Support Vector Machine to predict the discharge coefficient of Sharp crested w-planform weirs

A. Parsaie, A. H. Haghiabi*

Water Engineering Department, Lorestan University, Khorramabad, Iran

**ABSTRACT:** In this paper, the discharge coefficient ($C_d$) of triangular labyrinth weir was predicted using Multilayer Perceptron Neural Network (MLPNN), Radial Basis Neural Network (RBFNN) and support vector machine (SVM). To this end, 223 data sets related to the effective parameters on $C_d$ were collected. Using dimensional analysis techniques, the involved dimensionless parameters on $C_d$ were derived. To find out the most effective parameters on $C_d$, the Gamma test (GT) was analyzed. Results of GT demonstrated that $H/P$, $L_w/L_c$, and $L_w/W$ are the most effective parameters on $C_d$. To develop ANN and SVM, different types of transfer and kernel functions were tested. During the testing of transfer and kernel functions for developing the ANN and SVM models, respectively, it was found that tansig and RBFNN have the best performance for predicting the $C_d$. Overall evaluation of the results of developed models indicated that both models have a suitable accuracy in predicting the $C_d$; however, the SVM is a bit more accurate. Comparing the outcomes of the applied models in terms of DDR index shows that the data dispersivity of SVM is less than the others; therefore, this model is more reliable.

1- Introduction

Labyrinth weir is a novel approach for improving the hydraulic efficiency of weirs considered instead of conventional linear weirs. Due to the effect of climate change on regime of river flow and increasing the probability of occurrence of the probable maximum flood (PMF), improving the discharge capacity of weirs specifically in earthen dam is necessary [1]. This approach has been proposed to insert the existing weirs in dam projects to improve their discharge capacity. Several approaches related to increasing the discharge capacity of weirs have been proposed whereas the labyrinth weir is the most practical approach [2]. Among non-linear weirs, labyrinth weirs have been widely welcome by researchers due to their high efficiency in term of passing the flow especially in low head projects. Comparison of labyrinth weir with conventional sharp linear weir has showed that they are more efficient about 3 to 5 times [3]. Study of labyrinth weirs has been started by Taylor [4]. He investigated several shapes such as triangular, trapezoidal and rectangular for the crest of labyrinth weirs and found that the trapezoidal crest is more efficient compared to others. From Taylor [4] to now several studies have been conducted on the labyrinth weir. In this regard, conducted studies by Hay and Taylor [2] on labyrinth weirs and their approach to design labyrinth weir can be mentioned. In follow Houston [5] assessed the method of Hay and Taylor [2] for practical purposes and found that the proposed approach of Hay and Taylor [2] include obvious errors compared to the measured data. Recently, Ghodsian [6] has investigated the effect of rounding the top of the side walls of weirs on its hydraulic efficiency. He stated that rounding the top of side wall has a significant effect on increasing its efficiency. Another type of labyrinth weirs named piano key weir has been proposed to be constructed in places that the footprint of weirs is limited such as top of the dam and narrow channels [7-11]. Due to the high performance of labyrinth weirs, these important roles in hydraulic systems and also complex hydraulic behavior of them, from 2003 to the present day, two international technical conferences have been held [12,13]. It is notable that the concept of labyrinth weir has been used for improving the discharge capacity of side weirs, as well [14]. Due to the high cost of experiments and constructing the scaled laboratory model, investigators have attempted to use mathematical methods for modeling of hydraulic properties of labyrinth weirs [15,16]. In the field of mathematical modeling using the computational fluid dynamic (CFD) and soft computing techniques as the two main parts of mathematical modeling has been reported. In the field of CFD, Robertson [17] has used Flow 3D for numerical modeling of flow over the weirs and showed that this model has suitable performance for modeling the hydraulic properties of labyrinth weirs. Using the optimized radial basis neural network for predicting the discharge coefficient of labyrinth weir has been proposed by Zaji et al. (2015). Based on those studies, the optimized radial basis function neural network has suitable performance for modeling the discharge coefficient. Reviewing the literature showed that the labyrinth weir is an accepted approach for improving the discharge capacity of existing weirs. The main

Corresponding author, E-mail: haghiabi.a@lu.ac.ir
parameter related to labyrinth weir is discharge coefficient. Hence, it is necessary to use powerful soft computing techniques such as support vector machine (SVM) methods to predict this parameter. Therefore, in this study a SVM model is prepared to predict the $C_d$ and in the follow its performance is compared with MLPNN. The performance of developed models (MLPNN and SVM) in this study is compared with applied models in the previous studies (RBFNN model), as well. The MLPNN and SVM are powerful soft computing methods which have been widely used for function fitting, pattern recognition, image processing, etc. [15-17]. These methods have successfully applied in most areas of water engineering such as river sediment load prediction, water quality modeling, river engineering, etc. [15,18].

