A COMPARISON OF PERFORMANCE OF SEVERAL ARTIFICIAL INTELLIGENCE METHODS FOR ESTIMATION OF REQUIRED ROTATIONAL TORQUE TO OPERATE HORIZONTAL DIRECTIONAL DRILLING

H. Fattahi[*, †] and Z. Bayatzadehfard
Department of Mining Engineering, Arak University of Technology, Arak, Iran

ABSTRACT

Horizontal Directional Drilling (HDD) is extensively used in geothechnical engineering. In a variety of conditions it is essential to predict the torque required for performing the reaming operation. Nevertheless, there is presently not a convenient method to accomplish this task. To overcome this problem, in this research, the application of computational intelligence methods for data analysis named Support Vector Regression (SVR) optimized by differential evolution algorithm (DE) and Adaptive Neuro-Fuzzy Inference System (ANFIS) to estimate of the required rotational torque to operate horizontal directional drilling is demonstrated. Three ANFIS models were implemented, ANFIS–subtractive clustering method (ANFIS-SCM), ANFIS–grid partitioning (ANFIS-GP) and ANFIS–fuzzy c–means clustering method (ANFIS-FCM). The estimation abilities offered using SVR-DE, ANFIS-FCM, ANFIS-SCM, ANFIS-GP were presented by using field data given in open source literatures. In these models, the rotational torque (M) is used as the output parameter, while the length of drill string in the borehole (L), axial force on the cutter/bit (P), rotational speed (revolutions per minute) of the bit (N), the radius for the ith reaming operation (D_i), the mud flow rate (W), the total angular change of the borehole (K_L), and the mud viscosity (V) are the input parameters. To compare the performance of models for rotational torque to operate horizontal directional drilling prediction, the coefficient of correlation (R^2) and mean square error (MSE) of the models were calculated, indicating the good performance of the ANFIS-SCM model.

---

[*]Corresponding author: Department of Mining Engineering, Arak University of Technology, Arak, Iran
[†]E-mail address: H.fattahi@arakut.ac.ir (H. Fattahi)

1. INTRODUCTION

The origin of Horizontal directional drilling (HDD) dates back to the early 1970s. The HDD has been used extensively throughout the world to construct underground pipeline systems [1]. Most pipelines, including those employing HDD, are installed in soil formations for which engineers have accumulated a great amount of experience [2,3]. Reasonable mechanical models and corresponding equations have therefore been developed for calculating various construction related parameters. However; there is a lack of such a methodology in more difficult situations. A major concern of many HDD projects is what amount of rotational torque should be used. It has been established that the required rotational torque at the drill rig depends on different parameters, including length of drill string in the borehole, drilling method, borehole trajectory, geological conditions, rotary speed, axial force on bit, reamer cutter/bit size and type, drilling mud properties and borehole diameter [4,5]. However, relatively little quick research has been done in this area. In the field, Lan et al. [2] utilized regression model for prediction of rotational torque. Although previous effort is appreciated but in many cases, the aforesaid empirical models are not capable of distinguishing the sophisticated structures involved in dataset. In this study, utilize of developed methods such as artificial intelligence methods, which can successfully model the behavior of nonlinear involved in data, is useful. Some research works were carried out using artificial intelligence methods in the areas of drilling engineering. For example, Madandoust et al. [6] used ANFIS model for prediction of the concrete compressive strength by means of core testing. Feili Monfared et al. [7] presented an ANFIS model for advanced prediction of bottomhole circulating pressure in underbalanced drilling operations. Rad et al. [8] predicted the rock mass rating system based on continuous functions using Chaos–ANFIS model.

The main scope of this study is the comparison between several artificial intelligence methods for data analysis named ANFIS–subtractive clustering method (ANFIS-SCM), ANFIS-grid partitioning (ANFIS-GP), ANFIS–fuzzy C-means clustering method (ANFIS-FCM) and support vector regression (SVR) optimized by differential evolution algorithm (DE) to estimate of the required rotational torque to operate horizontal directional drilling is demonstrated.

In every SVR modeling, a series of user-defined parameters exist that required to be chosen by user precisely. Incorrect input of aforementioned parameters by user can lead to erroneous and even deceptive results. Hence, it is crucial to employ a potent optimization algorithm for searching the proper value of these parameters [9]. By now, there have been several optimization algorithms, such as genetic algorithm (GA) inspired by the Darwinian law of survival of the fittest and biogeography-based optimization (BBO) inspired by the migration behavior of island species. Also, recently new optimization algorithms are developed consisting of charged system search (CSS) [10], ray optimization (RO) [11], democratic particle swarm optimization (DPSO) [12], colliding bodies optimization (CBO) [13] and enhanced colliding bodies optimization (ECBO). In this paper, in order to achieve the above goal, DE is applied as the searching strategy for finding the optimal value of user-defined parameters. The prediction abilities offered using models were presented by using field data given in open source literature.

2. MATERIALS AND METHODS

2.1 The methodology of adaptive network-based fuzzy inference system

Jang [14] proposed an ANFIS algorithm, which is based on the Sugeno fuzzy inference model. The ANFIS can construct an input–output mapping based on both the fuzzy if–then rules and the stipulated input–output data pairs. The if-then rules of Sugeno fuzzy inference model are often applied for obtaining the inference of imprecise model, and can be made a conclusion in the indefinite system, which is better than human experience. These if–then rules base on stipulated input–output training data pairs by appropriate membership functions are produced. The ANFIS employs the neural training process to adjust the membership function and the associated parameter that approaches desired data sets [15].

Generally, the ANFIS system includes 5 layers excluding input layer.

Layer 0: this layer is the input layer. It has n nodes where n is the number of inputs to the ANFIS system.

Layer 1: this layer is the fuzzification layer. In this layer membership functions (MFs) of input variables are used. Each node i in this layer generates a membership grades of a linguistic label. For instance, the node function of the ith node that is defined as follows:,

$$Q_i^1 = \mu_{Ai}(x) = \frac{1}{1 + \left[\left(\frac{x - v_i}{\sigma_i}\right)^2\right]^{b_i}} \tag{1}$$

where, x is the input to node i, and Ai is the linguistic label associated with this node; and $\{\sigma_i, V_i, b_i\}$, is the parameter set that changes the shapes of the MF.

Layer 2: each node in this layer calculates the "firing strength" of each rule via multiplication (Eq.2).

$$Q_i^2 = W_i = \mu_{Ai}(x).\mu_{Bi}(y) \quad i = 1,2 \tag{2}$$

Layer 3: this layer is the normalization layer. Nodes are fixed in this layer and are labeled with N, indicating that they play a normalization role. This layer normalizes the strength of all rules according to the equation

$$Q_i^3 = \bar{W}_i = \frac{w_i}{\sum_{j=1}^{2} w_j}, \quad i = 1,2 \tag{3}$$

where w_i is the firing strength of the ith rule which is computed in Layer 2. Node i computes the ratio of the ith rule's firing strength to the sum of all rules' firing strengths.

Layer 4: this is a layer of adaptive nodes. Every node in this layer calculates a linear function where the function coefficients are adapted by using the error function of the multi-

layer feed-forward neural network.

$$Q_i^4 = \overline{W}_i f_i = \overline{W}_i (p_i x + q_i y + r_i) \tag{4}$$

where, $\overline{W}_i$ is the output of layer 3.

Layer 5: this is the output layer that its role is the summation of the net outputs of the nodes in the previous layer. The output is computed as:

$$Q_i^5 = Overall\ Output = \sum \overline{W}_i f_i = \frac{\sum w_i f_i}{\sum w_i} \tag{5}$$

For identify the antecedent MFs, clustering methods are extremely important for explorative data analysis. Three types of these methods (GP, SCM and FCM) are described below.

2.1.1 Grid partitioning method

Grid partition divides the data space into rectangular sub-spaces using axis paralleled partition based on pre-defined number of membership functions and their types in each dimension, as shown in Fig. 1. The wider application of grid partition in FL and FIS is blocked by the curse of dimensions, which means that the number of fuzzy rules increases exponentially when the number of input variables increases. For example, if there are averagely m MF for every input variable and a total of n input variables for the problem, the total number of fuzzy rules is m^n. It is obvious that the wide application of grid partition is threatened by the large number of rules. Grid partition is only suitable for cases with small number of input variables [16,14].
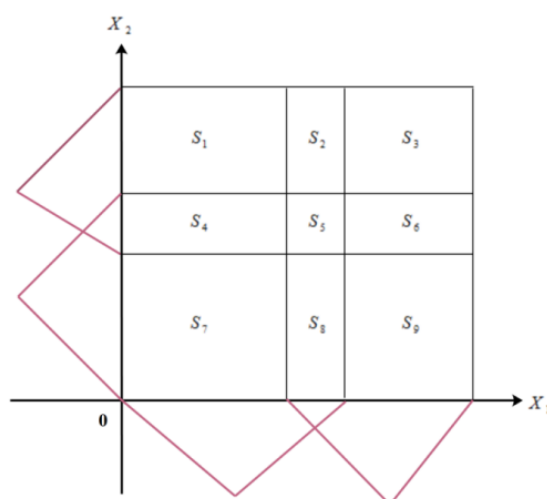

Figure 1. Grid partition of an input domain with two input variables and two membership functions for each input variable [16]

2.1.2 Subtractive clustering method

The subtractive clustering method (SCM) is proposed by Chiu [17], by extending the mountain clustering method [18]. The aim of Chiu's SCM identification algorithm is to estimate both the number and initial location of cluster centers and extract the TSK fuzzy rules from input/output data. SCM operates by finding the optimal data point to define a cluster center based on the density of surrounding data points. This method is a fast clustering method designed for high dimension problems with a moderate number of data points. The algorithm for this method is as follows:

Step 1: Consider a collection of n data points $\{X_1, X_2, X_3, ..., X_n\}$, in an M–dimensional space. Since each data point is a candidate for cluster center, a density measure at data point X_i that is defined as shown in Eq. 6,

$$D_i = \sum_{j=1}^{n} \exp\left(-\frac{\|x_i - x_j\|^2}{\left(\frac{r_a}{2}\right)^2}\right)$$

(6)

where, r_a is a positive constant.

