Available online at www.scientiaresearchlibrary.com

Scientia Research Library **ISSN 2348-0424**
USA CODEN: JETRB4

Journal of Engineering And Technology Research, 2014, 2 (2):70-80
(*http://www.scientiaresearchlibrary.com/arhcive.php*)

Prediction of Compressive Strength of Concrete with a Skewed Pattern-Application of Artificial Neural Network Approach

A. Rahman[1a], M. Hadiuzzaman[2], Ajit K. Majumder[3], M. N. Uddin[4]
[1a]*Lecturer, Department of Statistics, Jagannath University, Dhaka-1100, Bangladesh,*
[2]*Assistant Professor, Department of Civil Engineering, Bangaldesh University of Engineering and Technology (BUET), Dhaka, Bangladesh.*
[3]*Professor, Department of Statistics, Jahangirnagar University, Savar, Dhaka, Bangladesh.*
[4]*Bangladesh Council of Scientific and Industrial Research (BCSIR), Dhaka, Bangladesh.*

ABSTRACT
Concrete has a versatile use in the construction practice for its availability, cheap rate, and flexibility of handling. As a result, in the construction process it is always important to measure the concrete compressive strength as strength properties of cement paste mixture. A smart modeling approach Artificial Neural Networks(ANNs) have recently been introduced as an efficient and powerful modeling technique for applications involving a large number of variables, especially with highly non-linear and complex interactions among input/output variables. In this paper, an artificial neural network of the feed-forward back-propagation type has been applied as a data treatment technique. The 28-day compressive strength values are considered as the aim of the prediction. A total of 269 specimens are selected. The system is trained and validated using 188(70%) pairs chosen randomly from the data set and tested using the remaining 81(30%) pairs. Results indicate that models performed quite well in predicting the compressive strength in case of training dataset and also for independent data set.

Keywords : Concrete compressive strength, Artificial Neural Network (ANN), feed-forward back propagation, skewed distribution.

INTRODUCTION

Concrete has been used as a construction material for more than a century. Concrete is the only major building material that can be delivered to the job site in a plastic state. This unique quality makes concrete desirable as a building material because it can be modeled to virtually any form or shape. Concrete provides a wide latitude in surface textures and colors and can be used to construct a wide variety of structures, such as highways and streets, bridges, dams, large buildings, airport run ways, irrigation structures, breakwaters, and docks, silos and farm building, houses, and even barges and ships (Vahid and Mohammad, 2010). Concrete is inert mass which grows from a cementing medium. Concrete is a product of two major component one is the cement paste and another is the inert mass. In order to form the cementing medium, cement would mix water. Coarse aggregates and fine aggregates are the part of inert mass. In properly mixed concrete, these materials are completely surrounded and coated by cement paste filling all the void space between the particles (Raju, 1979). Strength is the design property by the concrete. An overall picture of concrete quality is being reflected by the concrete strength. There are many factors which control concrete compressive strength. Concrete mix proportioning, aggregate quality, type of cement, and the most important one is the water cement ratio. Water cement ratio has a critical input on concrete strength characteristic (Hasan, 2009).

Thus the main criterion for evaluating the compressive strength of concrete suggested is the strength of concrete on 28^{th} day. Therefore one should wait 28 days to achive 28-day strength of each layer of concrete (Vahid *et. al.,* 2010). Concrete mix design is a process done by using code recommendation and sometimes by experience. If due to some experimental error in mix design the test result fail to achieve the designed strength, then repetition of the entire process becomes mandatory, which can be costly and time consuming. For every failure it is necessary to wait at least 28 days, thus the need for an easy and suitable method for estimating strength is being felt all the time. Hence a rapid and reliable concrete strength prediction would be of great significance (Kheder *et. al*, 2003).

