A COMPARATIVE INVESTIGATION USING ARTIFICIAL NEURAL NETWORK (ANN) AND DECISION TREE (DT) METHODS IN THE PREDICTION OF SLUMP AND STRENGTH FOR CONCRETE SAMPLES

VAN TUAN VU[a*]
[a]Le Quy Don Technical University
[*]Corresponding author: *Email: vutuan2601@yahoo.com; Cell phone: 0961917618*
Article history: Received 17/2/2023, Revised 17/3/2023, Accepted 21/3/2023
https://doi.org/10.59382/j-ibst.2023.en.vol1-3

Abstract: *In the past few years, the application of Machine Learning Techniques (MLT) has become a popular way to enhance the accuracy of predicting concrete properties. This study aims to compare and contrast the performance of Artificial neural network (ANN) and Decision Tree (DT) methods in predicting the compressive strength and slump values of concrete samples. Experimental data used for model building and comparison were obtained from a previous research project. R-squared value (RSQ) and Mean Squared Error (MSE) metrics were used to determine which regression method was the most efficient in predicting concrete compressive strength and slump values. The results from the comparison between ANN and DT methods would be able to identify which of the two regression models is the better choice for forecasting concrete properties.*

Keywords: *Prediction, concrete samples; slump; specific strength; artificial neural network (ANN), decision tree (DT).*

1. Introduction

The complex composite material of concrete makes it highly difficult to accurately model its properties.. The large number of variables that can affect the response variables makes it difficult to do experimental designs. As the amount of effect variables increase, so does the amount of trials required. This, along with the unpredictable nature of concrete, makes getting the real response function a challenging task.

Machine Learning Techniques (MLT) is a field with many disciplines and techniques that can be used to obtain fresh data. It's primarily used to produce predictions. The forecasting of categorical variable values is termed classification, while predicting numerical variable values is called regression. Regression is the process that examines the connection between a dependent variable and one or more independent variables [15].

Recent years have seen a surge in the use of MLT methods to improve the accuracy of predictions made about concrete properties [8], and a variety of engineering applications have seen benefits from their use [11, 20]. Scientists have taken advantage of the data generated from literature sources to bolster their accuracy of predicting concrete properties[6, 12, 19] , and Chopra et al. [10] even applied the data generated under controlled laboratory conditions in their studies. MLT methods, while proven effective, remain underutilized and there is much potential for further research and applications.

Regression models are frequently employed to predict the compressive strength of high-strength concrete [14, 24], and how it is affected by the mixing ratios [18]. To further improve the accuracy of the forecast, Topçu and Sarıdemir [22] and Başyiğit et al. [7] built models using both neural network (NN) and fuzzy logic (FL) approaches. Both studies concluded that the compressive strength could be predicted using the developed NN and FL models without any additional experiments. NN performed better than other data mining methods, ultimately increasing the precision of the concrete compressive strength predictions [9, 10, 16, 23] Khademi et al. [16] compared the multiple linear regression, neural network, and adaptive neuro-fuzzy inference system (ANFIS) methods and found that the NN and ANFIS models yielded more accurate results for the 28-day compressive strength of concrete.

In this research, the compressive strength and slump values of the concrete samples were forecasted using decision tree (DT) and artificial neural networks (ANN) methods, which were compared against the published data collected from a previous study [13]. The R, RMSE, and MAE metrics were used to measure the prediction accuracy of the developed models and to identify the most effective regression method.

2. Data division and preprocessing

The data set consists of samples generated by altering the factors of factorial experiments. These

factors include the ratio of water to adhesive (coded variable x_1), the ratio of blast furnace slag to adhesive (coded variable x_2), the ratio of silica fume to adhesive (coded variable x_3), and the ratio of super-plastic additive to adhesive (coded variable x_4). Thus, the coded variables x_1, x_2, x_3, and x_4 were chosen as input variables, with slump and specific strength chosen as the output variables.

Box Hunter's method of statistics for experimenters was employed to establish the law of altering factors. The materials used in the experiment include PCB40 Nghi Son cement, blast furnace slag obtained as a by-product of cast iron production from Thai Nguyen Iron and Steel Joint Stock Corporation, Elkem's grey silica fume, and Mappei's Dynamon SP1 super-plastic additive. Further information regarding the experiment can be found in the document [13].

