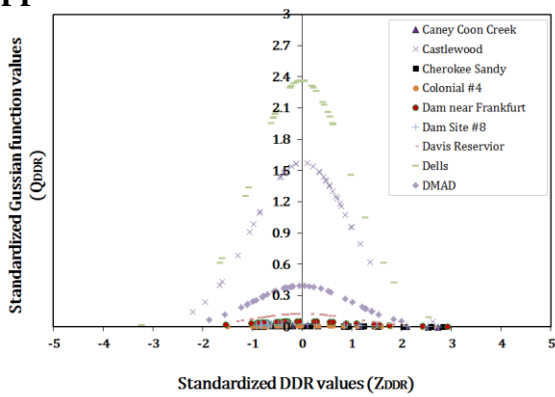
training and validation steps and then an ANN model was developed. This procedure was repeated thousand times and the outputs were stored in each time. Then 95PPU and d -factor parameters were calculated and compared with each other. The quantities showed that the results of LM algorithm is a little better than ICA. So, DDR parameter was applied and quantified, standardized, and normalized. Comparing the results showed that ICA has more acceptable performance than LM algorithm and in total ICA's uncertainty is less than that of LM algorithm.

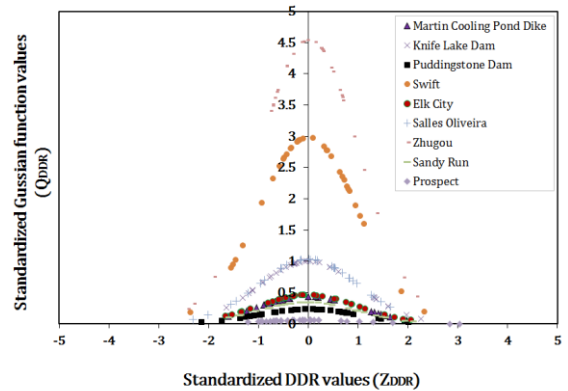
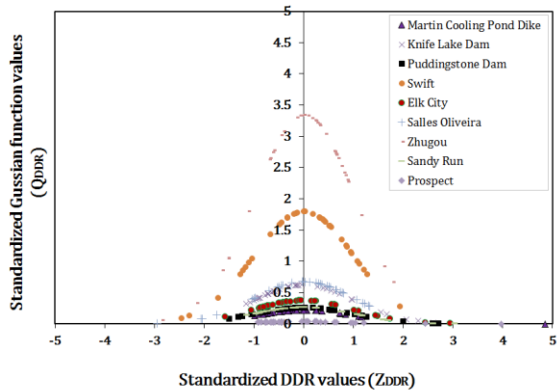
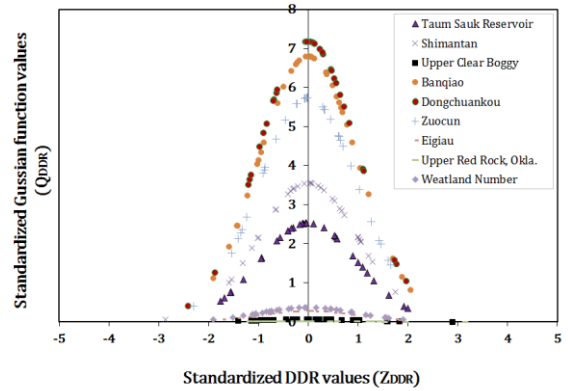
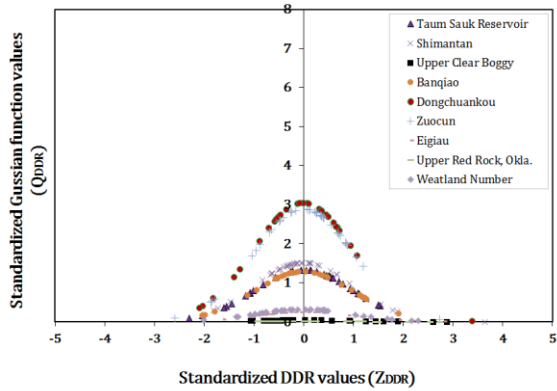
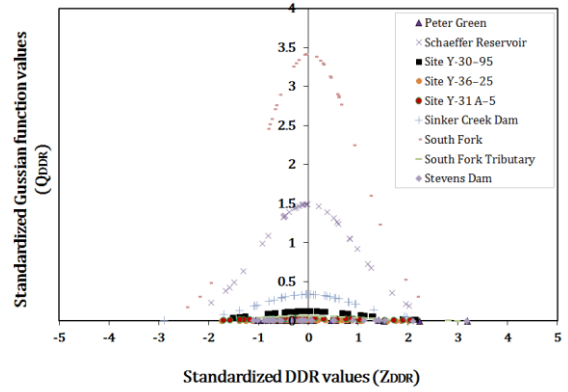
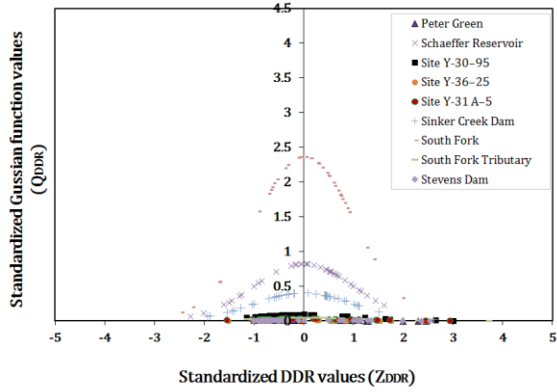
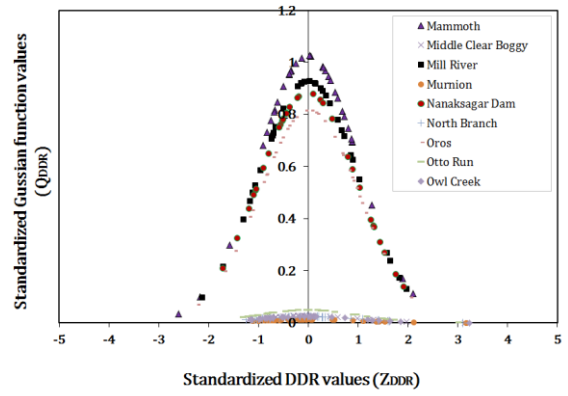
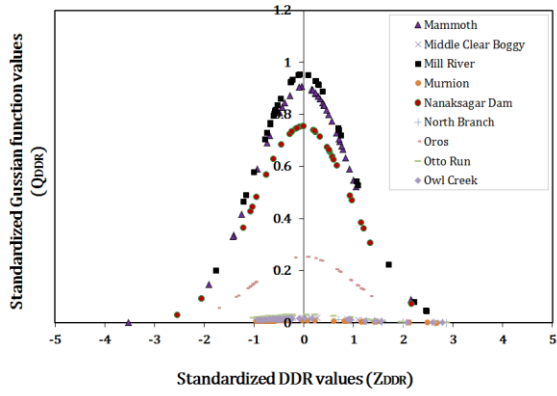
6. References