Step 2: The data point with the highest density measure is selected as the first cluster center. Let X_{c1}, be the point selected and D_{c1} its density measure. Next, the density measure for each data point x_i is revised as Eq. 7,

$$D_i = D_i - D_{ci} \exp\left(-\frac{\|x_i - x_{ci}\|^2}{\left(\frac{r_b}{2}\right)^2}\right)$$

(7)

where, r_b is a positive constant.

Step 3: The next cluster center X_{c2} is selected and all of the density calculations for data points are revised again. This process is repeated until a sufficient number of cluster centers are generated.

The combination of ANFIS and subtractive clustering has been widely applied in function approximation and resolving engineering problems [19-21].

2.1.3. Fuzzy C-means clustering method

Fuzzy c-means (FCM) method is proposed by Bezdek [22]. FCM is a method of clustering which allows one piece of data to belong to two or more clusters. This method is frequently used in pattern recognition. The FCM partitions a collection of n vector X_i, $i = 1, 2, ..., n$, into c

fuzzy groups, and finds a cluster center in each group such that a cost function of dissimilarity measure is minimized. The steps of FCM algorithm are therefore, first described in brief.

Step 1: Chose the cluster centers $c_i, i = 1, 2, ..., c$, randomly from the n points $\{X_1, X_2, X_3, ..., X_n\}$.

Step 2: Compute the membership matrix U using the Eq.8,

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{c} (\frac{d_{ij}}{d_{kj}})^{2/m-1}}$$ (8)

where, $d_{ij} = \|c_i - x_j\|$, is the Euclidean distance between i^{th} cluster center and j^{th} data point, and m is the fuzziness index.

Step 3: Compute the cost function according to the Equation (9). Stop the process if it is below a certain threshold.

$$J(U, c_1, ..., c_2) = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \cdot \sum_{j=1}^{n} \mu_{ij}^m d_{ij}^2$$ (9)

Step 4: Compute new c fuzzy cluster centers $c_i, i = 1, 2, ..., c$, using the Equation (10).

$$c_i = \frac{\sum_{j=1}^{n} \mu_{ij}^m x_j}{\sum_{j=1}^{n} \mu_{ij}^m}$$ (10)

go to step 2.

2.2 Hybrid support vector regression with differential evolution algorithm

2.2.1 Support vector regression

Support vector machines (SVMs) has been first proposed by Vapnik [23]. There are two main categories for SVMs: support vector regression (SVR) and support vector classification (SVC). SVM is a learning system using a high dimensional feature space. It yields prediction functions that are expanded on a subset of support vectors. SVM can generalize complicated gray level structures with only a very few support vectors and thus provides a new mechanism for image compression. A version of a SVM for regression has been introduced in 1997 by Vapnik et al. [24]. SVR is the most common application form of SVMs. An overview of the basic ideas underlying SVMs for regression and function

estimation has been given in [25].

Let the training samples be denoted as $XY = \{(x,y)|(x_1,y_1),...,(x_n,y_n)\},$ where n is the number of training samples. In SVR, the ultimate goal is to find linear relation between n-dimensional input vectors $x \in R^n$, and output variables $y \in R$ as follow:

$$f(x) = w^T x + b \tag{11}$$

where, b and w are offset of the regression line and the slope respectively. For determining the values of b and w, it is necessary to minimize following equation;

$$R = \frac{1}{2}\|w\|^2 + \frac{C}{l}\sum_{i=1}^{l}|y_i - f(x_i)|_\varepsilon \tag{12}$$

Loss function, utilized in SVR is ε-insensitive which has been proposed by Vapnik [23] as below;

$$|y_i - f(x_i)|_\varepsilon = \begin{cases} 0 & if \quad |y_i - f(x_i)| \le \varepsilon \\ |y_i - f(x_i)| - \varepsilon & Otherwise \end{cases} \tag{13}$$

This problem can be reformulated in a dual space by;

$$Maximize\, L_p(\alpha_i, \alpha_i^*) = -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)x_i^T x_j - \varepsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)y_i \tag{14}$$

$$Subject\, to \begin{cases} \sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0 \\ 0 \le \alpha_i \le C, \quad i = 1,...,l \\ 0 \le \alpha_i^* \le C, \quad i = 1,...,l \end{cases} \tag{15}$$

where, $\alpha_i, \alpha_i^* \ge 0$ are positive Lagrange multipliers. C is regulated positive parameter which determines trade-off between the weight vector norm $\|w\|$ and approximation error. After calculation of Lagrange multipliers α_i and α_i^*, training data points, those meeting the conditions $\alpha_i - \alpha_i^* \ne 0$, will be applied to construct the decision function. Hence, the best linear hyper surface regression is given by;

$$f(x) = w_0^T x + b = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)x_i^T x + b \tag{16}$$

which desired weight vector of the regression hyper plane is given by;

$$w_0 = \sum_{i=1}^{l} \left(\alpha_i - \alpha_i^* \right) x_i \tag{17}$$

In nonlinear regression, Kernel function is applied for mapping input data onto higher dimensional feature space in order to generate a linear regression hyper plane. In the case of the nonlinear regression, the learning problem is again formulated in the same way as in a linear case, i.e., the nonlinear hyperplane regression function becomes;

$$f(x) = \sum_{i=1}^{l} \left(\alpha_i - \alpha_i^* \right) K(x_i, x) + b \tag{18}$$

where, $K(x_i, x)$ is kernel function which is defined as follow;

$$K(x_i, x_j) = \Phi^T(x_i) \Phi(x_j) \qquad i, j = 1, ..., l \tag{19}$$

where, $\Phi(x_i)$ and $\Phi(x_j)$ are projection of the x_i and x_j in feature space respectively.

One may choose any arbitrary kernel functions, e.g., Radial Basis Function (RBF) $K\left(x_i, x_j\right) = \exp(-\|x_i - x_j\|/2\sigma^2), \sigma > 0$, linear kernel function $K\left(x_i, x_j\right) = \left(x_i, x_j\right)$, polynomial kernel function $K\left(x_i, x_j\right) = \left(\left(x_i, x_j\right) + 1\right)^d, d > 0$, etc. In highly non-linear spaces, RBF kernel usually yields more promising results in comparison with other mentioned kernels [26]. Thus, we use only RBF kernel functions in this study.