To reduce the size of the variables and to ensure that all variables are given the same attention during the training period, we conduct preprocessing by scaling the input and output variables between 0.0 and 1.0. The scaled value for each variable x, x_n, is calculated like this:

$$x_n = x / x_{max} \quad (1)$$

Where: x_{max} is maximum values of each variable x.

Table 1. *Training database of concrete samples [13]*

No	Coded variables				Factorial experiments				Slump	Specific strength
	x1	x2	x3	x4	W/A	BFS/A	SF/A	SPA/A	(cm)	(Mpa)
1	-1	-1	-1	-1	0.3	30	3	0.6	9	88.7
2	1	-1	-1	-1	0.34	30	3	0.6	16.5	78.0
3	-1	1	-1	-1	0.3	50	3	0.6	12	85.9
4	1	1	-1	-1	0.34	50	3	0.6	19.5	77.1
5	-1	-1	1	-1	0.3	30	9	0.6	0.5	89.5
6	1	-1	1	-1	0.34	30	9	0.6	4	81.8
7	-1	1	1	-1	0	50	9	0.6	2	85.8
8	1	1	1	-1	0.34	50	9	0.6	7	81.2
9	-1	-1	-1	1	0.3	30	3	1	19	89.0
10	1	-1	-1	1	0.34	30	3	1	21	77.6
11	-1	1	-1	1	0.3	50	3	1	20.5	86.1
12	1	1	-1	1	0.34	50	3	1	22	77.2
13	-1	-1	1	1	0.3	30	9	1	11	89.8
14	1	-1	1	1	0.34	30	9	1	17.5	81.9
15	-1	1	1	1	0.3	50	9	1	17	86.5
16	1	1	1	1	0.34	50	9	1	21	81.5
17	-2	0	0	0	0.28	40	6	0.8	11	94.7
18	2	0	0	0	0.36	40	6	0.8	21	77.4
19	0	-2	0	0	0.32	20	6	0.8	15.5	84.1
20	0	2	0	0	0.32	60	6	0.8	19.5	79.8
21	0	0	-2	0	0.32	40	0	0.8	21	80.1
22	0	0	2	0	0.32	40	12	0.8	4	84.7
23	0	0	0	-2	0.32	40	6	0.4	2.5	82.7
24	0	0	0	2	0.32	40	6	1.2	19.5	82.8
25	0	0	0	0	0.32	40	6	0.8	18	84.7
26	0	0	0	0	0.32	40	6	0.8	17.5	84.1
27	0	0	0	0	0.32	40	6	0.8	18.5	83.2
28	0	0	0	0	0.32	40	6	0.8	17	84.2
29	0	0	0	0	0.32	40	6	0.8	16.5	83.3
30	0	0	0	0	0.32	40	6	0.8	17.5	84.2
31	0	0	0	0	0.32	40	6	0.8	18.5	82.1

* In which W/A is water/adhesive ratio; BFS/A is blast furnace slag/adhesive ratio; SF/A is silicafume/adhesive ratio; SPA/A is super-plastic additive/adhesive ratio.

Table 2. *Testing - database of concrete samples [13]*

No	Coded variables				Factorial experiments				Slump	Specific strength
	x1	x2	x3	x4	W/A	BFS/A	SF/A	SPA/A	(cm)	(Mpa)
1	-2	1.675	-0.765	1	0.28	56.75	3.705	1	18.5	92.2
2	-1	0.551	-1.121	0	0.3	45.51	2.637	0.8	17.5	88.5
3	0	1.409	-1.548	-1	0.32	54.09	1.356	0.6	18.5	75.4
4	1	0.678	-0.863	-1	0.34	46.78	3.411	0.6	18.5	72.6
5	2	1.034	-0.39	-1	0.36	50.34	4.83	0.6	18	70.3