Over the last few decades, a considerable volume of research has been directed toward gathering the strength relations for concrete in compression. Many studies are being carried out to explore the concrete behavior and its prediction. Different approaches using regression functions have been proposed for predicting the concrete strength (Snell *et. al.,* 1989). Traditional approaches such as design of experiments are established based on empirical relationship and experimental data which are improving day by day (I-chang Yeh, 2006). Attempt has been developed by M. M. Hasan and A. Kabir (2011) for a relationship between concrete strength and its age and finally expresses this relationship with a simple mathematical equation. Some smart data driven self adaptive methods such as Artificial Neural Network (ANN), in that there are few a priori assumptions about the models for problem under study have been introduced (Kasperkiewicz *et. al.*, 1995; Vahid *et. al.*, 2010; E. Rasa *et. al.,* 2009; I-cheng Yeh, 2006, Guoqiang Zhang, *et. al.,* 1997; Vanluachene and Sun 1990). However, Hajela and Berke (1991) demonstrate that neural networks can be used for rapid analysis for structural optimization. Neural network also applied by Szewczyk and Noor (1996, 1997) for sensitivity and non-linear structural analysis; by Kushida *et. al.,* (1997) to develop a concrete bridge rating system; by Hegazy *et. al.,* (1998) to develop a model for the load defective behavior of concrete strain distribution at failure, reinforcing steel strain distribution at failure and crack-pattern formation of concrete slabs.

In recent years, Artificial Neural Network (ANN) has shown exceptional performance as regression tools, especially when used for pattern recognition and function estimation. They learn from examples and capture subtle functional relationships among the data even if the underlying relationships are unknown or hard to describe (Guoqiang Zhang *et. al.,* 1997). Unlike traditional parametric models, these models are able to construct a supposedly complete relationship between input and output variable with an excellent level of accuracy compared with that of conventional methods (Anderson, *et. al.,* 1992). In comparison to parametric methods, ANNs can deal with relatively imprecise or incomplete data and approximate results, and are robust. They are highly parallel that is, their numerous independent operation can be executed simultaneously (Haykin, 1994). Objective of all studies in relation with ANN, that have been carried out was to make concrete strength predictable and efficient prediction for a given level of concrete components structure.

In most application where neural networks are expected to model highly non-linear and skewed, it shows a non-uniform distribution. A recent study by Kumar (2005) which compares ANN and classical regression methods shows that skewness in data set should be reduced using some transformation like power transformation before carrying out ANN analysis. Altun *et. al.*, (2003) demonstrated that highly skewed data distribution deteriorates the performance of the multilayer perceptrons (MLP) type neural network.

The objective of this study is to make an attempt to gather predictable concrete compressive strength using Artificial Neural Network (ANN) considering different transformation technique where its component arise with continuous feature-with a problem of skewed distributional feature.

.

MATERIALS AND METHODS

Artificial Neural Network (ANN)

Artificial Neural Network (ANN) modeling, a paradigm for computation and knowledge representation, is originally inspired by the aspect of the information processing and physical structure of the brain with a web of neural connection (see figure 1). Therefore some writers classified it as a "microscopic", "whole box" system and an expert system as a "microscopic", "black-box" system (Eldon Y. Li, 1994). Artificial neural network are used in three main ways: (i) as models of biological nervous system and intelligence, (ii) as real-time adaptive signal processors controllers implemented in hardware for applications such as robots, (iii) as data analytic methods (Warren, 1994).

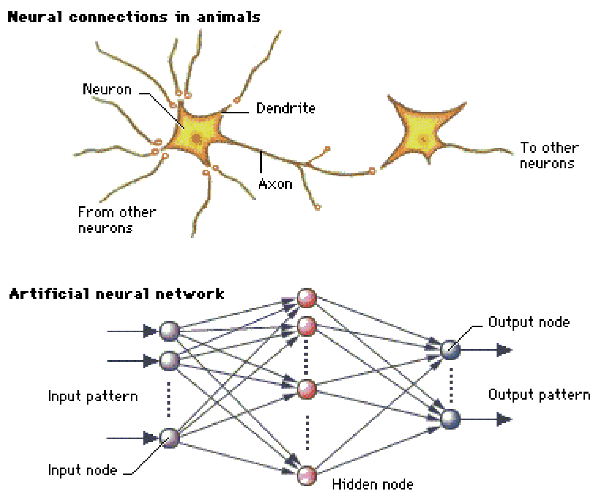


Figure 1: The Neural connection in animals' biological neuron (in top) and the counterpart Artificial Neural network structure (in bottom)

An artificial neural network is a network of many simple processors that are called nodes. A multilayer perceptron may be thought of as consisting of layers of parallel data processing cells (E. Rasa *et. al.*, 2009). Each node (neuron) has a small amount of local memory. Nodes in the input layer only act as buffers for distributing the input signals to nodes in the hidden layer. The nodes are connected by connections, each usually carrying numeric data called weights encoded by any of the existing methods. Each input signals after weighting them with the strength of respective connections from the input layers and computes its output as a function of the sum. The nodes operate on the local data and on the inputs they receive via the connections. The difference between the computed output and the target are combined together by an error function to give the network verification set and used to keep an independent check of the progress of the algorithm. Training of the neural network is stopped when the error for the verification set begins to increase (Anderson, *et. al.*, 1992; Haykin, S., 1994; Ju-Won *et. al.*, 1999).

