



*Article*

## Estimating Concrete Compressive Strength Using MARS, LSSVM and GP

Rahul Biswas<sup>1,a,\*</sup>, Baboo Rai<sup>1,b</sup>, Pijush Samui<sup>1,c</sup>, and Sanjiban Sekhar Roy<sup>2,d</sup>

<sup>1</sup> Department of Civil Engineering, National Institute of Technology Patna, 800005, India

<sup>2</sup> School of Computing Science and Engineering, VIT University, Vellore, India

E-mail: <sup>a</sup>[rahulbiswas.ce16@nitp.ac.in](mailto:rahulbiswas.ce16@nitp.ac.in) (Corresponding author), <sup>b</sup>[baboo.rai@nitp.ac.in](mailto:baboo.rai@nitp.ac.in),

<sup>c</sup>[pijushsamui@gmail.com](mailto:pijushsamui@gmail.com), <sup>d</sup>[sanjibanroy09@gmail.com](mailto:sanjibanroy09@gmail.com)

**Abstract.** The estimation of concrete compressive strength is utmost important for the construction of a building. Organizations have a limited budget for mix design; therefore, proper estimation of concrete data has a significant impact on site operations and the construction of the building. In this paper, the prediction of concrete compressive strength is done by Multivariate Adaptive Regression Spline (MARS), Least Squares Support Vector Machine (LSSVM) and genetic programming (GP) which is a very new approach in the field of concrete technology. MARS is a supervised technique, performs well for high dimensional data, interacts less with the input variables, whereas LSSVM is generally based on a statistical learning algorithm and GP builds equations that are generated for modeling. All the developed LSSVM, MARS and GP gives an equations for prediction of compressive strength which makes easy to predict the compressive strength of the concrete. The efficiency of the MARS, LSSVM and GP are measured by the comparative study of the statistical parameters and can be concluded that the all the models performed very well as the output results are very close to the desired value, while the MARS slightly outperformed the other two models.

**Keywords:** Concrete, prediction, multivariate adaptive regression spline, least square support vector machine, genetic programming.

**ENGINEERING JOURNAL** Volume 24 Issue 2

Received 8 July 2019

Accepted 9 January 2020

Published 31 March 2020

Online at <https://engj.org/>

DOI:10.4186/ej.2020.24.2.41

## 1. Introduction

The demand for predicting the compressive strength of concrete for building construction is high in the market. Most of the times, it is difficult to estimate the quality of the concrete [1]. Since organizations usually have a limited budget for the preparation of concrete mixtures, a good estimation of the compressive strength of the concrete would have a significant impact on the quality of the structure to be constructed. The estimation of the concrete compressive strength not only can be applied to new constructions, but also restoration undertakings, such as restoration of bridges, highways and buildings, etc. The relationship between the composition and the physical properties of the concrete is non-linear to a great extent, therefore it is difficult to build a mathematical model expressing that relationship between them. Prediction is the most important module for checking the estimation of the compressive strength, and that is the reason why machine learning techniques can be useful for estimating the final output of the concrete. In this work, we have adopted three of the most effective machine learning techniques, namely Multivariate Adaptive Regression Spline (MARS) [2]-[3], Least Squares Support Vector Machines (LSSVM) [4]-[6] and the Genetic Programming (GP) [7] separately to predict the final output parameter namely compressive strength of the concrete. Firstly, we apply the renowned MARS technique, developed by Friedman [2], which is an efficient and effective regression model for non-parametric estimations. The MARS model uses hinge functions as its basic functions, which is responsible for overtaking linear regression models in performance measures [8]. MARS is a supervised technique, performs well for high dimensional data, interacts less with the input variables. In this work, we have obtained a fitting model on the training and test data of concrete, and shown a detail analysis of regression techniques, which has allowed us to predict the output parameter accurately. LSSVM is developed based on a statistical learning algorithm [5, 9]. Estimation of non-linear and classification functions are determined by LSSVM with the combination of the kernel function for high dimensional space vector machines. GP is created based on a genetic algorithm [7]. The capability of Genetic Programming based prediction models, which fuses the genetic algorithm and the symbolic programming, was investigated by Chen and Wang [10]. We also obtained RMSE, MAPE, E, RSR, NMBE (%) and VAF values. Along with that, we obtained better performance of prediction on training and test data. In case of MARS model, we achieved prediction accuracy, using cross-validation method. We validated our proposed model by correctly choosing the model parameters. All the three models were adopted to predict the compressive because it is a very new approach in the field of the concrete technology and also the Engineers always like an equation for predicting design parameters. The developed LSSVM, MARS and GP give an equations for prediction of compressive strength. Hence, practitioners and engineers

can use the established equations for prediction of compressive strength of the concrete in practical purpose also.