3. Overview of artificial neural networks and decision trees

3.1 Artificial neural networks

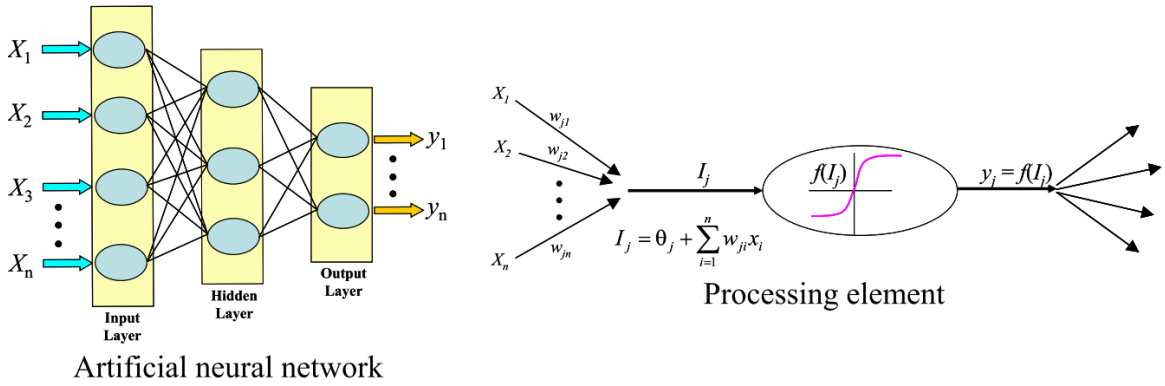

Figure 1. *Structure and operation of an ANN [5]*

Artificial neural networks (ANNs) are numerical modeling approaches inspired by the functioning of the human brain and nervous system [5]. Their purpose is similar to conventional statistical models, which is to figure out the link between the given inputs and corresponding outputs. However, ANNs don't rely on set mathematical equations to do this, and this enables them to surpass the constraints of standard models. For this research project, a Multi-layer feed-forward with the back-propagation algorithm training [4] was utilized. The multi-layer feed-forward neural network consists of several processing elements (called nodes or neurons) that are fully or partially linked by connection weights. These elements are generally classified into several different layers: an input layer; an output layer; and hidden layers (layers in between).

A number of authors have already documented the architecture and working of Artificial Neural Networks (ANNs). M.A Shahin [5] has produced a representation of the structure and operation of an ANN as shown in Figure 1. Every processing element receives inputs from the preceding layer (x_i) and these inputs are multiplied by the adjustable connection weights (w_{ji}). The weighted inputs are then totaled up, along with a bias (θ_j) that is either added or subtracted. This combined input (I_j) is then put through a non-linear transfer function (f(.)), like the sigmoidal function or the tanh function, to produce the output of the processing element (y_j).

The multi-layer feed-forward neural network begins at the input layer, followed by the application of a learning rule to receive the output from the network (as seen in Figure 1). Weights and bias are adjusted in order to reduce the amount of error that is found between the output desired and the output obtained from the previous step. Once the training phase is finished, the trained model has to be validated by an independent testing set. Maier & Dandy [3] have discussed the various steps needed to create an ANN.

In order to achieve the best possible performance of an Artificial Neural Network, there is no particular set of rules or regulations to follow. Consequently, numerous ANN configurations have been tested and experimented with.

According to Hornik [2], a single hidden layer network can approximate any continuous function with sufficient connection weights. Therefore, this ANN model utilizes only one hidden layer. The ReLU and tanh transfer functions are selected for the hidden and output layers, respectively. The training process is terminated after 5000 training cycles (epochs), which aligns with the method previously published by Shahin [5]. This number is sufficient as no significant improvement in error occurs and the training loss does not fluctuate or increase at the end of the process (refer to Figure 2).