2.2.2 SVR-DE model

Differential evaluation (DE) algorithm is one of the evolutionary algorithms, which was introduced by Storn, Price [27]. It has been successfully applied in a wide variety of fields, from computational physics to operations research [28,29]. DE belongs to the class of genetic algorithms (GA) that use the biology-inspired operations of mutation, crossover, and selection on a population to minimize an objective function over the course of successive generations [30,31]. DE uses floating-point instead of bit-string encoding on population members, and arithmetic instead of logical operations in mutation. It has several advantages such as its simple structure, ease of use, speed, and robustness [29]. Fig. 2 presents the process of optimizing the SVR parameters with the DE.
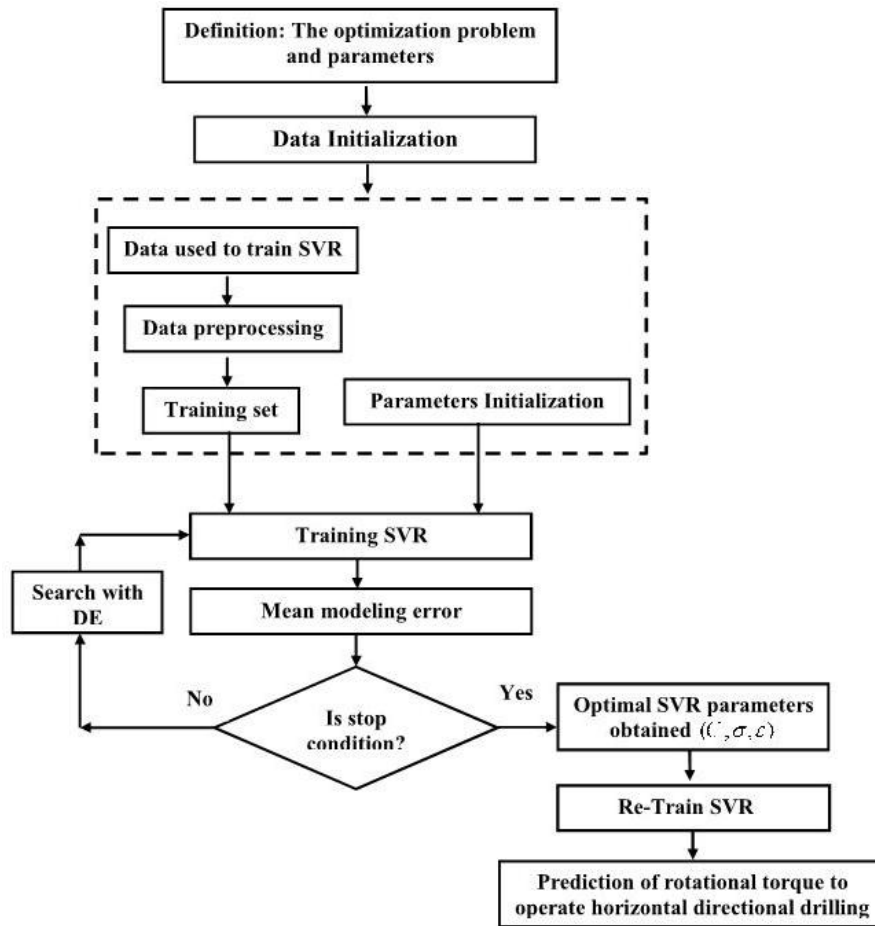
Figure 2. The process of optimizing the SVR parameters with the DE

3. ESTIMATION OF REQUIRED ROTATIONAL TORQUE TO OPERATE HORIZONTAL DIRECTIONAL DRILLING

3.1. Database information

The main scope of this work is to implement the above methodology in the problem of rotational torque estimation. Dataset applied in this study, given in previous paper is borrowed [2]. The collected data sets used to construct the database are from West–East Natural Gas Transmission Project in China. A total of 84 data sets were collected. Each data set contains the parameters of the axial force on the cutter/bit (P), rotational speed (r/min) of the bit (N), the length of drill string in the borehole (L), the total angular change of the borehole (K_L), the radius for the ith reaming operation (D_i), the mud flow rate (W), the mud viscosity (V) and the rotational torque (M). Partial dataset used in this study are presents in Table 1. Also, descriptive statistics of the all data sets are shown in Table 2.

Table 1: Partial dataset used for constructing the artificial intelligence models [2]

Case No.	Input parameters						Output parameter	
	P (KN×10)	N (r/min)	L (m)	K_L	D_i (mm)	W (L/min)	V (s)	M (KN.m)
1	11.5	15	146.01	1.177	558.8	3400	62	13
2	12	15	251.27	2.3672	558.8	3400	65	16.5
3	8.5	18	260.78	2.3777	762	3400	64	15.5
4	11.5	18	316.82	2.7888	762	3400	64	24.5
5	8	15	201.32	1.7156	558.8	2260	53	9.5
6	13	15	287.13	2.3456	558.8	2260	53	10.5
7	8.5	15	230.12	1.9308	762	3400	51	11
8	12	18	345.53	2.8973	762	3400	65	19
9	10	15	324.95	2.7761	762	3400	61	12
10	5	15	258.36	2.0609	914.4	3400	53	15.5

Table 2: Statistical description of dataset utilized for construction of the artificial intelligence models

Parameter	Min	Max	Average
P (KN×10)	2	30.50	13.84
N (r/min)	15	50	31.58
L (m)	116.68	586.06	322.56
K_L	1.09	3.54	2.42
D_i (mm)	457.2	1117.6	760.79
W (L/min)	500	4000	2233.09
V (s)	42	88	63.51
M (KN.m)	4	40	21.02

3.2. Pre-processing of data

In data-driven system modeling methods, some pre-processing steps are commonly implemented prior to any calculations, to eliminate any outliers, missing values or bad data. This step ensures that the raw data retrieved from database is perfectly suitable for modeling. In order to softening the training procedure and improving the accuracy of prediction, all data samples are normalized to adapt to the interval [0, 1] according to the following linear mapping function:

$$x_M = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{20}$$

where x is the original value from the dataset, x_M is the mapped value, and x_{min} (x_{max}) denotes the minimum (maximum) raw input values, respectively.

3.3. Estimation of required rotational torque using ANFIS models

In this paper, all programs for ANFIS modeling and model validation (for estimation of rotational torque) were written in MATLAB. Fig. 3 shows the ANFIS topology. A dataset that includes 84 data points was employed in current study, while 59 data points (70%) were applied for building the model and the remainder data points (25 data points) were used for calculation of degree of accuracy. The specifications of the ANFIS models are illustrated in Table 3. Also The optimal parameters of the ANFIS models is shown in Table 4. Also, for different ANFIS models (ANFIS-GP, ANFIS-SCM and ANFIS-FCM), the MFs of the input parameters are shown in Figs. 4 to 6.
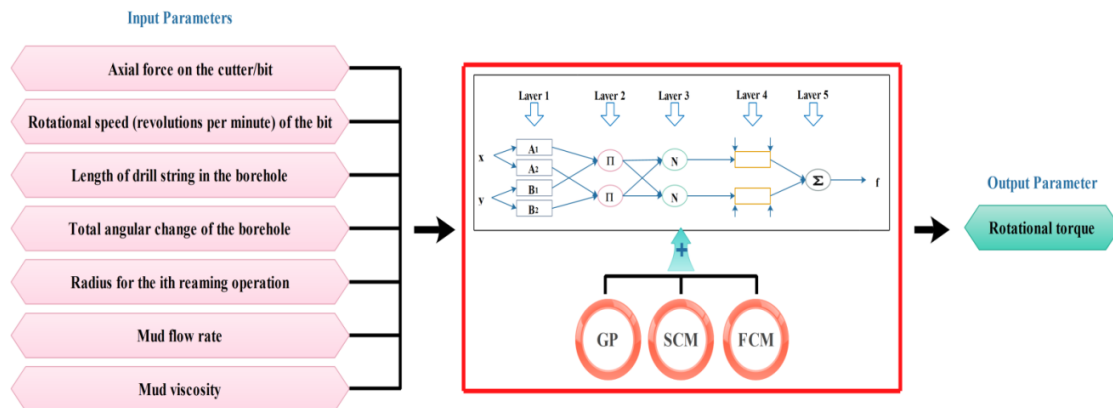
Figure 3. Architecture of the ANFIS based on the GP, SCM and FCM

Table 3: Characterizations of the ANFIS models

ANFIS parameter	ANFIS–GP	ANFIS–SCM	ANFIS–FCM
MF type	Gaussian	Gaussian	Gaussian
Output MF	Linear	Linear	Linear
Number of nodes	294	410	650
Number of linear parameters	1024	200	320
Number of nonlinear parameters	28	350	560
Total number of parameters	1052	550	880
Number of training data pairs	59	59	59
Number of testing data pairs	25	25	25
Number of fuzzy rules	128	25	40