The main principle of neural network computing is the decomposition of the input-output relationship into a series of linearly separable steps using hidden layers (Haykin, S., 1994). There are three distinct steps in developing an ANN based solution: i) data transformation or scaling, ii) network architecture definition, when the number of hidden layers, the number of nodes in each layer and the connectivity between the nodes and set, iii) construction of learning algorithm in order to train the network (Anderson, *et. al.*, 1992; Nehdi, M., *et. al.*, 2001). Figure 2 shows the simple architecture of a typical network that consists of an input layer, series of hidden layers, an output layer and connection between them. Nodes in the input layer represent possible influential factors that affect the network outputs and have no computation activities, while the output layer contains one or more nodes that produce the network output. Hidden layers may contain a large number of hidden processing nodes. A feedforward back-propagation network propagates the information through connection weights from the input layer to the output layers via the hidden layer and compares network outputs with known targets and propagates the error term from the output layer back to the input layer, using a learning mechanism to adjust the weights and biases (Anderson, *et. al.*, 1992; Ju-Won, *et. al.*, 1999).

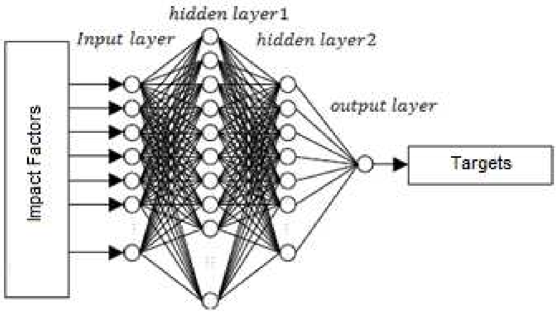


Figure 2: Simple feed-forward back-propagation network Architecture.

A functional link network introduces a hidden layer of neurons. If model includes estimated weights between the inputs and the hidden layers, and the hidden layers use nonlinear activation function such as the logistic function, the model becomes genuinely nonlinear. The resulting model is called a multilayer perceptron (MLP). In general, with the use of the mathematical notations, multilayer perceptron (MLP) works by the following formula