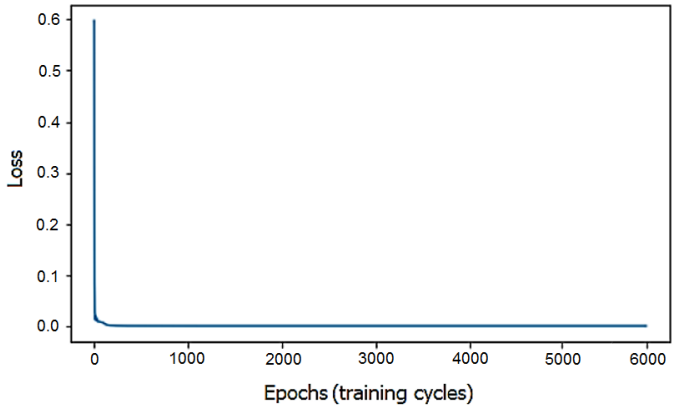

Figure 2. *Variation of Loss against Epoch*

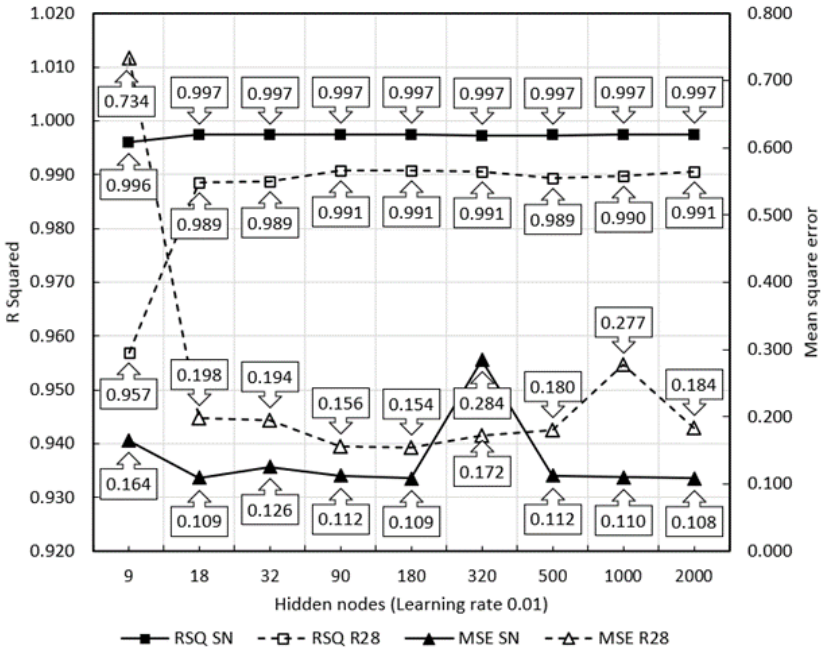

Figure 3. *Influence of Number of Hidden Nodes on the Performance of ANN Model*

Although Caudill [1] recommended that a network with I inputs should have 2I+1 hidden layer nodes to map any continuous function, the effect of hidden nodes on ANN model performance (refer to Figure 3) reveals that the ANN model with 180 hidden nodes has the lowest prediction error (highest R squared and lowest Mean square error). This number of hidden nodes exceeds the recommendation and previous usage of authors [1].

Figure illustrates the impact of learning rate on the performance of ANN models. It is evident that the ANN model with a learning rate lower than 0.01 has the lowest prediction error, and larger learning rates

result in increased prediction errors. The Adam gradient descent optimization algorithm is utilized, which incorporates momentum; hence, the momentum term is not examined.

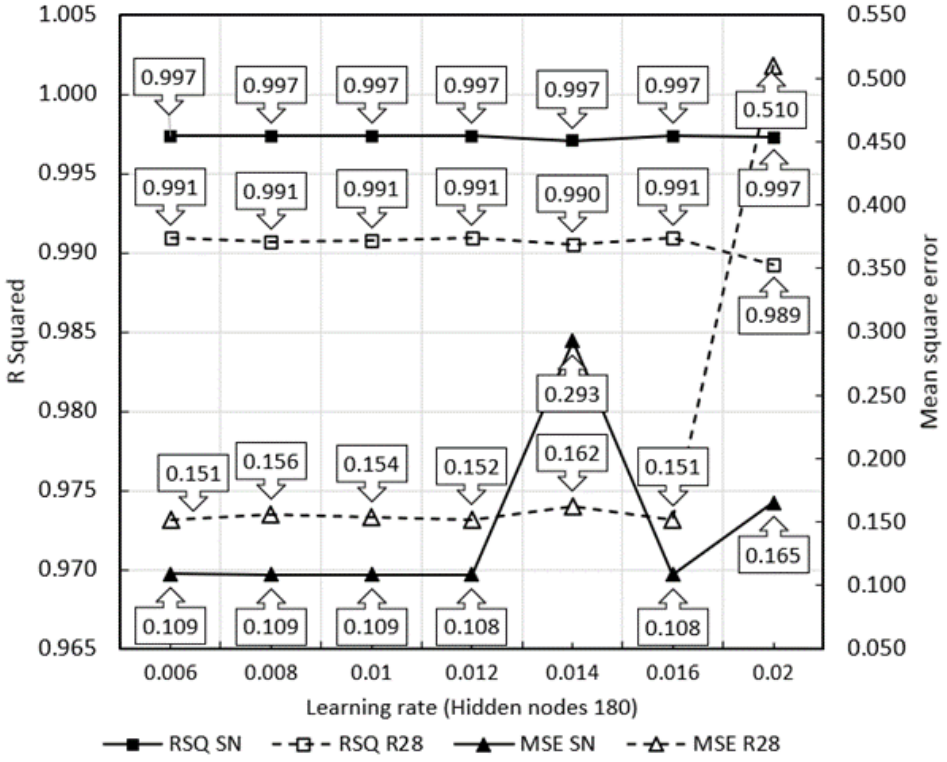

Figure 4. *Influence of Learning rate on the Performance of ANN Model*

After careful consideration, it was concluded that the most optimal model for predicting the compressive strength and slump values of the concrete samples was the one with a hidden layer, learning rate of 0.01, comprising 180 hidden nodes, and a training period of 5000 iterations (epochs). The Scikit-learn library was used to build the ANN and DT models.