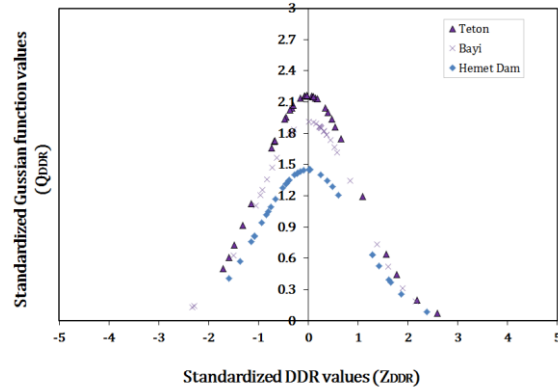
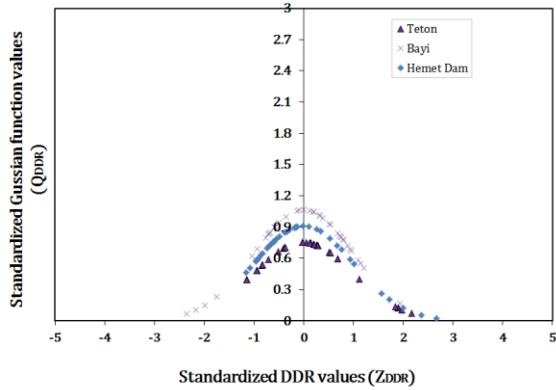
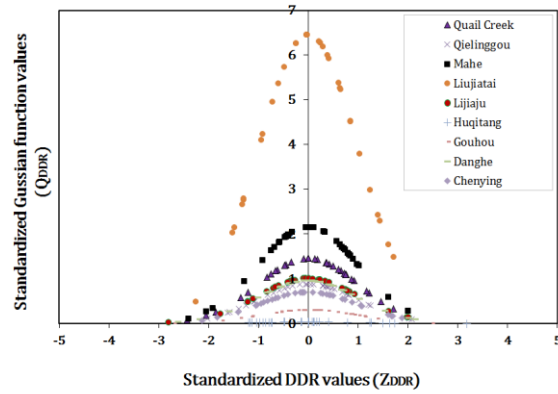
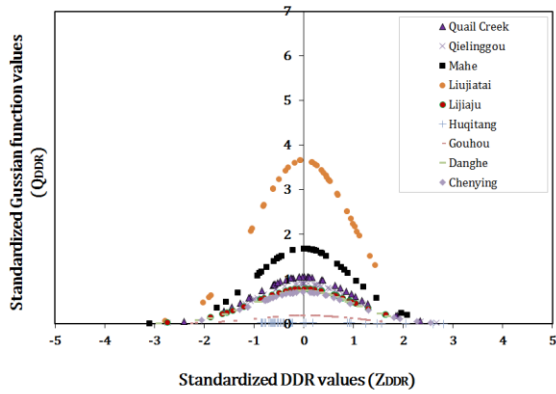
1. Singh VP, Scarlatos PD. Analysis of gradual earth dam failure, *J. Hydraul. Eng* 1988; **114**(1): 21–42.
2. US Bureau of Reclamation *Guidelines for defining inundated areas downstream from bureau of reclamation dams*. Reclamation Planning Instruction, 1982; No. 82-11.
3. Singh KP, Snorrason A. Sensitivity of outflow peaks and flood stages to the selection of dam breach parameters and simulation models, *J. Hydrol* 1984; **68**: 295–310.
4. MacDonald TC, Langridge-Monopolis J. Breaching characteristics of dam failures, *J. Hydraul. Eng* 1984; **110**(5): 567–586.
5. Froehlich DC. Peak outflow from breached embankment dam, *J. Water Resour. Plann. Manage* 1995; **121**(1): 90–97.
6. Coleman S, Andrews D, Webby MG. Overtopping breaching of noncohesive homogeneous embankments. *J. Hydraul. Eng* 2002; **128**(9): 829–838.
7. Xu Y, Zhang LM. Breaching parameters for earth and rockfill dams, *J. Geotech. Geoenviron. Eng* 2009; **135**(12): 1957-1970.
8. Fread DL. Some limitations of dam-break flood routing models. *Preprint, American Society of Civil Engineers*, Fall Convention, St. Louis, Mo. 1981.
9. Avarideh F, Banyhabib MA, Taher-shamsi A. Application of artificial neural networks in river Sediment estimation, *3th Iran Hydraulic Conference*, Kerman, Iran, 2001; 269-275.
10. Azmatullah HMd, Deo MC, Deolalikar PB. Neural networks for estimation of scour downstream of a ski-jump bucket, *J. Hydraul. Eng* 2005; **131**(10): 898–908.
11. Bateni SM, Jeng DS. Estimation of pile group scour using adaptive neuro-fuzzy approach, *Ocean Engineering* 2006; **34**(8-9): 1344–1354.
12. Taher-shamsi A, Menhaj MB, Ahmadian R. Sediment loads prediction using multilayer feedforward neural networks, *Amirkabir Journal of Science and Technology* 2006; **16**(63): 103-110.
13. Zounemat-Kermani M, Beheshti A, Ataie-Ashtiani B, Sabbagh-Yazdi S. Estimation of current-induced scour depth around pile groups using neural network and adaptive neuro-fuzzy inference system, *Appl. Soft Comput* 2009; **9**(2): 746–755.
14. Azamathulla HMd, Ghani A. ANFIS-based approach for Predicting the scour depth at culvert outlets, *J. Pipeline. Syst. Eng. Pract* 2011; **2**(1): 35-40.
15. Taher-shamsi A, Shetty AV, Ponce VM. Embankment Dam Breaching: Geometry and Peak Outflow Characteristic, *Dam Engineering* 2004; **14**(2): 73-87.
16. Wahl TL. *Prediction of embankment dam breach parameters, A literature review and needs assessment*. Bureau of Reclamation, U.S. Department of the Interior, Denver, 1998; 60, Rep. No. DSO-98-004.
17. Pierce MW, Thornton CI, Abt SR. Predicting peak outflow from breached embankment dams, *J. Hydrol. Eng* 2010; **15**(5): 338–349.
18. Lek S, Guegan JF. Artificial neural networks as a tool in ecological modelling, an introduction, *Ecol. Model* 1999; **120**(2-3): 65-73.