$n_x = number\ of\ input\ nodes\ in\ input\ layer\ (first\ layer)$

$n_h = number\ of\ hidden\ neurons\ in\ hidden\ layer\ (middle\ layer)$

$x_i = ith\ input\ (independent\ variable)\ for\ input\ layer$

$a_j = bias\ for\ hidden\ layer$

$b_{ij} = weight\ from\ ith\ input\ nodes\ to\ jth\ hidden\ layer$

$$g_j = net\ input\ to\ jth\ hidden\ layer = a_j + \sum_{i=1}^{n_x} b_{ij} x_i$$

$h_j = hidden\ layer\ value\ for\ jth\ node = act_h(g_j)$

$c_k = bias\ for\ output\ or\ intercept$

$d_{jk} = weight\ from\ jth\ hidden\ node\ to\ kth\ output\ node$

$$q_k = kth\ net\ input\ to\ output\ layer = c_k + \sum_{j=1}^{n_h} d_{jk} h_k$$

$p_k = kth\ predicted\ value(output\ value) = act_o(q_k)$

$y_k = dependent\ variable(training\ value)\ and\ r_k = residual = y_k - p_k$

Where, act(h) and act(o) are the activation functions for hidden and output layers respectively (Warren, 1994). MLPs are universal approximators (White 1992). MLPs can be used with little knowledge about the form of the relationship between the independent and dependent variable. They are in general purpose, flexible, non-linear models that, given enough hidden neurons and enough data, can approximate virtually any function to any desired degree of accuracy (Warren, 1994). The accuracy is based on minimizing total error between calculated and desired values at output layer during modification of connection weights. One should find the optimal network by trial and error. The most interesting property of a network is its ability to generalize new cases. For this purpose, independent data set is used to test the neural network and check its performance (Anderson, *et. al.*, 1992; Ju-Won, *et. al.*, 1999). Upon successful completion of the training process, a well-trained neural network is not only capable of computing the expected outputs of any input set of the data used in the training stage, but should also be able to predict, with an acceptable degree of accuracy, the outcome of any unfamiliar set of input located within the range of the training data (Anderson, *et. al.,* 1992; Nehdi, *et. al.,* 2001). However, inspection of inputs and target data sets should be necessary in case of skewed pattern, before carrying out neural network. It has been shown (by Altun *et. al.*, 2003) that high-dimensional data deteriorates the performance of the MLP type neural network. This inspection could be based on the modern transformation techniques

Transformation Techniques

In most application where neural networks are expected to model highly non-linear and multi-dimensional functions, experimental data shows a non-uniform distribution. This fact is in line with the central limit theorem which states that data from an experiment approaches to normal distribution as the number of sample taken approaches infinity. However, in real life problems with this assumption is not always realistic as data set might show non-normal and highly skewed distribution. The phenomenon is also mostly true for engineering problem. The common practice adopted by NN practitioners is to scale the data set linearly into a small range before training process. As a result of this treatment, the data is scaled into predefined range but the distribution characteristic of the data is preserved. If it is permissible to transform one set of measures into another, then many possibilities become available for modifying the data to fit more closely the underlying assumptions of statistical tests. An added benefit about most of the transformation is that when we transform the data to meet one assumption, we offers come closer to meeting other assumption as well. As it is well known that messaging data into 0 to 1 provides the best result of artificial neural network, it is required to scaling the input data and find whether the series is skewed. As a rule of thumb, if data are reasonably distributed and if variances are reasonably homogeneous, there is probably nothing to be gained by applying a transformation. As suggested by Tabachinick and Fidelt (2007) and Howell (2007), the following guidelines should be used when transforming data. If data distribution is i) moderately positive skewed use square root transformation, ii) substantially positive skewed (with zero values) use logarithmic transformation, iii) one possible transformation for the values lies between 0 to 1 is $P^q - (1 -$, if the series shows skewedness.

Selection of Database

As concrete compressive strength is of great importance the selection of the database chosen to train a neural network is such that it will be capable of capturing the relationship between the input parameters of cement paste mixtures and its mechanical properties. It must be trained on large and comprehensive sets of reliable experimental data that contain influential factors regarding concrete compressive strength. At a preliminary stage, the data set collected includes the eight input parameters, Cement (component 1), Blast Furnace Slag (component 2), Fly Ash (component 3), Water (component 4), Water-cement ratio (component 5), Superplasticizer (component 6), Coarsh Aggregate (component 7) and finally Fine Aggregates (component 8). The output parameter includes the concrete compressive strength. All the inputs measured as (Kg in a mixtures). Since the designing a strength of concrete normally reports its 28th day strength (Hamid *et. al* 2006), so we have collect data based on the 28th days from the construction work. A total of 269 observations of input and output parameters are collected.

Artificial Neural Network Architecture

There is no effective procedure for indentifying the optimal architecture of a network before training. However, it is important for the hidden layers to have small number of nodes. An excessive number of hidden nodes may cause the network to memorize the training data. In such case, the ANN would not be able to interpolate effectively between adjacent training data points (Rasa, 2009). Moreover, for highly skewed inputs and targets data, it deteriorates the performance of MLP type neural network (Altune *et. al.*, 2003). Also too few hidden nodes, on the other hand, will limit the networks ability to construct an adequate relationship between input and target variable (Anderson, *et. al.*, 1992).

Neural network works best when all the input and output values are between 0 and 1. This requires scaling input data set and output data set into 0 to 1. Inspection of input and target data sets as mentioned in database selection section reveals that their distributions are a substantially positively skewed with zero value. Then according to Tabachnick and Fidell (2007) suggestion, we use appropriate transformation of input data sets. The number of hidden nodes and layers are usually determined via a trial and error procedure. According to the method suggested by Dave Anderson and George McNeill (1992) an upper bound for the number of processing nodes in the hidden layers can be calculated by dividing the number of input-output pair in the training set by the total number of input and output nodes in the network, multiplied by scaling factor between five and ten.