Table 4: The optimal parameters of the ANFIS models

parameter	ANFIS–GP	ANFIS–SCM	ANFIS–FCM
Error goal	0	0	0
The initial step size	0.01	0.01	0.01
Step size decrease rate	0.5	0.5	0.5
Step size increase rate	1.3	0.9	0.9
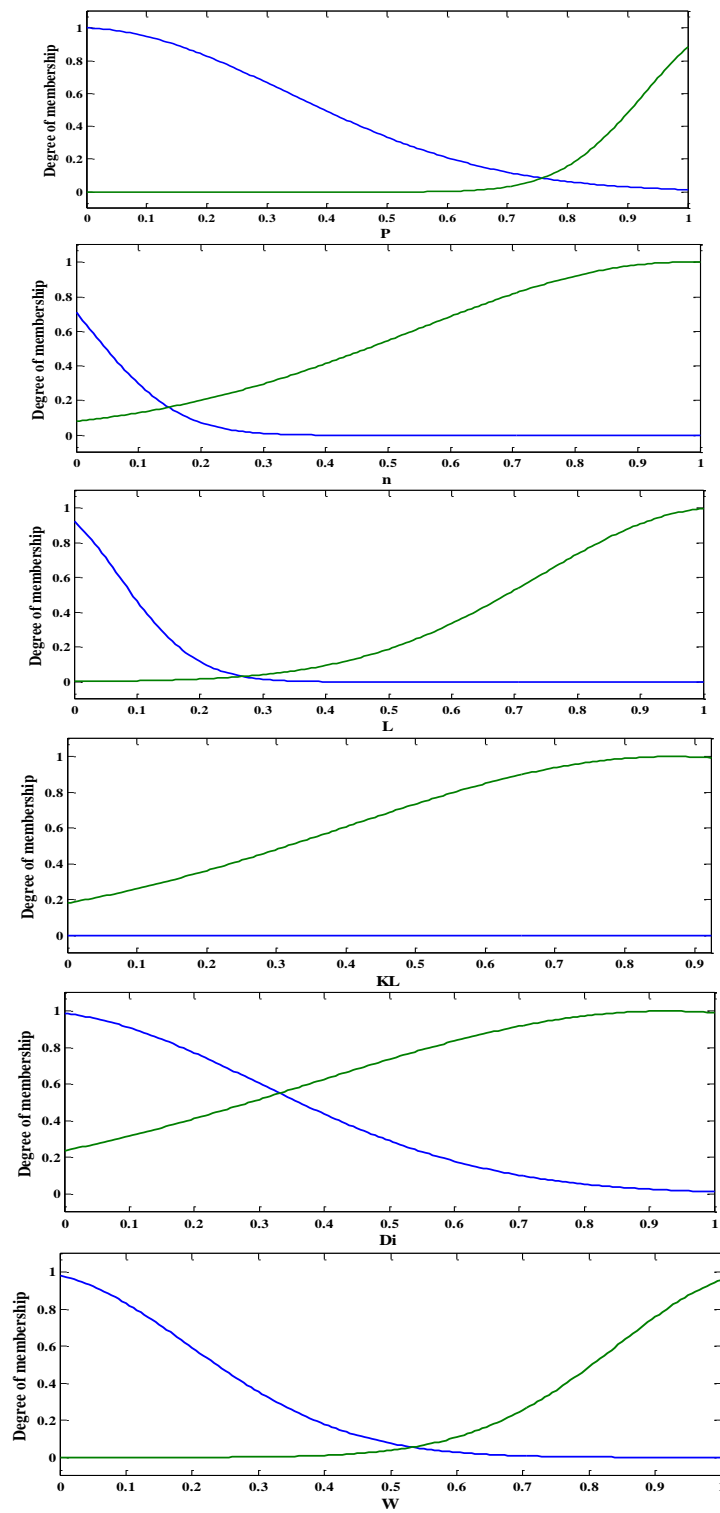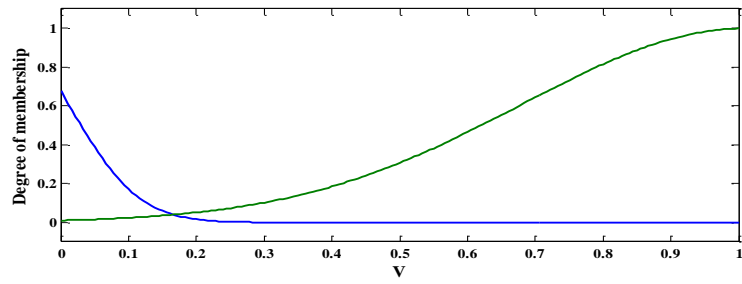
Figure 4. MFs obtained by ANFIS-GP model
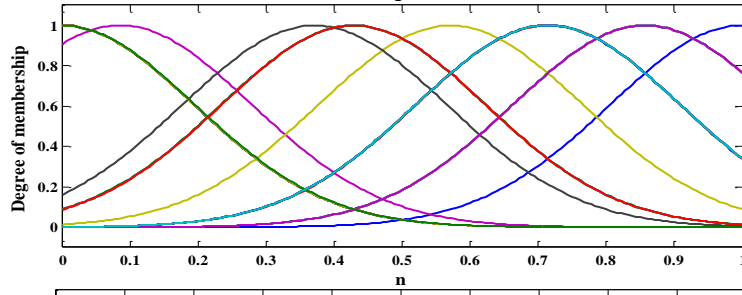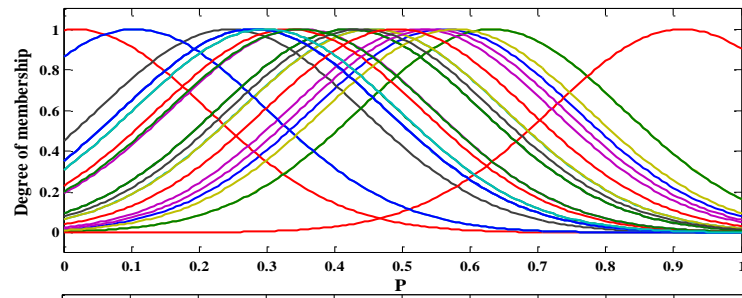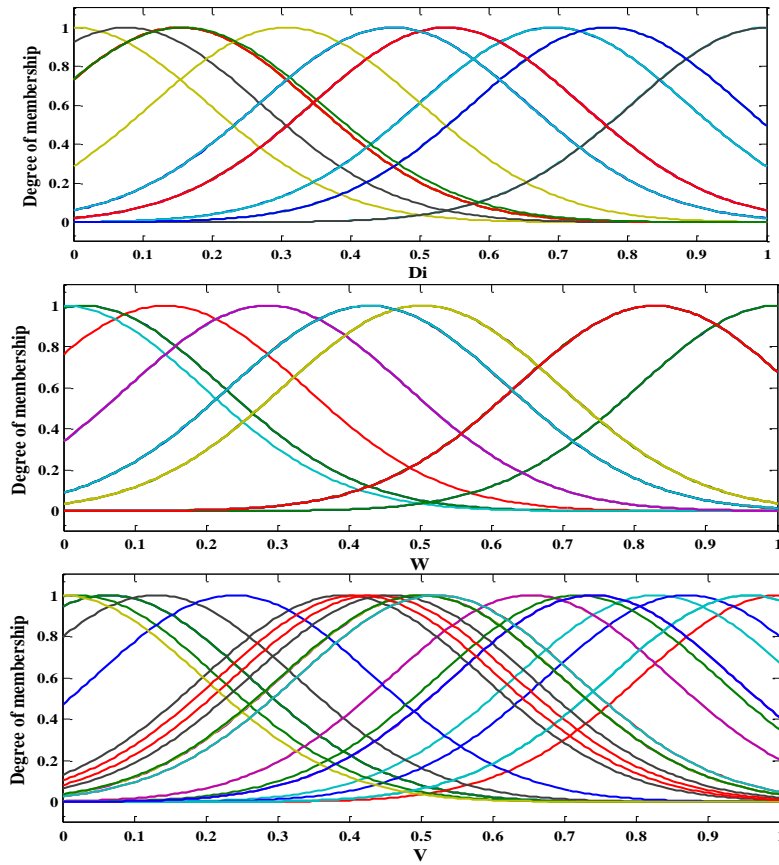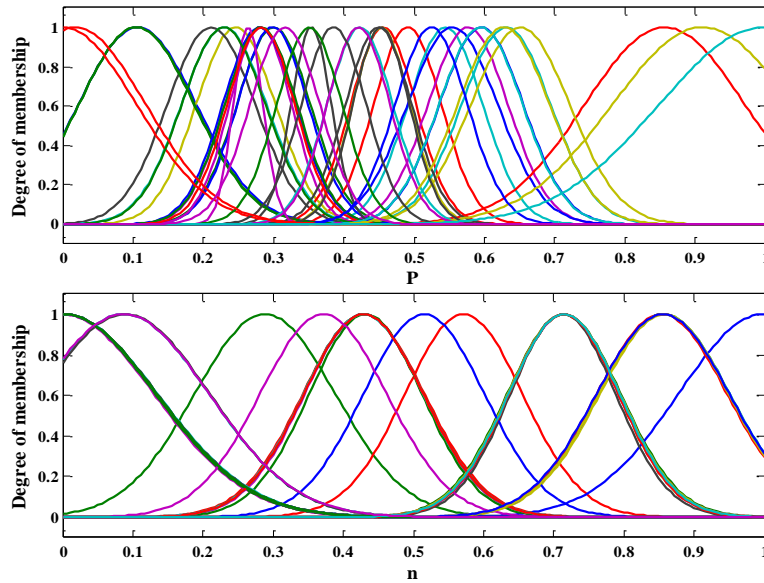
H. Fattahi and Z. Bayatzadehfard
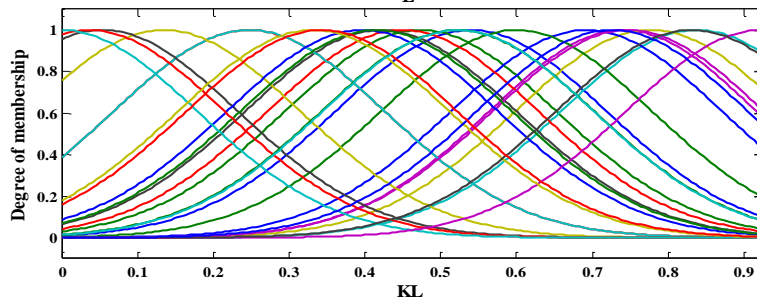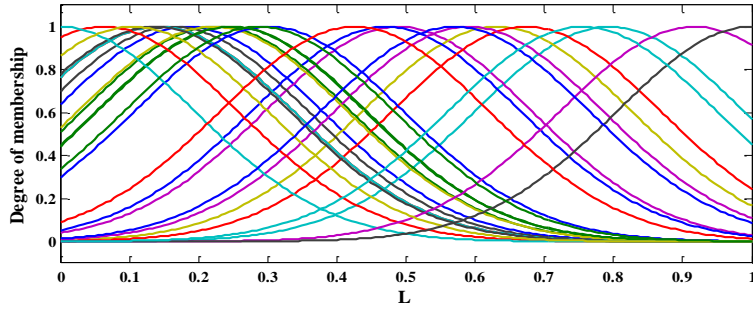
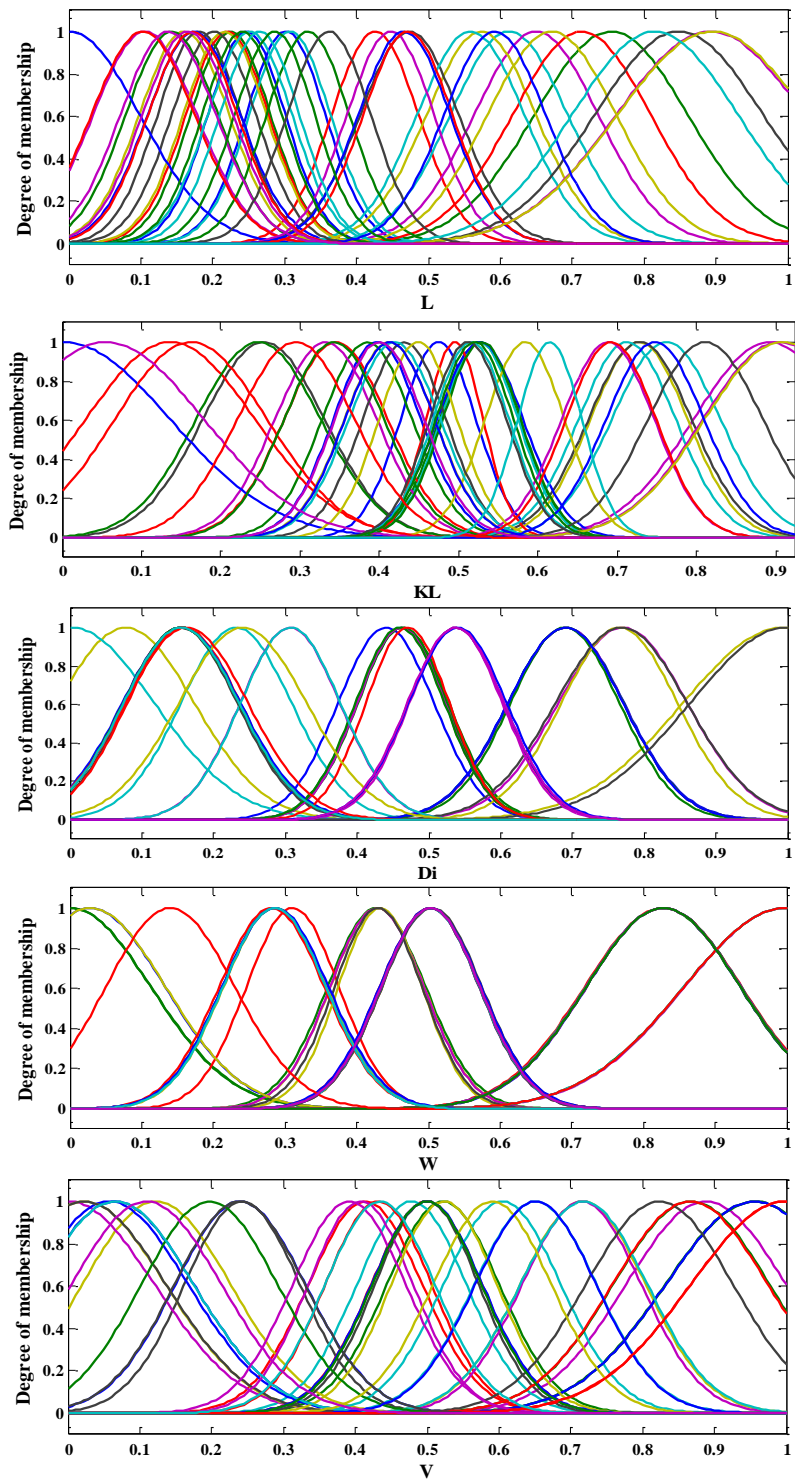Figure 5. MFs obtained by ANFIS-SCM model