2- Method and Materials
Discharge coefficient of a triangular labyrinth weir is proportional to the properties of weir geometry and hydraulic conditions. The main geometrical and hydraulic parameters effective the discharge coefficient of sharp crested w-planform weirs are shown in Figure 1. Where $P$ is the weir height, $W_{mc}$ is the main channel width, $W_c$ is the width of one cycle and $H$ is the total Head of flow. Formulation of the involved parameters on discharge coefficient is presented in Equation 1 (Carollo et al., 2012).

$$C_d = f \left( H, W_{mc}, W_c, L_c, P, g, V, \sigma, \mu, \rho \right)$$

In which $L_c$: total length of the weir, $L_c$: length of one cycle, $V$: flow velocity, $g$: gravitational acceleration, $\sigma$ is Surface tension and $\rho$ is the Density of flow. Using the dimensional analysis, including $\pi$ theorem, the influenced dimensionless parameters on the discharge coefficient are derived as Equation 2. It is notable the flow in the channel is turbulence and usually, investigators in hydraulic experience have tried to remove the effect of surface tension, therefore, the Reynolds and Weber numbers can be negligible [6].

$$C_d = f \left( \frac{H}{P}, \frac{W_{mc}}{P}, \frac{L_c}{P}, \frac{g}{P^2}, \frac{\sigma}{P^2}, \frac{\mu}{P}, \frac{\rho}{P^2} \right)$$

Developing the soft computing techniques is based on the dataset. This means to predict a phenomenon by soft computing techniques, its behavior and influential parameters on it should be recorded previously. To predict the $C_d$ 223 data sets were collected from [6]; Kumar et al. [18]. The range of the collected data is given in Table 1.

<table>
<thead>
<tr>
<th>Parameters range</th>
<th>Min</th>
<th>Max</th>
<th>Avg</th>
<th>STDEV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weir length</td>
<td>0.245</td>
<td>1.200</td>
<td>0.475</td>
<td>0.282</td>
</tr>
<tr>
<td>Channel width</td>
<td>0.245</td>
<td>0.300</td>
<td>0.271</td>
<td>0.019</td>
</tr>
<tr>
<td>Cycle width</td>
<td>0.123</td>
<td>0.280</td>
<td>0.213</td>
<td>0.075</td>
</tr>
<tr>
<td>Cycle length</td>
<td>0.123</td>
<td>1.082</td>
<td>0.373</td>
<td>0.263</td>
</tr>
<tr>
<td>Weir height</td>
<td>0.092</td>
<td>0.170</td>
<td>0.110</td>
<td>0.024</td>
</tr>
<tr>
<td>Total head</td>
<td>0.007</td>
<td>0.145</td>
<td>0.046</td>
<td>0.024</td>
</tr>
<tr>
<td>Discharge coefficient</td>
<td>0.148</td>
<td>0.906</td>
<td>0.595</td>
<td>0.172</td>
</tr>
</tbody>
</table>

2- 1- Gamma Test
The Gamma test was used to examine the relationship between inputs and outputs in numerical data-sets without a need to construct the prediction model. The Gamma test is used to estimate the variance of the output before modeling, even though the model is unknown. This error variance estimate presents a target Mean Squared Error that any smooth non-linear function should attain on unseen data. Suppose we have a set of observed data represented by:

$$\left( x_1, \ldots, x_M, y \right) = (x, y)$$

where the vector $X=(x_1, \ldots, x_M)$ is the input, confined to a closed bounded set $C \in \mathbb{R}^M$ and the scalar $y$ is the corresponding output, without loss of generality. The only assumption made is that the relationship of the system is in the following form:

$$y = f \left( x_1, \ldots, x_M \right) + r$$

where $f$ represents a smooth function and $r$ denotes an indeterminable part, which may be due to the real noise of lack of functional determination in the assumed input/output relationship.