Water cement materials ratio (W/cm), the unit contents of cement, the blast furnace slag, water and fly ash are represented by input nodes, while the output layer contains one node with compressive strength of concrete. The neural network feedforward back-propagation type is used by the "neuralnet" package in R i386 2.15.1. Table 2 shows the neural network parameters used in training in "neuralnet" package. Biases and weights between nodes are modified on a function of input and target data distribution (Altun *et. al.*, 2007). The error between the predicted error and target value is then calculated and stored. The network is presented with the testing data set. The root-mean-square (RMS) of the error is then calculated and back propagated to the network. To avoid the over fitting of the neural network model to the data during iterative, training, a separate set of data was to validate the model at some intervals during training. Training is stopped when the error for the validation set begins to increase.

Table 1: Neural network parameters used in training in "nueralnet" package

Parameter	Value	Description
Hidden	5,3	No. of hidden nodes in each layer
Threshold	0.005	Threshold for error function for stopping criteria
Learning rate	0.10	Learning rate
Algorithm	**"bp"**	Type of algorithm used
Act.fct	**"tanh"**	Type of activation function use
Likelihood	**"True"**	Provides AIC, BIC value

In this study the network was trained and validated, based on 188(70%) training patterns chosen randomly from 269 available data. The remaining 81(30%) pairs of the independent data were used to test the network after completion of training and validation in order to assess its performance on the data to which it has never before been exposed. Then we can use the network to predict the compressive strength of concrete with the different values of inputs. Here, it uses the learning rate .10 and number of iteration 10000 with an error goal 0.005.

RESULT AND DISCUSSION

The network was trained to predict cement paste properties such as compressive strength of concrete using a total of 188 training and validating data sets and 81 testing data sets. To test accuracy of the ANN model, the final trained model was called upon to recall the data not used in the training process.

The final information about the training process provided by the “neuralnet” package with chosen network parameter is given below:

error	*0.557420980652*
reached.threshold	*0.004737527497*
steps	*6295*
Intercept.to.1layhid1	*-3.641857614278*

Here, the training process needed 6295 steps until all absolute partial derivatives of the error function were smaller than 0.005 the required threshold. The estimated weights are range from -37.92 to 11.46. For instance, the bias of the first hidden layer is -3.6418. The estimated weights of the respective input parameters are visualized in figure 4 with trained data set.

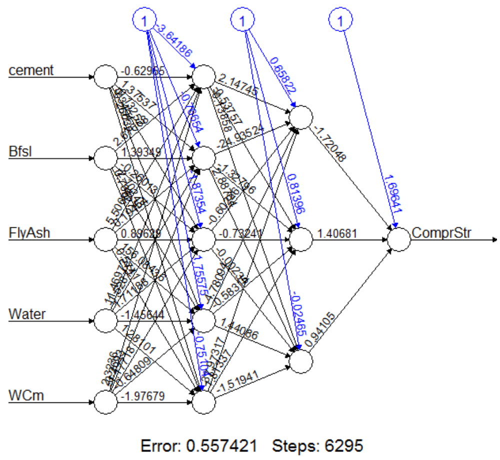


Figure 3: Plot of a trained neural network including trained synaptic weights and basic information about the training process.