Figure 6. MFs obtained by ANFIS-FCM model

3.4. Estimation of required rotational torque using SVR-DE model

In this paper, a hybrid SVR with DE was proposed to predict the required rotational torque, using MATLAB environment. A dataset that includes 84 data points was employed in current study, while 59 data points (70%) were applied for building the model and the remainder data points (25 data points) were used for model performance evaluation.

Furthermore, as shown in section 2.2.1, the generalization ability of SVR is highly dependent upon its learning parameters, i.e., $\{C, \sigma, \varepsilon\}$. Consequently, the DE was used to manipulate these parameters and to form hybrid SVR– DE. 10-fold cross-validation performance measure was applied to training dataset along with SVR– DE to achieve reliable results. Related to the purpose, parameters regularizations for run of optimization models are presented in Table 5.

Table 5: Regulated parameters for run the DE

Parameter	Value
Maximum number of iterations	100
Population numbers (N)	50
Mutation factor (F)	0.9
Crossover rate (R)	0.2
Upper and lower bound	[0 1]
Fitness	Root Mean squared error

The adjusted parameters $\{C, \sigma, \varepsilon\}$ with maximal accuracy are selected as the most appropriate parameters. Then, the optimal parameters are used to train the SVR model. The best parameters which obtain by models are presented in Table 6.

Table 6: Optimal values of the SVR parameters which obtain from used DE algorithm

	Optimal value of σ parameter	Optimal value of C parameter	Optimal value of ε parameter
SVR-DE model	3.3529	3714.45	0.2029

4. MODELS PERFORMANCE EVALUATION

4.1 Performance criteria

To verify the performance of the models, two statistical criteria viz. mean squared error (MSE) and squared correlation coefficient (R^2) were chosen to be the measure of accuracy. Let t_k be the actual value and $\hat{t}_k$ be the predicted value of the kth observation and n be the number of observations, then MSE and R^2 could be defined, respectively, as follows:

$$MSE = \frac{1}{n}\sum_{k=1}^{n}(t_k - \hat{t}_k)^2 \tag{21}$$

$$R^2 = 1 - \frac{\sum_{k=1}^{n}(t_k - \hat{t}_k)^2}{\sum_{k=1}^{n}t_k^2 - \frac{\sum_{i=1}^{n}\hat{t}_k^2}{n}} \tag{22}$$

4.2. Results and discussion

In this study an attempt has been made to show the capability of the SVR-DE and ANFIS models to predict the rotational torque. The models of ANFIS and SVR-DE, for the prediction of rotational torque, were constructed using seven inputs. The part of the sensitivity analysis of ANFIS–SCM and ANFIS–FCM models are shown in Table 7 and Table 8. Furthermore, a comparison between the results of the ANFIS–GP, ANFIS–SCM, ANFIS–FCM and SVR–DE models for testing and training datasets is shown in Table 9. As it can be observed from this table, in the prediction of rotational torque using the ANFIS–SCM model, R^2 values of 0.85 and 0.99 for the training and testing suggest the superiority of this model in predicting the rotational torque to operate horizontal directional drilling. Also, it is found that the ANFIS–FCM is best method in the second order. Furthermore, a correlation between estimated values of rotational torque by ANFIS–GP, ANFIS–SCM, ANFIS–FCM and SVR–DE models and measured values for 84 data sets at training and testing phases is shown in Figs. 7 and 8.

Table 7: Part of the sensitivity analysis of the ANFIS-SCM model

Influence radius	The number of periodic training process	R^2_{Train}	MSE_{Train}	R^2_{Test}	MSE_{Test}
0.9	1000	0.9904	0.0006	0.6643	0.0452
1.6	5000	0.8371	0.0091	0.7287	0.0128
1.42	5000	0.8913	0.0061	0.7983	0.0072
0.6	1000	0.9977	0.0001	0.8008	0.0064
1.5	3000	0.8671	0.0074	0.8039	0.0076
1.52	100	0.8756	0.0070	0.8242	0.0073
1.2	100	0.9392	0.0034	0.8304	0.0070
1.1	100	0.9643	0.0020	0.8387	0.0054
1.39	100	0.9455	0.0030	0.8494	0.0060
0.55	100	0.9982	0.0001	0.8517	0.0065

Table 8: Part of the sensitivity analysis of the ANFIS-FCM model

Number of clusters	The number of periodic training process	R^2_{Train}	MSE_{Train}	R^2_{Test}	MSE_{Test}
2	1000	0.8013	0.0111	0.6920	0.0139
6	500	0.8247	0.0099	0.6925	0.0151
10	100	0.8598	0.0080	0.7406	0.0132
20	200	0.8710	0.0072	0.7509	0.0145
8	1000	0.8686	0.0075	0.7653	0.0126
22	100	0.8832	0.0065	0.7697	0.0182
30	1000	0.8904	0.0061	0.7667	0.0169
18	500	0.8840	0.0065	0.7922	0.0152
25	100	0.9013	0.0055	0.8008	0.0134
40	300	0.9386	0.0034	0.8066	0.0099