## 2. Theoretical Background and Development of the Predictive Models

### 2.1. Multivariate Adaptive Regression Spline

In this first proposed model, we have built a MARS model for the prediction of compressive strength of concrete. In the literature, MARS has been used to extensively demonstrate problems for prediction and classification problems [11]–[16]. The application of general regression models can be found in the literature [17]–[23]. The reader can also find details of MARS from the website maintained by Salford Systems (<http://www.salford-systems.com/MARSCITE.PDF>). MARS can be useful for the data which has the high dimensions. The recursive partitioning approach towards regression model has influenced MARS model. It is based on a non-parametric approach. Limiting the maximum interaction among variable is an important aspect of MARS. Therefore, maximum interaction among variables should be a small number as it helps to make the resulting model more interpretable. Collinearity among the variables which are used for prediction is a major problem in MARS. Therefore to overcome this problem MARS does increase the order of interaction among variables. Also, by increasing the variable parsimony with a little penalty can overcome this situation. For choosing the basis function of the multivariate spline, any criteria can be chosen. The advantage of recursive partitioning in MARS model is that it reduces the outlier responses. With the use of local variables subset MARS can achieve adaptability, which is the main advantage of recursive partitioning. The adopted MARS method shapes the model with the help of two-sided truncated functions of the forecaster 'x' and that has the form as below.

$$(x - t)_+ = \begin{cases} x - t, & x > t \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The above expression works as basis functions to both nonlinear and linear extension and is used to approximate any fundamental function  $f(x)$ .

In multivariate adaptive regression spline model, the dependent variable (output), say as  $y$  and the following equation expresses the term  $M$ ,

$$y = f(x) = \beta_0 + \sum_{m=1}^M \beta_m H_{km}(x_v(k, m)) \quad (2)$$

Here, summation works over  $M$  term with the parameters  $\beta_0$ ,  $\beta_m$  and known  $t$ .

In this work, we have estimated the known value i.e.  $t$  from the model. The hinge function (H) can be expressed through the following equation:

$$H_{km}(x_{v(k,m)}) = \prod_k -1^k h_{km} \quad (3)$$

Here the product of  $k^{\text{th}}$  of the  $m^{\text{th}}$  term is,  $X_{V(k,m)}$ . With the value of  $K=1$  and  $K=2$ , we are able to find pairwise and additive interaction. In this work, we have opted value of  $K=2$ .

It is seen that the quantity of basis functions which is added to the proposed model during the forward stepwise based on the previously calculated maximum, which is actually higher than the best least squares fit (optimal).

For the goodness of the fit, we also consider 20% of the training data as cross-validation data (GCV), which is given as,

$$GCV = \sum_{i=1}^N \frac{(y_i - f(x_i))^2}{(1 - \frac{c}{N})^2} \quad (4)$$

where  $C = 1 + cd$ . Here,  $N$  = number of instances in the dataset,  $d$  = degrees of freedom, that is identical to the number of autonomous basis functions. The term  $c$  is used as the penalty because of adding a basis function. In this work, we have it has been observed that the best probable value of  $C$  can be obtained between  $2 < d < 3$ .

## 2.2. Experimental Outcomes of MARS

The dataset for regression modelling compressive strength has been obtained from the UCI machine learning repository. It has nine quantitative attributes [24]. We have considered the ninth attribute as dependent attribute and remaining are independent attributes. The data organization was done in two steps. We have split the data into 70%-30% training set and testing set. This dataset has 1030 numbers of instances. Concrete is the most vital material in structural designing; it has a high nonlinear function with the attributes of constituents and age. Also, for checking cross validations we take 20% of the train data named as CV, especially for MARS model. For the MARS implementation, we have considered R open source programming environment along with that for LSSVM and GP, we have adopted Matlab programming environment. Firstly, we have shown the importance of variety of attributes of train data, keeping the output parameter as the dependent variable. The graph we have obtained as follows:

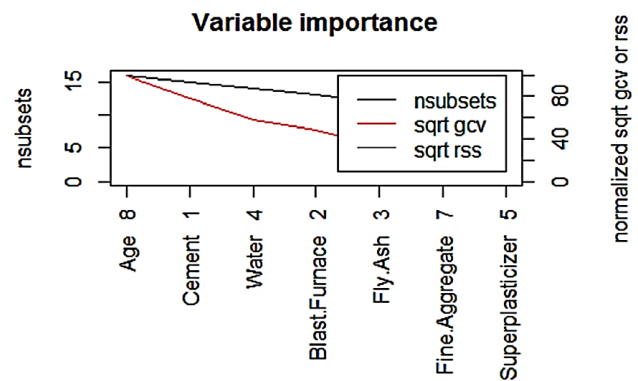


Fig. 1. Variable importance using MARS Model.

Here in Fig. 1, rows of the training data sorted as subsets and it can be perceived that AGE is the foremost important variable followed by CEMENT and so on. The values of GCV (Generalized Cross-Validation) and RSS (Residual Sum of Squares) for the variable are normalized so the major clear decrease is 100. As mentioned above, for Cross-Validation (CV) consist of 20% of the training data, we obtained errors (GRsq and RSq) which have been shown in the following graph.



Fig. 2. Cross-validation graph obtained by MARS.