3.2 Decision trees

The DT (Decision Tree) model is a popular approach employed in literature due to its ability to accurately model higher order nonlinearity and offer greater interpretability [21, 25]. In the form of a tree structure, the DT splits data into subsets composed of three distinct nodes: the Root Node (RN), the Decision Node (DN), and the Leaf Node (LN). At the topmost node, the RN, a conditional test is performed. Depending on the outcome of the test, further subtrees are created. The LN is the terminal node (output) of the tree. Entropies (which measure the homogeneity of the dataset)[17] are calculated by DT algorithms in two ways.

Entropy using the frequency table of one attribute

$$E(D) = \sum_{k=1}^{c} -p_k log_2 p_k \quad (2)$$

where p_k is the proportion of D belonging to class k.

Entropy using the frequency table of two attributes

$$E(T,X) = \sum_{c \in X} P(c)E(c) \quad (3)$$

where T is the target attribute, X is the decision attribute, c is the tuple values of attribute X, P(c) is the probability of occurrence of c, and E(c) is the entropy of c.

Next step is to calculate the decrease in the entropy or information gain.

$$Gain(T,X) = E(T) - E(T,X) \quad (4)$$

where T is the target attribute, X is the decision attribute, c is the tuple values of attribute X, E(T) is the calculated entropy of the target attribute, and E(T,X) is the entropy of X tuples in T attribute.