19. Cybenko G. Approximation by superposition of a sigmoidal function, *Math. Cont. Sig. Syst* 1989; **2**(4): 303-314.
20. Noori R, Karbassi AR, Mehdizadeh H, Sabahi MS. A framework development for predicting the longitudinal dispersion coefficient in natural streams using artificial neural network, *Environ. Prog. Sustain. Energ* 2011; **30**(3): 439-449.
21. Coulibaly p, Anctil F, Bobe´e B. Daily reservoir inflow forecasting using artificial neural networks with stopped training approach, *J. Hydrol* 2000; **230**(3-4): 244–257.
22. Levenberg K. A method for the solution of certain non-linear problems in least squares, *Quart. Appl. Math* 1944; **2**(2): 164–168.
23. Marquardt DW. An algorithm for least-squares estimation of nonlinear parameters, *J. Soc. Ind. Appl. Math* 1963; **11**(2): 431–441.
24. Atashpaz-Gargari E, Lucas C. Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition, In *IEEE Congress on Evolutionary Computation*, 2007; 4661–4667.
25. Atashpaz-Gargari E, Hashemzadeh F, Rajabioun R, Lucas C. Colonial competitive algorithm, a novel approach for PID controller design in MIMO distillation column process, *Int. J. Intelligent Computing and Cybernetics* 2008; **1**(3): 337–355.
26. Kaveh A, Talatahari S. Optimum Design of Skeletal Structures Using Imperialist Competitive Algorithm, *Computers and Structures* 2010; **88**(21-22):1220-1229.
27. Tahershamsi A, Sheikholeslami R. Optimization to Identify Muskingum Model Parameters Using Imperialist Competitive Algorithm, *Int. J. Optim. Civil Eng* 2011; **1**(3): 473-482.
28. Froehlich, D. (2008).”Embankment Dam Breach Parameters and Their Uncertainties.” *J. Hydraul. Eng.*, 134(12), 1708–1721.
29. Hooshyaripor, F., Tahershamsi, A., 2012. Comparing the Performance of Neural Networks for Predicting Peak Outflow from Breached Embankments when Back Propagation Algorithms meet Evolutionary Algorithms. *Int. J. Hydraul. Eng.*, 1(6), December 2012.
30. Nourani, V., Hakimzadeh, H., Babaeyan Amini, A., 2012. Implementation of artificial neural network technique in the simulation of dam breach hydrograoh. *Journal of Hydroinformatics*. 14(2), 478–496.
31. Abbaspour, K.C., Yang, J., Maximov, I., Siber, R., Bogner, K., Mieleitner, J., Zobrist, J., Srinivasan, R., 2007. Modeling hydrology and water quality in the pre-alpine/alpine Thur watershed using SWAT. *J. Hydrol.* 333, 413-430.
32. Tibshirani, R., 1994. A Comparison of Some Error Estimates for Neural Network Models. Technical Working Paper No. 94-100, Department of Statistics, University of Toronto.
33. Noori R., Khakpour A., Omidvar B., and Farokhnia A. 2010 Comparison of ANN and principal component analysis-multivariate linear regression models for predicting the river flow based on developed discrepancy ratio statistic. *Expert Systems with Applications*, Vol. 37, 5856–5862
- 34.
- 35.