---

Fig. 1. Sketch of triangular labyrinth weir
relationship. The Gamma test was used to return a data-derived estimate for Var(\(\gamma\)) without knowing the underlying function \(f\), just directly from the data. The estimate of the model’s output variance called the Gamma statistic and represented by \(\Gamma\) cannot be accounted for by a smooth data model. The Gamma test is derived from the Delta function of the input vectors:

\[
\gamma_{\delta\eta}(k) = \frac{1}{2M} \sum_{i=1}^{M} (y_{N[i,\eta]} - y_i)^2
\]

where \(x_{N[i,\eta]}\) denotes the index of the \(k\)th nearest neighbour to \(x_i\) and \(|\cdot|\) denotes Euclidean distance. Thus \(\delta_{\eta}(k)\) is the mean square distance to the \(k\)th nearest neighbour. The corresponding Gamma function of the output values is:

\[
\gamma_{\delta\eta}(k) = \frac{1}{2M} \sum_{i=1}^{M} (y_{N[i,\eta]} - y_i)^2
\]

The Gamma test computes the mean-squared \(k\)th nearest neighbour distances \(\delta_{\eta}(k)\), \((1 \leq k \leq k_{\text{NN}})\) and the corresponding \(\gamma_{\eta}\). In order to compute \(\Gamma\) the best line is constructed for the \(p\) points\((\delta_{\eta}(k)\), \(\gamma_{\eta}(k)\)) and the vertical intercept, \(\Gamma\) is returned as the gamma value. The regression line slope is also returned to show the complexity of the model \(f\). The \(V_{\text{ratio}}\) is the standardized results by considering \(\Gamma/\text{Var}(\gamma)\). It returns a scale invariant noise estimate which normally lies between zero and one (Noori et al. 2011).

2- 2- Artificial Neural Network techniques

ANN is a non-linear mathematical model that is able to simulate arbitrarily complex non-linear processes that relate the inputs and outputs of any system. In many complex mathematical problems that lead to solving complex non-linear equations, Multilayer Perceptron Networks are the common types of ANN that are widely used in the research studies. To use MLP model, the definition of appropriate functions, weights and bias should be considered. Due to the nature of the problem, different activity functions in neurons can be used. An ANN may have one or more hidden layers. Inputs introduced to each neuron are multiplied in weights \((w)\) and then summed by a constant value called bias \((b)\), then passed through a transfer function. Weight and biases’ values will be justified progressively and corrected during training process comparing the predicted outputs with the known outputs. Such networks are often trained using back propagation algorithm. In the present study, ANN is trained by Levenberg–Marquardt technique because this technique is more powerful and faster than the conventional gradient descent technique (Parsaie et al. 2017; Parsaie and Haghjibi 2017).

2- 3- Radial Basis Function (RBFNN) Neural Network

RBFNN is a type of MLPNN that contains only three layers with a feed-forward structure. The first layer is used for input introduction. The last layer is used for summarization of mathematical operation of the hidden layer. The hidden layer is used for main mathematical computation to map input features to the output. The hidden layer can get numbers of neurons. The governing function of neurons is Radial Basis Function (RBF). To design RBFNN, it is required to justify the number of neurons in the hidden layer. The aim of RBFNN model training is mapping the input space to output space as \(f : R^n \rightarrow R\). The transfer function of the RBFNN model is defined as Equation 7.

\[
f(v) = \sum_{i=1}^{n} w_i \varphi(\|v - c_i\|)
\]

Where \(v\) is the inputs variable, \(w_i\) is the weight coefficients, \(\varphi\) is Gaussian function, which is the basic function used as kernel function in RBFNN model development and is defined as Equation 8.

\[
\varphi(v) = e^{-\frac{|v|}{2\sigma^2}}
\]

RBFNN model training usually is carried out by Gradient Descent approach. The aim of RBFNN model is defining the value of kernel function parameters and weights. The initial value of weights is defined randomly. The error for each sample of the data set is calculated by Equation 9.

\[
e_i = t_i - y_i = t_i - \sum_{j=1}^{n} w_j \varphi(\|v - c_j\|)
\]

The error for total input data set is calculated as Equation 10.

\[
E = \frac{1}{2} \sum_{i=1}^{n} e_i^2
\]

RBFNN model preparation is finished when the error of RBFNN model for all data sets is lower than the threshold error which is defined by the designer (Liu 2013; Parsaie and Haghjibi 2015; Parsaie et al. 2016).