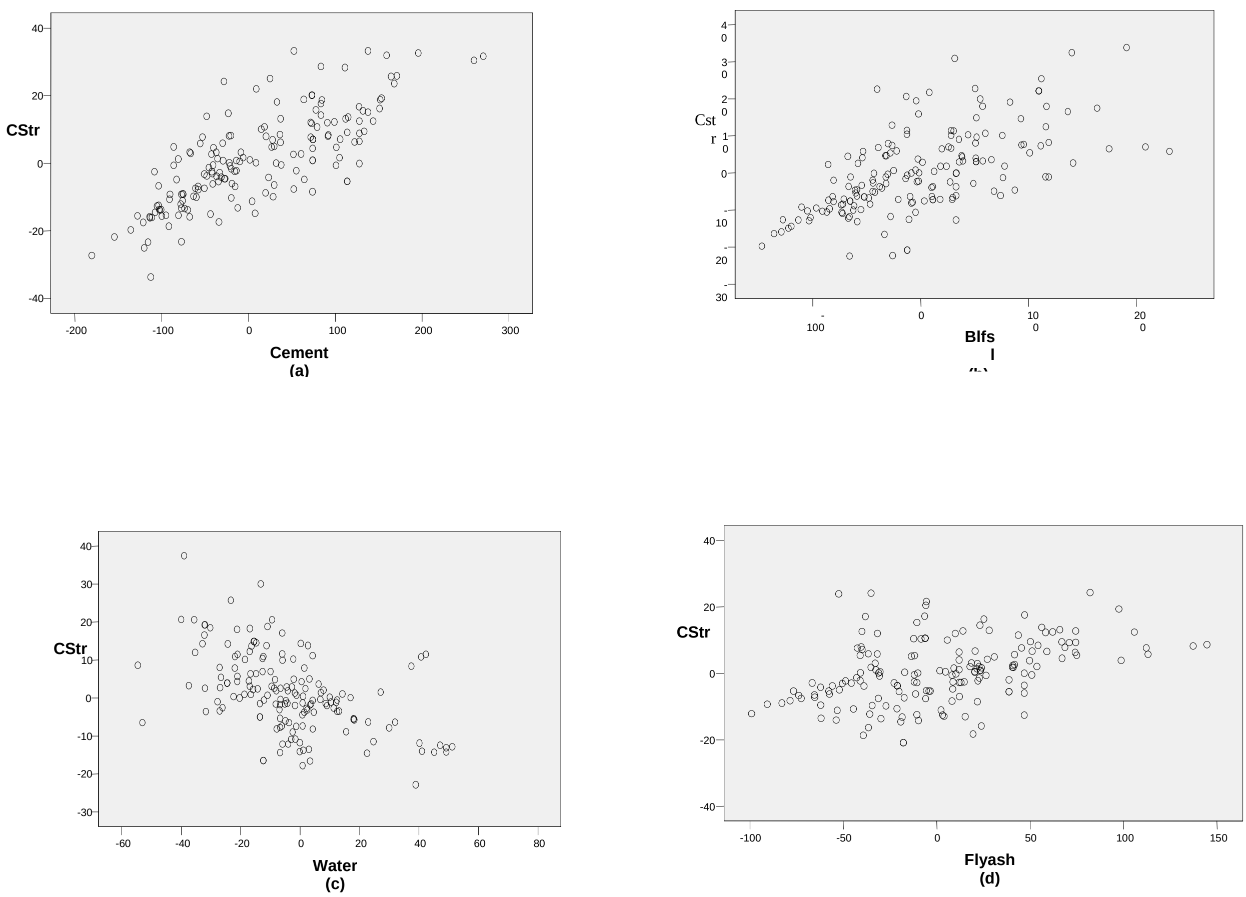
Among all the input parameters, we have select the significant input parameters for output parameter through variable selection procedure and also based on the correlation structure. The variable selection procedure and correlation structure conducted by SPSS 13.0, shows that cement, blast furnace slag, water, water-cement ratio, fly ash are significant components for measuring compressive strength of concrete.

Table 2 shows the related information, the number of data, range, average values and standard deviation of significant input and output variables based on 28^{th} day outcome.

Table 2: Summery statistics regarding measured variables

Variables	**Number of data**	**Range**	**Average**	**Standard deviation**
Cement (kg/ı	269	438	294.478	107.8007
Blast Furnace Slag (kg/ı	269	359.4	73.8094	89.2571
Fly Ash (kg/ı	269	200	48.1180	61.5793
Water	269	125.25	180.558	20.1797
Water Cement Ratio (W/cm)	269	1.6154	0.71177	0.306244
Concrete Compressive Strength(MP_a)	269	69.5060	39.7425	15.31567

Moreover the following figures 4(a) to 4(e) of partial regression plot depicts the individual significant effect of cement, blast furnace slag, water, fly ash and water cement ratio on compressive strength. Here, cement, blast furnace slag, fly ash and W/cm has positive effect.



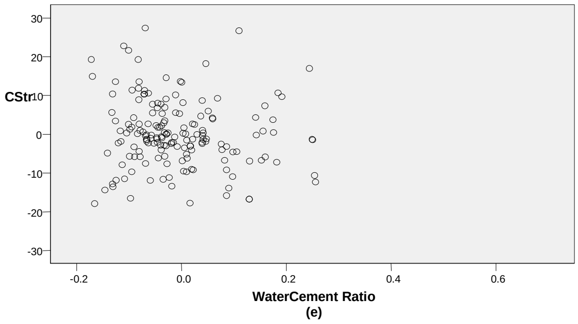


Figure 4: Partial regression plot of compressive strength (output) and each input variables shown in 4(a) to 4(e).