$R^2 = 0.8658$

(a)

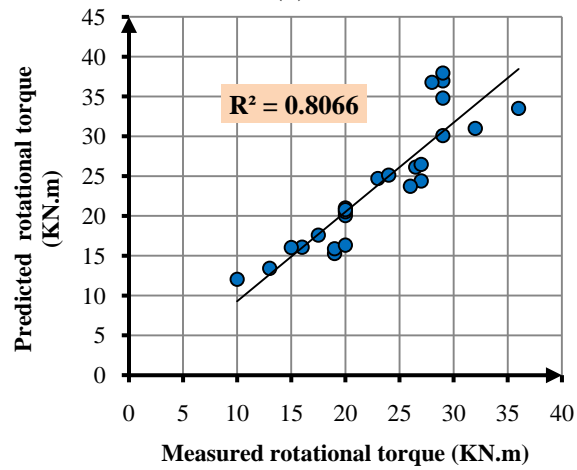$R^2 = 0.9982$
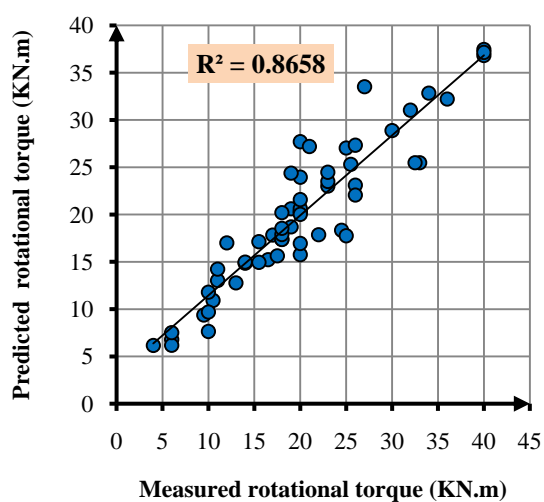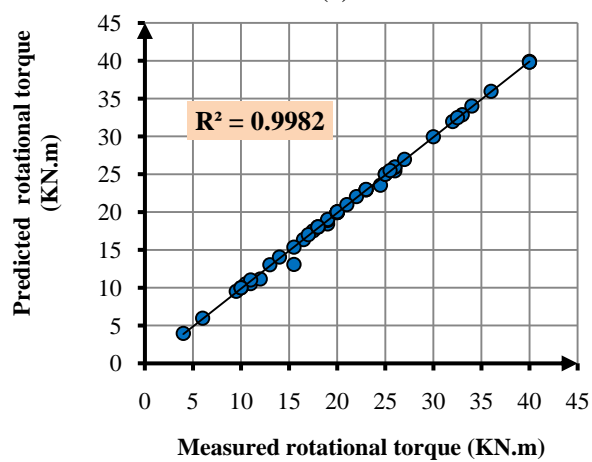
(b)
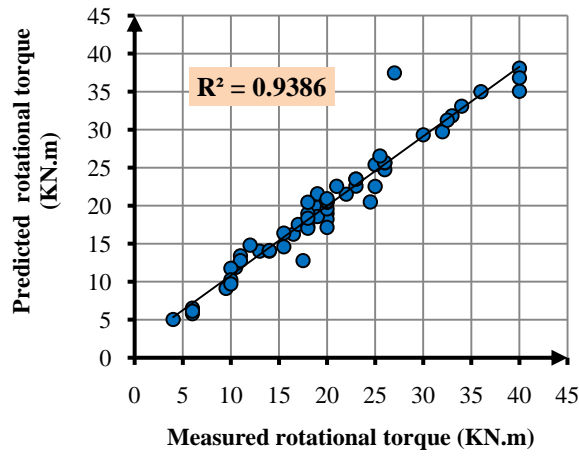
(c)
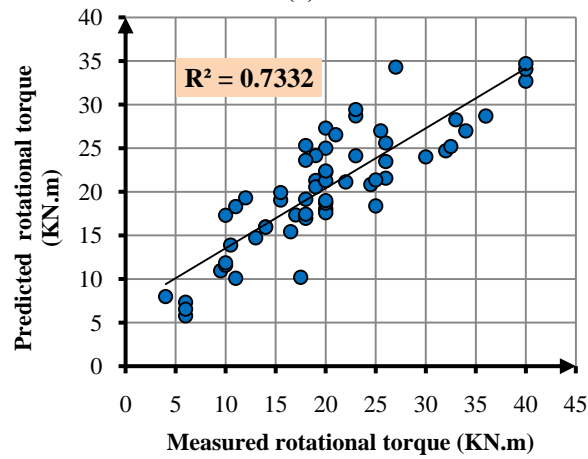
(d)

Figure 7. Correlation between measured and estimated rotational torque for training datasets, (a) ANFIS-GP, (b) ANFIS-SCM, (c) ANFIS–FCM, (d) SVR-DE
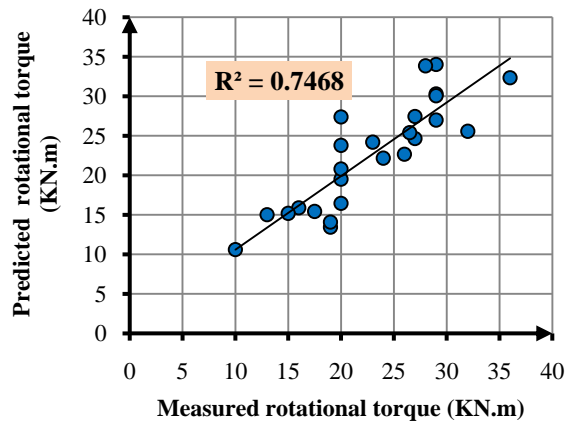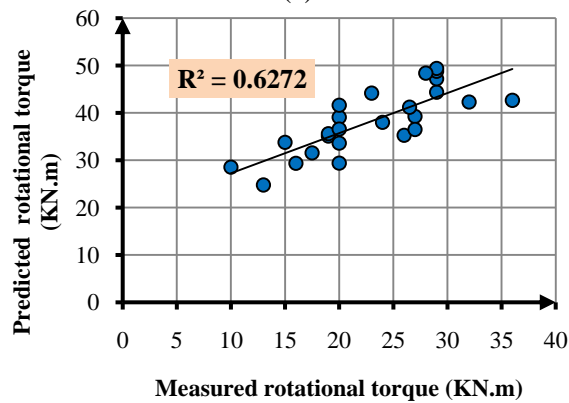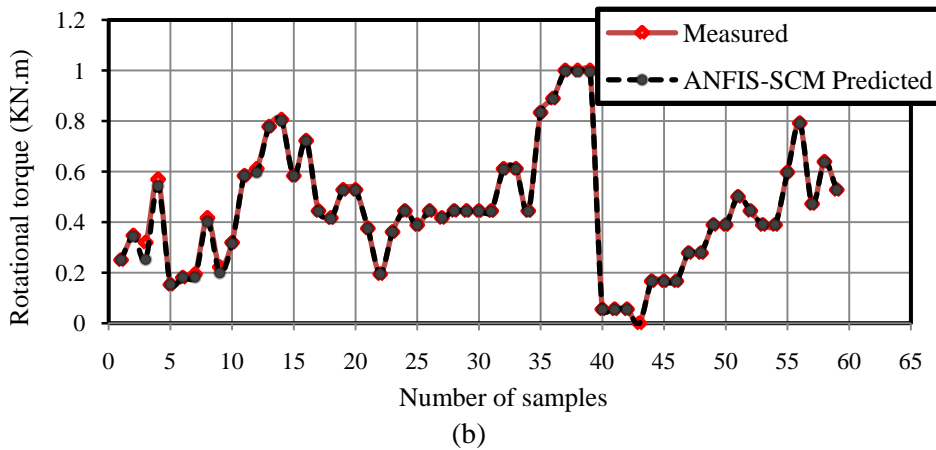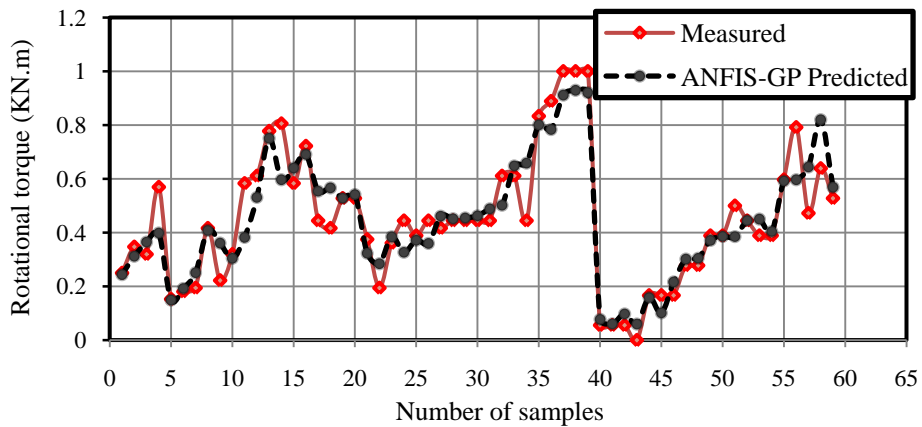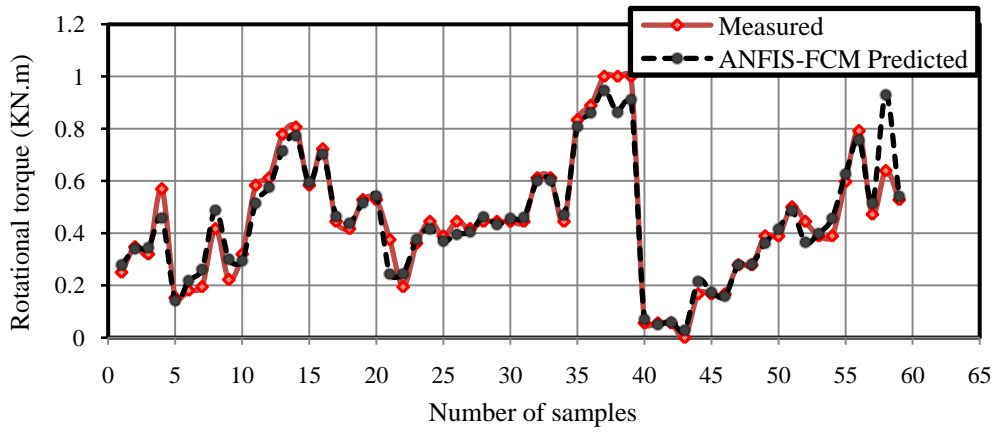