2- 4- Support vector regression

SVMs are a set of related supervised learning methods used for classification and regression. In many applications, a non-linear classifier provides a better accuracy. In SVM, the input \(x\) is first mapped onto a m-dimensional feature space using some fixed (non-linear) mapping, and then a linear model is constructed in this feature space. The naive way of making a non-linear classifier out of a linear classifier is to map our data from the input space \(X\) to a feature space \(F\) using a non-linear function \(\varphi : x \rightarrow f\) In the space \(F\), the discriminant function is:

\[
f(x) = w^T \varphi(x) + b
\]

Using mathematical notation, the linear model (in the feature space) \(f(x, w)\) is given by:

\[
w = \sum_{i=1}^{n} \alpha_i x_i
\]

\[
f(x, w) = \sum_{j=1}^{n} \alpha_j x_i \varphi_j(x) + b
\]

\[
f(x) = \sum_{i=1}^{n} \alpha_j x_i^T x + b
\]

In the feature space, \(F\), this expression takes the form:
There are many kernel functions in SVM, hence how to select a good kernel function is also a research issue. However, for general purposes, there are some popular kernel functions.

I. Linear kernel: \( k(x_1, x_2) = x_1^T x_2 \)

II. Polynomial kernel: \( k(x_1, x_2) = (\gamma x_1^T x_2 + r)^d, \gamma > 0 \)

III. RBF kernel: \( k(x_1, x_2) = \exp(-\gamma x_1^T x_2), \gamma > 0 \)

IV. Sigmoid kernel: \( k(x_1, x_2) = \tanh(\gamma x_1^T x_2 + r), \gamma > 0 \)

Here, \( \gamma, r \) and \( d \) are kernel parameters. It is well known that SVM generalization performance (estimation accuracy) depends on a good set of meta-parameters parameters \( \gamma, r \) and \( d \) are the kernel parameters. The choices \( \gamma, r \) and \( d \) control the prediction (regression) model complexity. The problem of optimal parameter selection is further complicated because the SVM model complexity (and hence its generalization performance) depends on all three parameters. Kernel functions are used to change the dimensionality of the input space to perform the classification [20,26,27].

3- Results and Discussion

In this study, to find out the most effective factors on \( C_f \) using GT, different scenarios were considered. In each scenario, the effect of one of the input variables was evaluated. Firstly, at scenario number one, all variables were included in GT analysis and in the next scenarios, one of them was removed and again the GT was analyzed. The results of scenarios are given in Table 2. The GT parameters such as gamma, gradient, standard error and V-ratio were chosen as criteria to define the most effective factors. The scenario which had a minimum value for the GT parameters include the most influential parameters on \( C_f \). The variation of the V-ratio is between the 0 and 1. This point is notable that whatever this factor attends to zero shows that related scenario could accurately predict the output.

Reviewing Table 2 shows that scenario number (1) that involves all input variables has a minimum value for the GT parameters. Table 2 shows removing parameters such as \( H/P, L_1/L_2 \), and \( L_1/W_c \) causes the Gamma value to increase significantly; therefore, it was found that these parameters are the most important parameters on \( C_f \). The variation of gamma, along the standard error values through the data set based on all input variables are shown in Figure 2. Figure 2 shows that the standard error curve and gamma curve are almost flat after point 180. It means for modeling discharge coefficient regarding the collected data set, qualification of 180 data set (80 percent of all dataset) is enough.

Table 2. Results of gamma test analysis in the absence of one variable

<table>
<thead>
<tr>
<th>Row</th>
<th>absence</th>
<th>Gammas</th>
<th>Gradient</th>
<th>Standard error</th>
<th>V-ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-</td>
<td>0.0019</td>
<td>0.1557</td>
<td>0.0014</td>
<td>0.0075</td>
</tr>
<tr>
<td>2</td>
<td>H/P</td>
<td>0.0074</td>
<td>0.2218</td>
<td>0.0013</td>
<td>0.0295</td>
</tr>
<tr>
<td>3</td>
<td>L_1/W_c</td>
<td>0.0053</td>
<td>0.2018</td>
<td>0.00016</td>
<td>0.0274</td>
</tr>
<tr>
<td>4</td>
<td>L_1/W_c</td>
<td>0.0033</td>
<td>0.2915</td>
<td>0.00027</td>
<td>0.0133</td>
</tr>
<tr>
<td>5</td>
<td>L_1/W_c</td>
<td>0.0060</td>
<td>0.2619</td>
<td>0.0015</td>
<td>0.0083</td>
</tr>
<tr>
<td>6</td>
<td>w_1/W_f</td>
<td>0.0033</td>
<td>0.1758</td>
<td>0.0011</td>
<td>0.0135</td>
</tr>
</tbody>
</table>