A total of 10 cement paste mixtures, unfamiliar to the network in the range of training data sets, were presented to the ANN model and the network was required to predict the compressive strength associated with each significant mixture. The mixture proportion and the measured and predicted values are listed in Table 3.

Table 3: Measured and predicted values of outputs variable used in testing of ANN model.

Serial No.	*Testing Data Sets*					*Compressive Strength*	
	Cement (Cm)	Blast Furnished Stag	Fly Ash	Water (W)	W/Cm	Measured	Predicted
1	236	157	0	192	0.81	32.8845	32.5035
2	203	173	0	192	0.94	22.3479	21.9546
3	203.5	305.3	0	203.5	1.00	41.6843	42.0034
4	200	133	0	192	0.96	30.4368	30.4134
5	250.2	166.8	0	203.5	0.81	36.9641	35.7895
6	310	0	0	192	0.62	27.8272	27.8045
7	273	103	82	210	0.77	33.7567	33.2457
8	192	190	148	179	0.93	37.1696	36.9456
9	250	144	112	220	0.88	16.4991	16.5076
10	234	115	89	202	0.86	19.9879	19.5198

As mentioned earlier, a set of experimental data, including 269 pairs of data, was used in this study, from which 188 training and validating patterns were chosen randomly and remaining 81 pairs were used as measured data, to test and to verify the efficiency and validity of the predicted values by the network. A good agreement between the measured and predicted values of the compressive strength is observed as shown in figure 5. It can therefore be concluded that the proposed ANN model is adequately able to predict the compressive strength of cement paste.

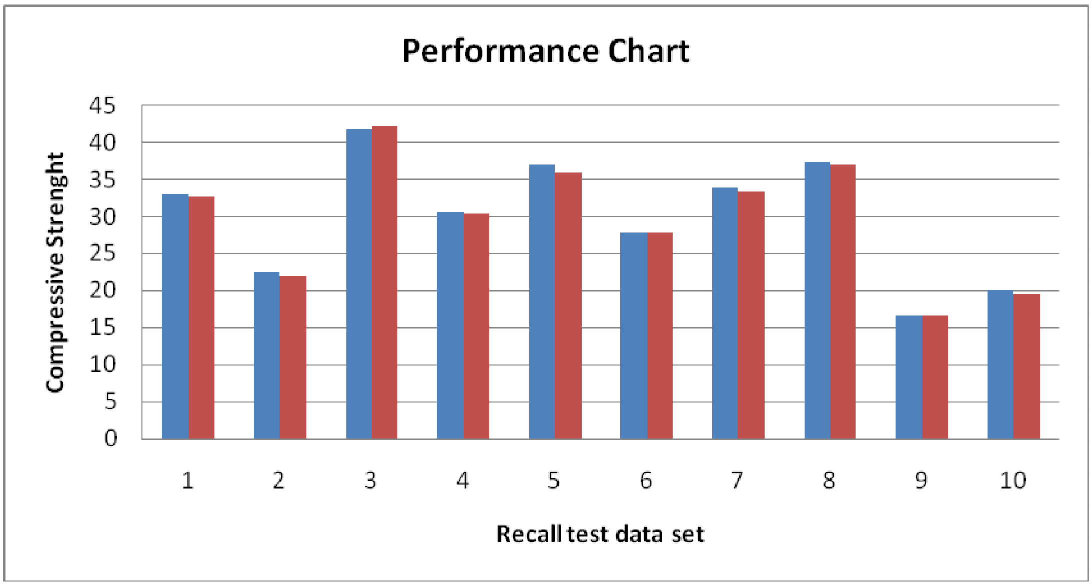


Figure 5: Performance of ANN model predictions for concrete compressive strength for test data sets (blue color bar represents measured and red color bar represents predicted value).

CONCLUSION

This study presents a smart statistical data-mining approach for prediction of the compressive strength of a concrete, based on Artificial Neural Network (ANN) approach. Here, it may be concluding that the proposed model demonstrates the ability of a feed-forward back-propagation neural network to predict the properties of cement paste portion of concrete efficiently. Moreover, the model performed quite well in predicting compressive strength of concrete used in the training process as well as in case of testing data sets that were unfamiliar to the neural network.