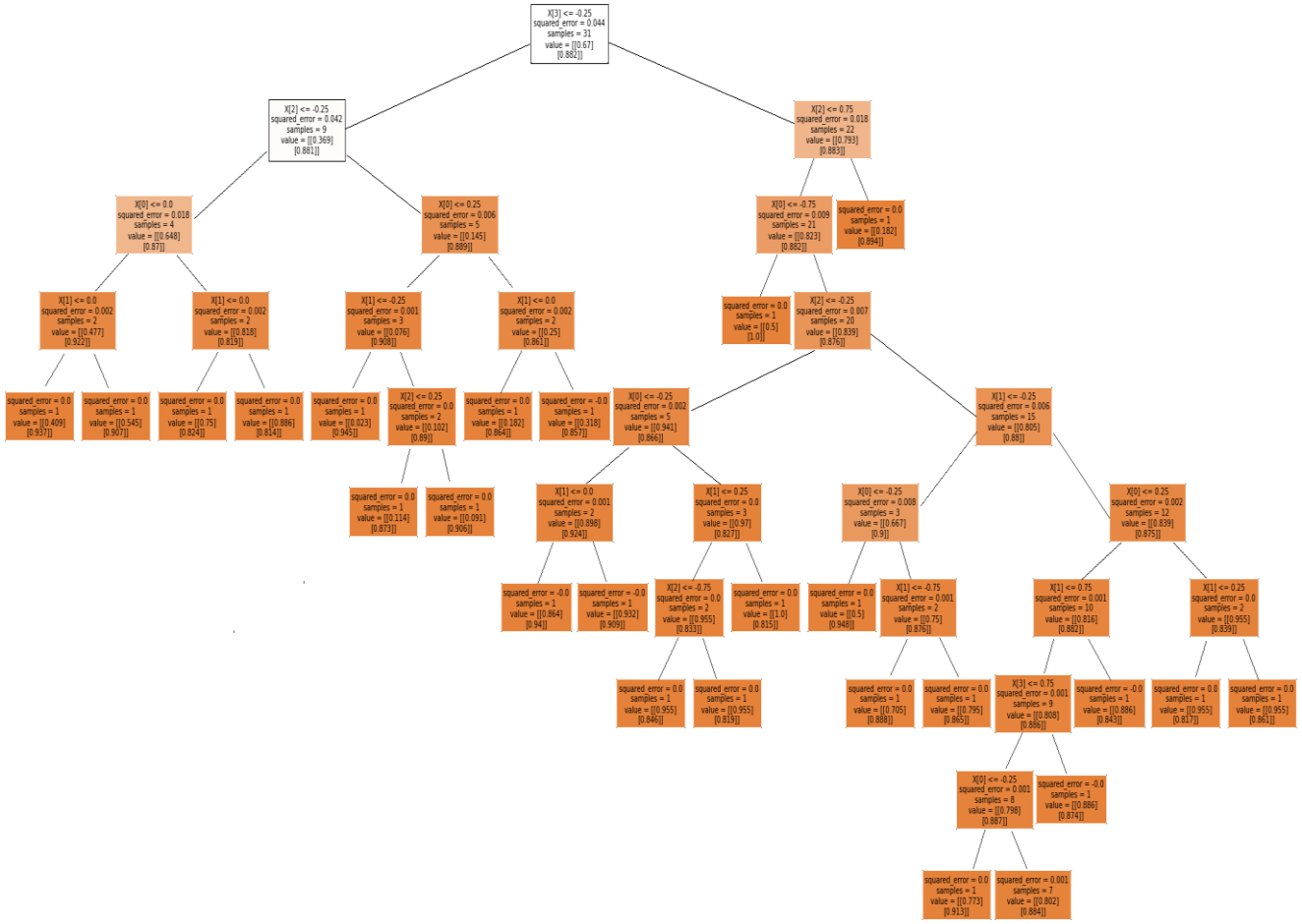

Figure 5. *A summary of the structure of the decision tree model*

The information gain of the individual branches of the dataset is assessed; the attribute with the greatest information gain is then selected as the decision node while branches with a gain of 0 are labeled as leaf nodes. The algorithm is then recursively performed on the non-leaf branches until all of the data are classified.

In the context of regression problems, the mean squared error (MSE) around the mean response of the node is widely used as a measure of node purity. The criterion for selecting the splitting variable and the segmentation point of each node is based on the maximum gain in the mean squared error (MSE), which is determined as follows:

$$\Delta MSE\left(S, x_j^{\alpha}\right) = MSE(S) - \frac{|S_1|}{|S|} MSE(S_1) - \frac{|S_2|}{|S|} MSE(S_2) \tag{5}$$

$$MSE(S) = \frac{1}{|S|} \sum_{i=1}^{|S|} (y_i - \hat{y})^2 \tag{6}$$

$$\hat{y} = \frac{1}{|S|} \sum_{i=1}^{|S|} y_i \tag{7}$$

where $|S|$ is the number of samples in dataset S that reach the node; S_i is the dataset resulting from splitting at the node, which falls into a subspace according to the given variable x_j (j = 1, 2, …M) and segmentation α; and y_i is the response value of the i^{th} sample in dataset S.

The process of partitioning continues until the maximum MSE gain is achieved. After constructing the tree, any sample's response can be predicted by tracing the path to the corresponding leaf node and computing the average of the responses in that node. Figure 5 summarizes the decision tree (regression) model's structure that was used in this paper.

4. Results and discussion

Table 3. *Results of training and testing of the DT model and ANN model for compressive strength of concrete*

Model	Slump				Compressive strength			
	Results of training set		Results of testing set		Results of training set		Results of testing set	
	RSQ	MSE	RSQ	MSE	RSQ	MSE	RSQ	MSE
ANN	0.99737	0.10857	0.065969	1.854848	0.9908	0.1542	0.793717	30.79767
DT	0.99738	0.10829	0.083643	45.9	0.99103	0.14931	0.708535	52.264

Table 4. *Accuracy of ANN model (testing set)*

No	Slump			Specific strength		
	Measured value (cm)	Predicted value ANN (cm)	Predicted value DT (cm)	Measured value (Mpa)	Predicted value ANN (Mpa)	Predicted value DT (Mpa)
1	18.5	19.77	11.00	92.2	93.25	94.7
2	17.5	17.99	20.50	88.5	88.14	86.10
3	18.5	19.36	12.00	75.4	77.44	85.90
4	18.5	18.22	19.50	72.6	76.04	77.10
5	18	20.57	7.01	70.3	81.99	81.20

Table 3 and 4 present the results of the DT and ANN models for predicting the compressive strength and slump of concrete. Figure 6 and Figure 7 show the graphical representation of the training and testing datasets for the two models, respectively. It can be seen that the ANN model gives more accurate predictions than the DT model, especially when it comes to the test dataset. For the training dataset, the R-squared value (RSQ) of the ANN model is 0.9908 with a Mean Squared Error (MSE) of 0.1542, while the R-squared value (RSQ) of the DT model is 0.99103 with a Mean Squared Error (MSE) of 0.1493. Comparatively, the R-squared value (RSQ) of the ANN model for the testing dataset is 0.7937 with an MSE of 30.7977 and for the DT model is 0.7085 with an MSE of 52.264. This could be interpreted to mean that the ANN model combines the intricacies of many statistical techniques with machine learning techniques and is known as a "black box" due to its mysterious inner workings.