Fig. 2. The variation of Gamma test and standard error with unique data points

collected datasets were divided into two groups as training and testing. The dimensionless parameters which have presented in Equation 2 were considered as inputs and discharge coefficient was desired as model output. Data selection for the preparation of MLPNN model was carried out using a random approach. Based on the GT results, 80 percent of the total dataset was considered for training and remains (20 percent) were used for testing. Training data sets were used for calibration and testing dataset was considered for the model validation. It is notable that the dedicated data must be a good representative of the entire collected data. This means that the range of training and testing should be close together. Designing of structure of MLPNN model is a trial and error process but the experience of the designer reduces the time to try; however, recommendations of the investigators who conducted similar works are useful. In this paper, the recommendations of Parsaie and Haghiabi (2015) was used. Designing of MLPNN model includes the number of hidden layer(s), the number of the neuron(s) in each hidden layer(s), definition of suitable transfer function for the neurons of hidden layer(s), definition of the suitable transfer function for output layer and learning algorithm. To obtain an optimal structure for the MLPNN model, step by step development of MLPNN structure was considered. In this approach, firstly one hidden layer which includes the number of neurons equal to input features is considered. Then, different transfer functions were tested to find out best transfer function with the best performance. After justifying the transfer function, to increase the accuracy of model, increasing the number of hidden layer or (and) number of neurons may be considered. It is notable that the number of the neurons in the hidden layer...
is increased one by one. This process continues to obtain a model with a suitable performance. In this study, various transfer functions such as log-sigmoid (logsig), tan-sigmoid (tansig), and linear (purelin) were tested. All stages of preparation of MLPNN were conducted in Matlab software. The developed MLPNN model consists of two hidden layers with five and three neurons existed in the first and second hidden layers, respectively. The tansig and purelin were considered as hidden layers and output layer transfer functions, respectively. During the development of MLPNN model, it was found that adding the number of hidden layers and neurons in the hidden layers has not a significant effect on increasing the model precision and just increases the computation cost. The structure of the developed MLPNN is shown in Figure 3. It is notable that the Levenberg–Marquardt technique was used to learn MLPNN model. The results of MLPNN model in training and testing stages are shown in Figure 4. In this figure, the results of MLPNN model were plotted versus the observed data. Developing the RBFNN was as similar to MLPNN. This means that the same approach was considered. The structure of the developed RBFNN to predict the discharge coefficient is shown in Figure 3. As shown in this figure, the developed RBFNN model includes one hidden layer which consists of eight neurons. The performance of RBFNN is shown in Figure 4. Comparison the results of RBFNN model with MLPNN shows that the accuracy of this is a bit less than the MLPNN model.

Development of support vector machine (SVM) as a powerful soft computing method is based on the dataset. To this end, the same that had used for developing the RBFNN and MLPNN model was used for SVM preparation. As mentioned in the SVM model section, preparation of SVM includes selection of kernel function, arrangement of dataset for training and testing and also choosing the input variable. For this purpose, all of four kernel functions mentioned in the SVM model section were tested. The results of each kernel function are given in Table 3. As presented in this table, the RBF function is more accurate among the kernel functions. To find out the kernel function with the best performance, all the variables were considered as inputs. During the development of SVM model, it was found that the $\gamma=0.3$ and epsilon=15.30 for RBF kernel function. The structure of developed SVM model for predicting the $C_d$ is shown in Figure 3. Results of SVM model with RBF as kernel function are shown in Figure 4. Reviewing Figure 4 indicates that the best performance is related to SVM model among the applied models. This result is based on error indices. Comparing the obtained results in the study conducted by Haghiabi et al. (2017) reveals that developing the ANFIS model based on the results of GT leads to increasing the precision of modeling and prediction. The accuracy of ANFIS is a bit more accurate than the SVM model. This is due to the facility of ANFIS model to assign more weight to parameters which are more effective on output. To present more information for outcomes through the dataset, another index introduced by Noori et al. (2009) that was named developed discrepancy ratio (DDR) was calculated by Equation 18. The results of DDR are shown in Figure 5. Reviewing Figure 5 shows that the best performance is related to SVM model.

$$DDR = \left( \frac{Predicted \ Value}{Observed \ Value} \right) - 1$$ (18)

As recently stated, the best accuracy among the applied models was related to SVM. Therefore, another scenario for the prediction of $C_d$ is developing the SVM model based on the most effective parameters. It should be noted that the most important parameters were defined previously by GT.