REFERENCES

[1]. Altun H., Curtis K. M., and Yaleinz T. (**2003**) Neural learning for articulatory speech synthesis under different statistical characteristics of acoustic input patterns. *Comp. and Elec. Eng.,* 29(1): 687-702.

[2]. Anderson, D. and McNeill, G. (**1992**) Artificial Neural Network Technology. Data and Analysis Center for Software.

[3]. Faruke G., and Stefan F. (**2010**) Neuralnet: Training of neural Networks. *The R journal,* 2(1): 30-38.

[4]. Hamid-Zadeh N., Jamil A., Nariman-Zadeh N., and Akbarzadeh H. (**2006**) A polynomial model for concrete compressive strength prediction using GMDH-type neural networks and genetic

algorithm, Proceeding of the 5th WSEAS International Conference on System Science and Simulation in Engineering, Canany Islands, Spain, Dec: 16-18, **2006**, pp. 13-18.
[5]. Hasan M. M. and Kabir A. (**2011**) Prediction of compressive strength of concrete from early age test result, 4th Annual Paper Meet and 1st Civil Engineering Congress, December 22-24, 2011, Dhaka, Bangladesh.
[6]. Haykin S. (**1994**) Neural Networks: A comprehensive Foundation. 2nd Ed. (MacMillan, New York) 1995, pp 275.
[7]. Kasperkiewicz J., Rach J., and Dubrawski A. (**1995**) HPC strength prediction using Artificial Neural Network, *J. of Comp. in Civil Eng.*, 9(4):279-284.
[8]. Kheder G. F., Al-Gabben A. M. and Suhad M. A. (**2003**) Mathematical model for the prediction of cement compressive strength at the age of 7 and 28 days within 24 hours, *Mat. And Stru.*, 36(1): 693-701.
[9]. Kumar U. A. (**2005**) Comparison of neural networks and regression analysis: A new insight. *Ex. Sys. with app.*, 29(2): 424-430.
[10]. Nehdi M., Jebbar Y. D., and Khan A. (**2001**) Neural network model for cellular concrete. *ACI Mat. J.*, 98(5): 402-409.
[11]. Oluokun F. A., Burdette E. G. and Harold D. J. (**1990**) Early-age concrete strength prediction by maturity-another look. *ACI Mat. J.*, 87(6):565-572.
[12]. Popovics S. (**1998**) History of a mathematical model for strength development of Portland cement concrete. *ACI Mat. J.*, 95(5):593-600.
[13]. Raju N.K., Basavarjah B. S. and Ramakrishna N. (**1979**) A comparative study of concrete mix design procedures. *Ind. Con. J.*, 4(1):13-16.
[14]. Rasa, E., Ketabchi, H., and Afshar, M. H. (**2009**) Predicting Density and Compressive Strength of Concrete Cement Paste Containing Silica Fume Using Artificial Neural Networks, *Transaction A: Civil Eng.*, 16(1): 33-42.
[15]. Snell L. M., Roekel J. V. and Wallace N. D. (**1989**) Predicting early concrete strength. *Concr. Inter.*, 11(12): 43-47.
[16]. Tabacknick B. G., and Fidell L. S. (**2007**) Using multivariate statistics, 5th Ed., Boston: Allyn and Bacon.
[17]. Vahid K., Alilou and Mohammad (**2010**) Prediction of 28-day compressive strength of concrete on the third day using artificial neural network. *Inter. J. Eng.*, 3 (6): 565-575.
[18]. Warren S. S. **1994**: Neural Networks and Statistical Models, Preceedings of the 19th Annual SAS Users Group International conference, April 20-22, 1994.
[19]. White H. (**1992**) Artificial Neural Networks: Approximation and learning theory. 3rd Ed. Oxford, UK Blackwell.
[20]. Won J. O., Won I. L., Tae T. K., and Won J. L. (**1999**) Application of neural networks for proportioning of concrete mixes. *ACI Mat. J.*, 96(1): 61-67.
[21].Yeh C. I. (**2006**) Analysis of strength of concrete using design of experiments and neural networks. *J. of Mat. in Civil Eng.*, 32(1): 597-604.