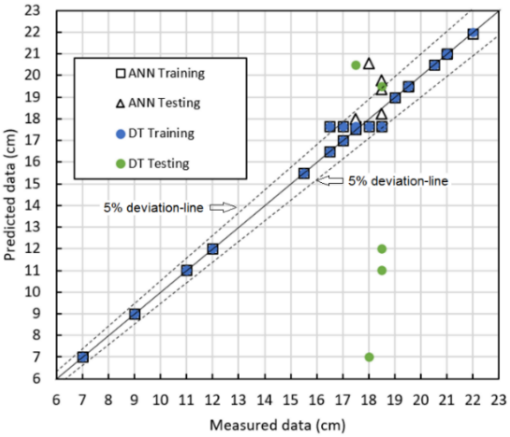

Figure 6. *Comparison between predicted and actual values of slump of concrete using DT and ANN models*

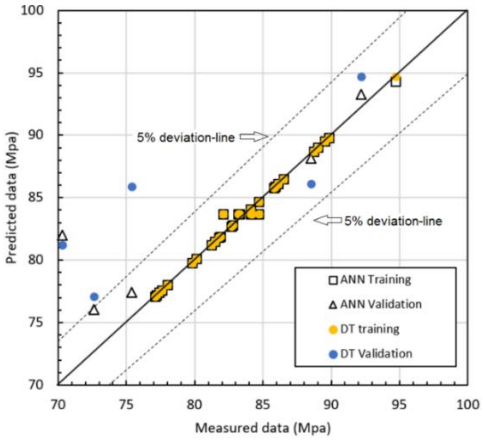

Figure 7. *Comparison between predicted and actual values of compressive strength of concrete using DT and ANN models*

The intricacies of many statistical techniques refer to the complex statistical procedures and methods that are used to analyze data in a variety of ways. These techniques involve complex calculations and algorithms that allow data scientists to identify patterns, relationships, and make predictions based on the data. Some examples of these techniques include regression analysis, ANOVA (it is a statistical method that is used to compare the means of two or more groups of data to determine whether there are significant differences between them), PCA (it is a statistical technique that is used to reduce the dimensionality of large datasets by identifying patterns and correlations within the data), and cluster analysis, among others. On the other hand, machine learning techniques are algorithms and models that enable computers to learn from data and make predictions based on that learning. The ANN model,

in particular, is a machine learning technique that combines the strengths of both statistical and machine learning techniques to develop accurate and robust models for a wide range of applications. While the inner workings of the ANN model may seem mysterious or complex, it is a powerful tool that can provide valuable insights and predictions. This enables the ANN model to be more proficient in predicting the compressive strength and slump of concrete samples when compared to the DT model. However, some of the predicted values of the ANN model are beyond the 5% deviation-line (Figure 6 and Figure 7, Table 4), particularly when it comes to values that are not within the range of the training data. This indicates that, like many empirical models, the ANN model is better suited for interpolation than extrapolation. This is also true for some data points with the same output values but vastly different input values, as the model might be 'confused' if the training data in these points is inadequate (see Figure 6 and Figure 7). To further improve the performance of the ANN model, the training dataset should have a wider range of values.

5. Conclusion

After comparing the ANN and DT methods, the following conclusions can be drawn:

- The ANN model is more reliable for predicting the compressive strength and slump of concrete samples than the DT model. It could be the ANN model combines the intricacies of many statistical techniques with machine learning techniques and is known as a "black box" due to its mysterious inner workings;

- The ANN model is better suited for interpolation than extrapolation, like many empirical models. To further improve the performance of the ANN model, the training dataset should have a wider range of values.