(a)
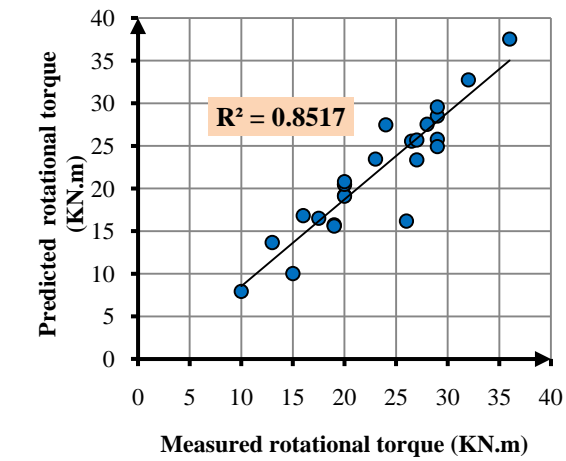
(b)

(c)

(d)

Figure 8. Correlation between measured and estimated rotational torque for testing datasets, a) ANFIS-GP, b) ANFIS-SCM, c) ANFIS–FCM, d) SVR-DE

Table 9: A comparison between the results of the ANFIS–GP, ANFIS–SCM, ANFIS–FCM and SVR-DE models

Model		R^2	MSE
ANFIS-GP	Training datasets	0.7468	0.0089
	Testing datasets	0.8658	0.0075
ANFIS–SCM	Training datasets	0.8517	0.0065
	Testing datasets	0.9982	0.0001
ANFIS–FCM	Training datasets	0.8066	0.0099
	Testing datasets	0.9386	0.0034
SVR-DE	Training datasets	0.7332	0.01522
	Testing datasets	0.6272	0.1929

Also, a comparison between estimated values of rotational torque by ANFIS–GP, ANFIS–SCM, ANFIS–FCM and SVR–DE models and measured values for 84 data sets at training and testing phases is shown in Figs. 9 and 10.
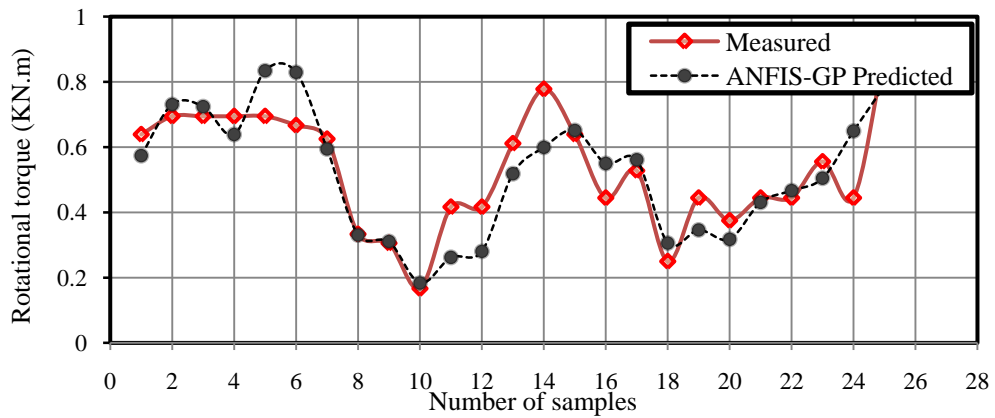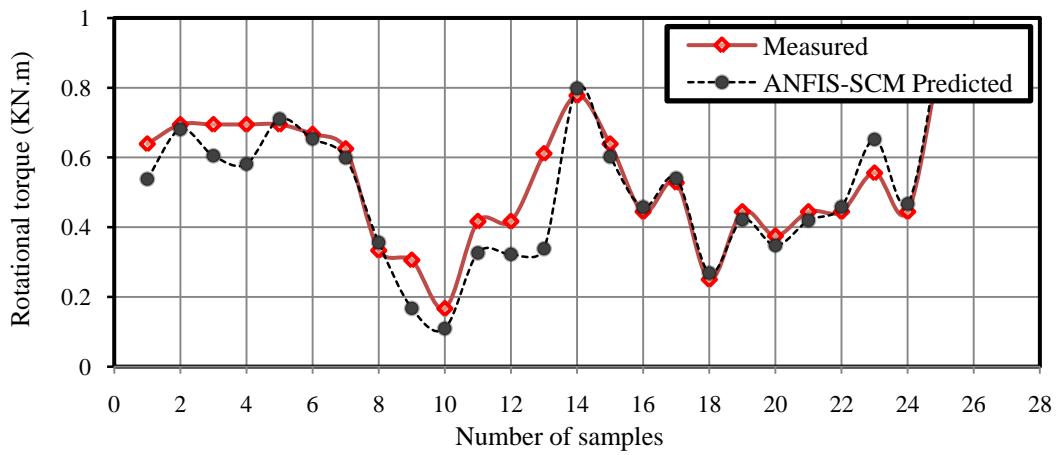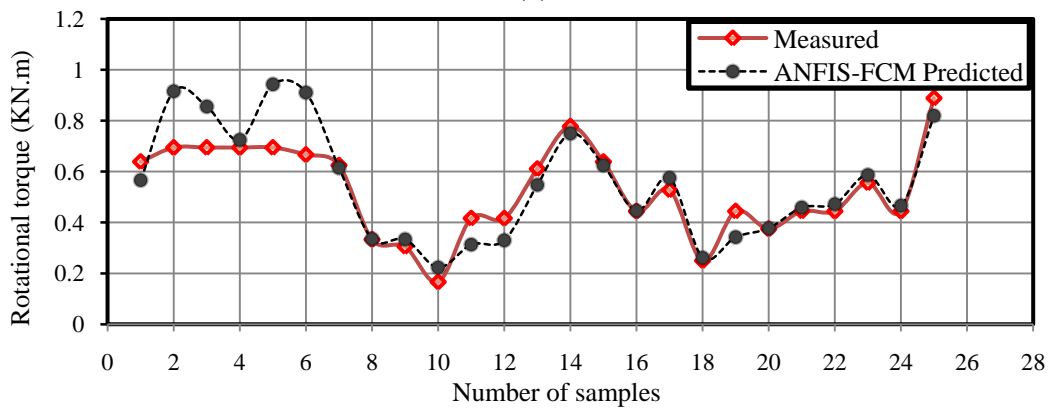

(a)