In other words, in a new scenario, the performance of SVM model for the prediction of the $C_d$ with regard to $H/P$, $L_w/L_c$, and $L_w/W_m$ as inputs are investigated. Results of predicting the $C_d$ based on the most effective parameters are shown in Figures 10 and 11. As shown in these figures, the accuracy of SVM model based on the most effective parameters was not significantly reduced.

### 4- Conclusion

Modeling of the hydraulic structure is the main part of hydraulic engineering activities. The estimation of discharge coefficient of hydraulic structure especially weirs is one of the main parameters to design the optimal operational program of hydro-systems. Labyrinth weirs are the novel subjects in the field of hydraulic structure which have been considered as a rational approach for improving the performance of conventional linear weirs. In this paper, the discharge coefficient ($C_d$) of triangular labyrinth weir was predicted using the multilayer perceptron neural network, radial basis neural network and support vector machine. To this end, the involved parameters were derived using dimensional analysis and related dataset were collected from the literature. To define the most effective parameters on the $C_d$, the gamma test (GT) was applied. Results of GT declared that $H/P$, $L_w/L_c$, and $L_w/W_m$ are the most effective parameters for predicting the $C_d$. Results of applied models indicated that all the applied models have an acceptable performance; however, the SVM model was more accurate. The SVM model was prepared in two scenarios, first, all the involved parameters were considered as inputs and in the second

<table>
<thead>
<tr>
<th>Row</th>
<th>Output</th>
<th>Kernel Function</th>
<th>Training $R^2$</th>
<th>Training RMSE</th>
<th>Testing $R^2$</th>
<th>Testing RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$C_d$</td>
<td>RBFNN</td>
<td>0.94</td>
<td>0.039</td>
<td>0.97</td>
<td>0.038</td>
</tr>
<tr>
<td>2</td>
<td>$C_d$</td>
<td>Linear</td>
<td>0.73</td>
<td>0.0678</td>
<td>0.65</td>
<td>0.1360</td>
</tr>
<tr>
<td>3</td>
<td>$C_d$</td>
<td>Polynomial</td>
<td>0.86</td>
<td>0.0539</td>
<td>0.76</td>
<td>0.0970</td>
</tr>
<tr>
<td>4</td>
<td>$C_d$</td>
<td>Sigmoid</td>
<td>0.90</td>
<td>0.0480</td>
<td>0.91</td>
<td>0.0794</td>
</tr>
</tbody>
</table>

Table 3. Summary of SVM results during the development stages

![Fig. 4. Results of applied models in training and testing stages](image)

![Fig. 5. Results of DDR for outcomes of applied models](image)
Fig. 6. Results of SVM based on the most effective parameters as inputs

Fig. 7. Results of SVM based on the most effective parameters as inputs
scenario, the SVM model was prepared based on the most effective parameters that had driven from the GT analysis. Comparison of the performance of SVM model in terms of both scenarios shows that preparation of SVM model based on the most effective parameters has no significant effect on decreasing its accuracy.

**Notation**

- ANN's: Artificial Neural Networks
- $C_d$: Discharge Coefficient
- CFD: Computational Fluid Dynamic
- DDR: Developed Discrepancy Ratio
- $g$: Gravitational Acceleration
- GMDH: Group Method Of Data Handling
- GP: Genetic Progaming
- GT: Gamma Test
- $H$: Total Head
- $L_c$: Length Of One Cycle
- logsig: Log-Sigmoid
- $L_w$: Total Length Of The Weir
- MARS: Multivariate Adaptive Regression Splines
- MLPNN: Multilayer Perceptron Neural Network
- $P$: Weir Height
- PMF: Probable Maximum Flood
- purelin: Linear
- $R^2$: Coefficient Of Determination
- RBF: Radial Basis Function
- RBFNN: Radial Basis Neural Network
- RMSE: Root Mean Square Error
- SVM: Support Vector Machine
- tansig: Tan-Sigmoid
- $V$: Flow Velocity
- $W_c$: Width Of One Cycle
- $W_m$: Main Channel Width
- $\rho$: Density Of Flow
- $\sigma$: Surface Tension

**References**


Please cite this article using:
DOI: 10.22060/ceej.2017.13005.5309