REFERENCES

[1] Caudill Maureen (1988). "Neural networks primer, Part III". *AI Expert*, *3 (6), pp 53-59.*

[2] Hornik Kurt, Stinchcombe Maxwell, and White Halbert (1989). "Multilayer feedforward networks are universal approximators". *Neural networks, 2 (5), pp 359-366.*

[3] Maier HR and Dandy GC (2000), Application of artificial neural networks to forecasting of surface water quality variables: issues, applications and challenges, in Artificial neural networks in hydrology.*Springer. p. 287-309.*

[4] Rumelhart David E, Hinton Geoffrey E and Williams Ronald J (1985), Learning internal representations by error propagation. *California Univ San Diego La Jolla Inst for Cognitive Science.*

[5] Shahin Mohamed A (2010). "Intelligent computing for modeling axial capacity of pile foundations". *Canadian Geotechnical Journal, 47 (2), pp 230-243.*

[6] Akkurt Sedat, Tayfur Gokmen and Can Sever (2004). "Fuzzy logic model for the prediction of cement compressive strength". *Cement and concrete research*, *34 (8), pp 1429-1433.*

[7] Başyigit C, et al. (2010). "Prediction of compressive strength of heavyweight concrete by ANN and FL models". *Neural Computing and Applications, 19 pp 507-513.*

[8] Boukhatem Bakhta, et al. (2011). "Application of new information technology on concrete: an overview". *Journal of Civil Engineering and Management, 17 (2), pp 248-258.*

[9] Chopra Palika, Sharma Rajendra Kumar, and Kumar Maneek (2015). "Artificial neural networks for the prediction of compressive strength of concrete". *International journal of applied science and engineering, 13 (3), pp 187-204.*

[10] Chopra Palika, et al. (2018). "Comparison of machine learning techniques for the prediction of compressive strength of concrete". *Advances in Civil Engineering*.

[11] Cihan Pınar, Gokce Erhan, and Kalipsiz Oya (2017). "A review of machine learning applications in veterinary field". *Kafkas Universitesi Veteriner Fakultesi Dergisi*, 23 (4).

[12] Diab Ahmed M, et al. (2014). "Prediction of concrete compressive strength due to long term sulfate attack using neural network". *Alexandria Engineering Journal, 53 (3), pp 627-642.*

[13] Đinh Quang Trung (2010). "Nghiên cứu sử dụng phụ gia khoáng hỗn hợp từ xỉ lò cao và silicafume chế tạo bê tông cường độ cao". *Tạp chí Khoa học và Kỹ thuật - Học viện KTQS, 135 (7-2010), pp 233-241.*

[14] FM Zain M and M Abd S (2009). "Multiple regression model for compressive strength prediction of high performance concrete". *Journal of applied sciences*, *9 (1), pp 155-160.*

[15] Hofmann Markus and Klinkenberg Ralf (2016), RapidMiner: Data mining use cases and business analytics applications: *CRC Press.*

[16] Khashman Adnan and Akpinar Pinar (2017). "Non-destructive prediction of concrete compressive strength using neural networks". *Procedia Computer Science, 108 pp 2358-2362.*

[17] Kingsford Carl and Salzberg Steven L (2008). "What are decision trees?". *Nature biotechnology, 26 (9), pp 1011-1013.*

[18] Namyong Jee, Sangchun Yoon and Hongbum Cho (2004). "Prediction of compressive strength of in-situ concrete based on mixture proportions". *Journal of Asian Architecture and Building Engineering, 3 (1), pp 9-16.*

[19] Ni Hong-Guang and Wang Ji-Zong (2000). "Prediction of compressive strength of concrete by neural networks". *Cement and Concrete Research*, *30 (8), pp 1245-1250.*

[20] Ozbas Emine Elmaslar, et al. (2019). "Hydrogen production via biomass gasification, and modeling by supervised machine learning algorithms". *International Journal of Hydrogen Energy, 44 (32), pp 17260-17268.*

[21] Song Yan-Yan and Ying LU (2015). "Decision tree methods: applications for classification and prediction". *Shanghai archives of psychiatry, 27 (2), pp 130.*

[22] Topcu Ilker Bekir and Sarıdemir Mustafa (2008). "Prediction of compressive strength of concrete containing fly ash using artificial neural networks and fuzzy logic". *Computational Materials Science, 41 (3), pp 305-311.*

[23] Wankhade MW and Kambekar AR (2013). "Prediction of compressive strength of concrete using artificial neural network". *International Journal of Scientific Research and Reviews, 2 (2), pp 11-26.*

[24] Wu SS, et al. (2010). "Predictive modeling of high-performance concrete with regression analysis". *in 2010 IEEE International Conference on Industrial Engineering and Engineering Management. IEEE.*

[25] Yang Lingjian, et al. (2017). "A regression tree approach using mathematical programming". *Expert Systems with Applications, 78 pp 347-357.*