Appendix I







Appendix II

| Dam Name | LM | ICA | Dam Name | LM | ICA | Dam Name | LM | ICA |
|--------------------|------|------|-----------------|------|------|--------------------|------|------|
| Caney Coon Creek | 0.01 | 0.01 | Frias | 0.22 | 0.20 | Kodaganar | 0.71 | 0.88 |
| Castlewood | 1.57 | 1.35 | Goose Creek Dam | 0.11 | 0.22 | lake Avalon | 0.85 | 1.28 |
| Cherokee Sandy | 0.00 | 0.01 | Grand Rapids | 0.00 | 0.01 | Lake Latonka | 0.16 | 0.25 |
| Colonial #4 | 0.01 | 0.01 | Hatchtown | 1.41 | 1.45 | Lake Tanglewood | 0.68 | 0.65 |
| Dam near Frankfurt | 0.05 | 0.06 | Hatfield | 1.90 | 3.32 | Laurel Run | 0.59 | 0.54 |
| Dam Site #8 | 0.03 | 0.04 | Haymaker | 0.01 | 0.02 | Lawn Lake | 0.30 | 0.45 |
| Davis Reservoir | 0.12 | 0.19 | Hell Hole | 1.03 | 1.98 | Lily Lake | 0.03 | 0.04 |
| Dells | 2.36 | 2.37 | Horse Creek | 2.16 | 3.84 | Little Deer Creek | 0.48 | 0.46 |
| DMAD | 0.39 | 0.73 | Horse Creek #2 | 0.18 | 0.19 | Little Wewoka | 0.03 | 0.03 |
| Fred Burr | 0.40 | 0.48 | Ireland No. 5 | 0.05 | 0.07 | Lower Latham | 0.19 | 0.29 |
| French Landing | 0.57 | 0.76 | Johnstown | 2.25 | 2.92 | Lower Reservoir | 0.10 | 0.11 |
| Frenchman Dam | 0.75 | 1.09 | Kelly Barnes | 0.41 | 0.46 | Lower Two Medicine | 0.88 | 1.32 |

| Dam Name | LM | ICA | Dam Name | LM | ICA | Dam Name | LM | ICA |
|--------------------|------|------|----------------------|------|------|--------------------------|------|------|
| Mammoth | 0.91 | 1.03 | Site Y-36-25 | 0.00 | 0.00 | Eigiau | 0.25 | 0.27 |
| Middle Clear Boggy | 0.02 | 0.03 | Site Y-31 A-5 | 0.02 | 0.03 | Upper Red Rock, Okla. | 0.00 | 0.01 |
| Mill River | 0.95 | 0.93 | Sinker Creek Dam | 0.40 | 0.34 | Weatland Number | 0.32 | 0.36 |
| Murnion | 0.01 | 0.01 | South Fork | 2.36 | 3.41 | Martin Cooling Pond Dike | 0.22 | 0.44 |
| Nanaksagar Dam | 0.75 | 0.88 | South Fork Tributary | 0.04 | 0.06 | Knife Lake Dam | 0.62 | 1.00 |

| | | | | | | | | |
|---------------------|------|------|---------------------|------|------|------------------|------|------|
| North Branch | 0.02 | 0.02 | Stevens Dam | 0.00 | 0.00 | Puddingstone Dam | 0.26 | 0.23 |
| Oros | 0.25 | 0.81 | Taum Sauk Reservoir | 1.33 | 2.53 | Swift | 1.80 | 2.97 |
| Otto Run | 0.03 | 0.05 | Shimantan | 1.51 | 3.55 | Elk City | 0.37 | 0.46 |
| Owl Creek | 0.02 | 0.02 | Upper Clear Boggy | 0.04 | 0.06 | Salles Oliveira | 0.67 | 1.03 |
| Peter Green | 0.00 | 0.00 | Banqiao | 1.28 | 6.79 | Zhugou | 3.34 | 4.54 |
| Schaeffer Reservoir | 0.82 | 1.49 | Dongchuankou | 3.03 | 7.17 | Sandy Run | 0.27 | 0.34 |
| Site Y-30-95 | 0.09 | 0.12 | Zuocun | 2.88 | 5.73 | Prospect | 0.03 | 0.06 |

| Dam Name | LM | ICA |
|-----------------|-----------|------------|
| Quail Creek | 1.05 | 1.46 |
| Qielinggou | 0.90 | 0.88 |
| Mahe | 1.69 | 2.16 |
| Liujiatai | 3.66 | 6.46 |
| Lijiaju | 0.78 | 1.01 |
| Huqitang | 0.03 | 0.04 |
| Gouhou | 0.19 | 0.31 |
| Danghe | 0.75 | 0.96 |
| Chenying | 0.71 | 0.70 |
| Teton | 0.75 | 2.16 |
| Bayi | 1.07 | 1.91 |
| Hemet Dam | 0.91 | 1.45 |