(b)
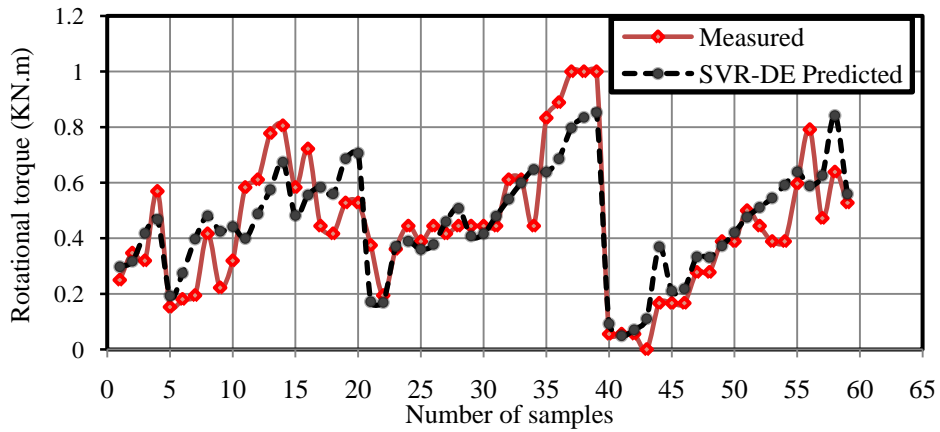
Figure 9. Comparison between measured and estimated rotational torque for training datasets, (a) ANFIS-GP, (b) ANFIS-SCM, (c) ANFIS–FCM, (d) SVR-DE
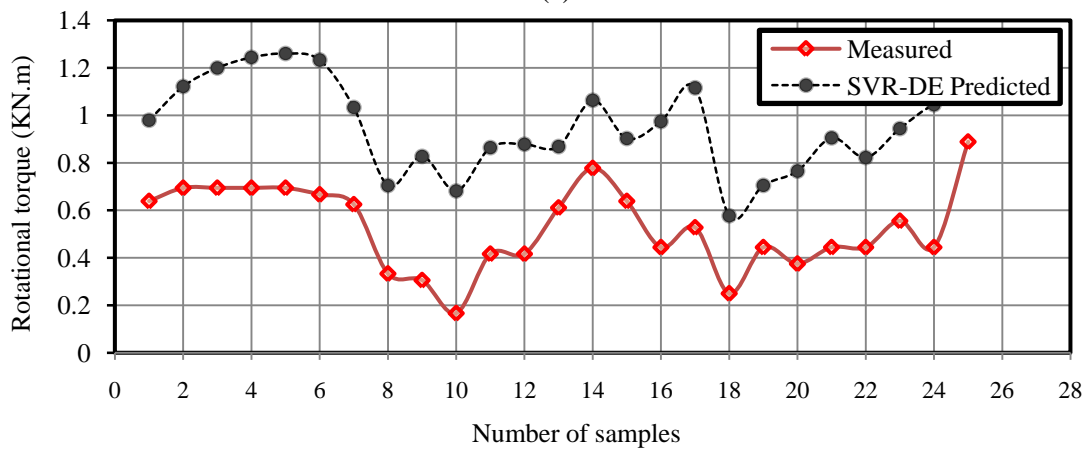
Figure 10. Comparison between measured and estimated rotational torque for testing datasets,
(a) ANFIS-GP, (b) ANFIS-SCM, (c) ANFIS–FCM, (d) SVR-DE

5. CONCLUSIONS

In this paper, the application of artificial intelligence methods for data analysis named ANFIS–FCM, ANFIS–GP, ANFIS–SCM and SVR–DE to estimate of the required rotational torque to operate horizontal directional drilling is demonstrated. The following remarks were concluded:

- In this paper, a new approach namely support vector regression optimized by DE is proposed for predicting the required rotational torque to operate horizontal directional drilling. In our methodology, DE is applied as optimization tool for determining the optimal value of user defined parameters existing in formulation of SVR. The optimization implementation increases the performance of SVR model.

- A comparison was made between SVR-DE model and ANFIS models (ANFIS–GP, ANFIS–SCM, ANFIS–FCM), using 84 data samples, and based upon the performance indices; MSE and R^2, ANFIS-SCM was selected as the best predictive model. Also, it is found that the ANFIS–FCM is best method in the second order.

- Consequently, it may conclude that ANFIS-SCM is a reliable system modeling technique for estimating required rotational torque to operate horizontal directional drilling with highly acceptable degree of accuracy and robustness.

This study shows that the SVR-DE and ANFIS models (ANFIS–FCM, ANFIS–GP, ANFIS–SCM) approaches can be used as a powerful tool for modeling of some problems involved in geothechnical engineering.

REFERENCES

1. Gokhale S, Hamm R, Sterling R. A comprehensive survey on the state of horizontal directional drilling in North America provides an inside look at this increasingly growing industry, *Direct Drill* 1999; **7**: 20-3.
2. Lan H, Ma B, Shu B, Wu Z. Prediction of rotational torque and design of reaming program using horizontal directional drilling in rock strata, *Tunn Undergr Sp Tech* 2011; **26**: 415-21.
3. Allouche EN. Implementing quality control in HDD projects-a North American prospective, *Tunn Undergr Sp Tech* 2001; **16**: 3-12.
4. Maidla EE, Wojtanowicz AK. A field method for assessing borehole friction for directional well casing, *J Pet Sci Eng* 1988; **1**: 323-33.
5. Niżnik D, Gonet A. Identification of rotational torque and power in HDD, *Arch Min Sci* 2007; **52**: 49-60.
6. Madandoust R, Bungey JH, Ghavidel R. Prediction of the concrete compressive strength by means of core testing using GMDH-type neural network and ANFIS models, *Comp Mater Sci* 2012; **51**: 261-72.
7. Feili Monfared A, Ranjbar M, Nezamabadi-Poor H, Schaffie M, Ashena R. Development of a neural fuzzy system for advanced prediction of bottomhole

circulating pressure in underbalanced drilling operations, *Petrol Sci Technol* 2011; **29**: 2282-92.

8. Rad HN, Jalali Z, Jalalifar H. Prediction of rock mass rating system based on continuous functions using Chaos-ANFIS model, *Int J Rock Mech Min Sci* 2015; **73**: 1-9.

9. Üstün B, Melssen W, Oudenhuijzen M, Buydens L. Determination of optimal support vector regression parameters by genetic algorithms and simplex optimization, *Anal Chim Acta* 2005; **544**: 292-305.

10. Kaveh A, Talatahari S. Hybrid charged system search and particle swarm optimization for engineering design problems, *Eng Computation* 2011; **28**: 423-40.

11. Kaveh A, Khayatazad M. A new meta-heuristic method: ray optimization, *Comput Struct* 2012; **112**: 283-94.

12. Kaveh A, Zolghadr A. Democratic PSO for truss layout and size optimization with frequency constraints, *Comput Struct* 2014; **130**: 10-21.

13. Kaveh A, Mahdavi V. Colliding bodies optimization: a novel meta-heuristic method. *Comput Struct* 2014; **139**: 18-27.

14. Jang JSR. ANFIS: adaptive-network-based fuzzy inference system, *IEEE T Syst Man Cyb* 1993; **23**: 665-85.

15. Wu JD, Hsu CC, Wu GZ. Fault gear identification and classification using discrete wavelet transform and adaptive neuro-fuzzy inference, *Expert Syst Appl* 2009; **36**: 6244-55.

16. Wei M, Bai B, Sung AH, Liu Q, Wang J. ME Cather: Predicting injection profiles using ANFIS, *Inform Sciences* 2007; **177**: 4445-61.

17. Chiu SL. Fuzzy model identification based on cluster estimation, *J Intell Fuzzy Syst* 1994; **2**: 267-78.

18. Yager RR, Filev DP. Generation of fuzzy rules by mountain clustering, *J Intell Fuzzy Syst* 1994; **2**: 209-19.

19. Fattahi H, Shojaee S, Farsangi MAE, Mansouri H. Hybrid Monte Carlo simulation and ANFIS-subtractive clustering method for reliability analysis of the excavation damaged zone in underground spaces, *Comput Geotech* 2013; **54**: 210-21.

20. Fattahi H. Indirect estimation of deformation modulus of an in situ rock mass: an ANFIS model based on grid partitioning, fuzzy c-means clustering and subtractive clustering, *J Geosci* 2016; 1-10.

21. Bayatzadeh Fard Z, Ghadimi F, Fattahi H. The use of artificial intelligence techniques to predict the distribution of heavy metals in groundwater at the Lakan lead-zinc mine in Iran, *J Min Environ* 2016; in press.

22. Bezdek JC. *Fuzzy Mathematics in Pattern Classification*, Cornell university, Ithaca, 1973.

23. Vapnik V. The nature of statistical learning theory, Springer, 1999.

24. Vapnik V, Golowich SE, Smola A. Support vector method for function approximation, regression estimation, and signal processing, *Adv Neur In* 1997; 281-87.

25. Smola AJ, Schölkopf B. A tutorial on support vector regression, *Statist Comput* 2004; **14**: 199-222.

26. Geem ZW. *Music-Inspired Harmony Search Algorithm: Theory and Applications*, Springer Verlag 2009.
27. Storn R, Price K. Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces, ICSI Berkeley 1995.
28. Price K, Storn RM, Lampinen JA. Differential evolution: a practical approach to global optimization, Springer Science & Business Media, 2006.
29. Wang FK, Du T. Implementing support vector regression with differential evolution to forecast motherboard shipments, *Expert Syst Appl* 2014; **41**: 3850-55.
30. Ardia D, Boudt K, Carl P, Mullen KM, Peterson BG. Differential evolution with deoptim, *R Journal* 2011; **3**: 27-34.
31. Mullen KM, Ardia D, Gil DL, Windover D, Cline J. DEoptim: An R package for global optimization by differential evolution, *J Statist Soft* 2011; **40**: 1-26.