From Fig. 2, it can be observed that GCV would have picked 13 terms, but cross-validation picked 12 terms. The out of the fold  $R^2$  for every fold model which is represented as pink lines. The red line represents the mean out of the fold. Next figure (Fig. 3), we have shown all predictors that should participate as a linear predictor in its place of hinge function; It can be observed from Fig. 3 attribute that 'Cement' is not as important as others. As it is almost linear by nature.

In the following Fig. 4, we have elaborately shown residuals vs. fitted, cumulative distribution and normal quantile vs. residual quantile graphs, when MARS applied upon test data of concrete.

The 'actual vs predicted' graph obtained from train data using 'MARS' model has been shown below in Fig. 5.

The 'actual vs. predicted' graph obtained from test data set by 'MARS' model has been shown below in Fig. 6

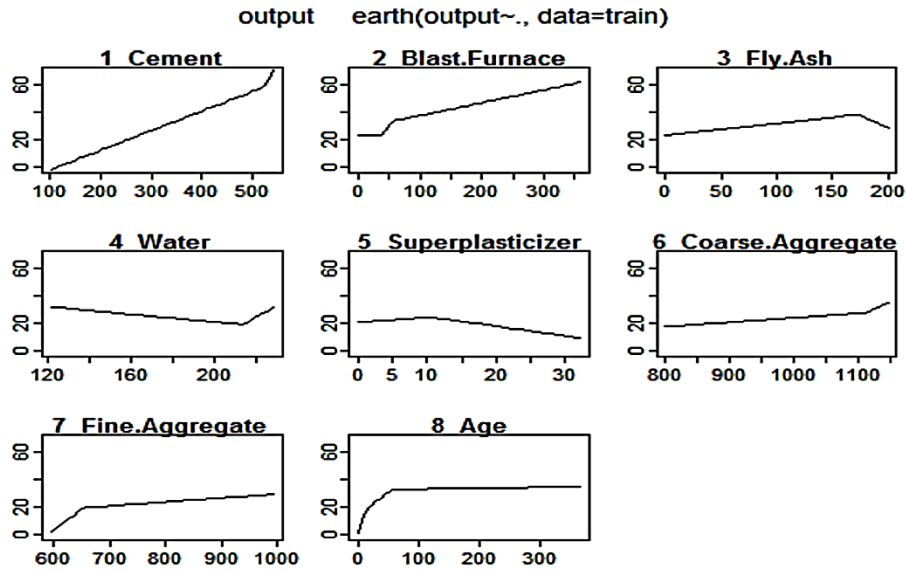


Fig. 3. Participation of predictors obtained by MARS.

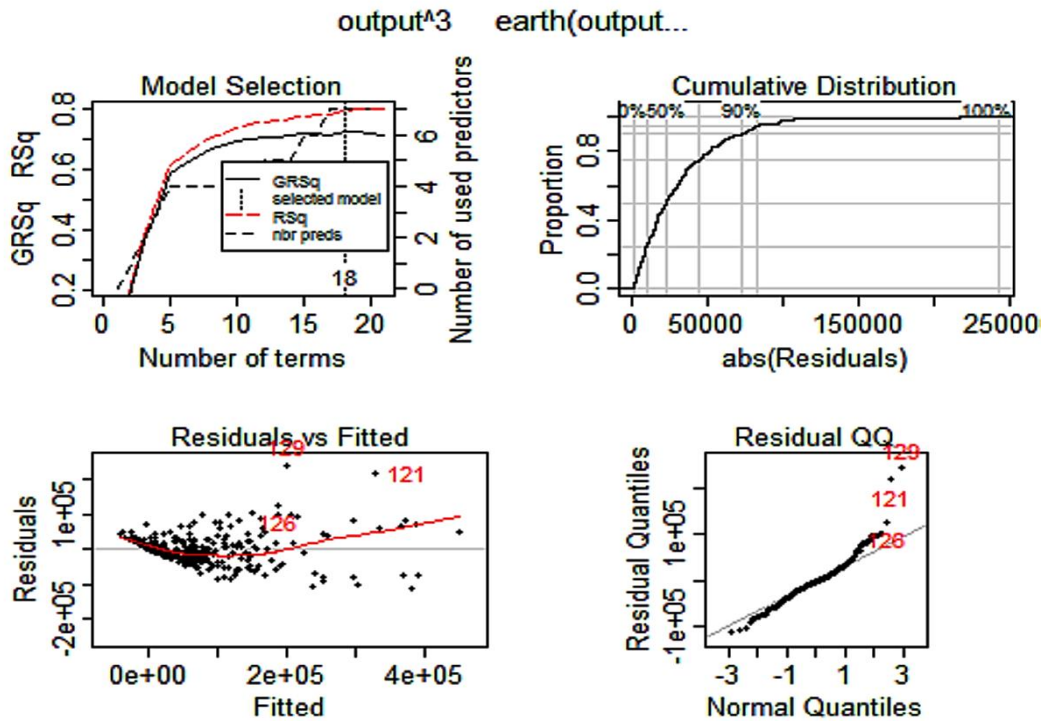


Fig. 4. Details of Residuals applied on Test Data obtained by 'MARS' model.

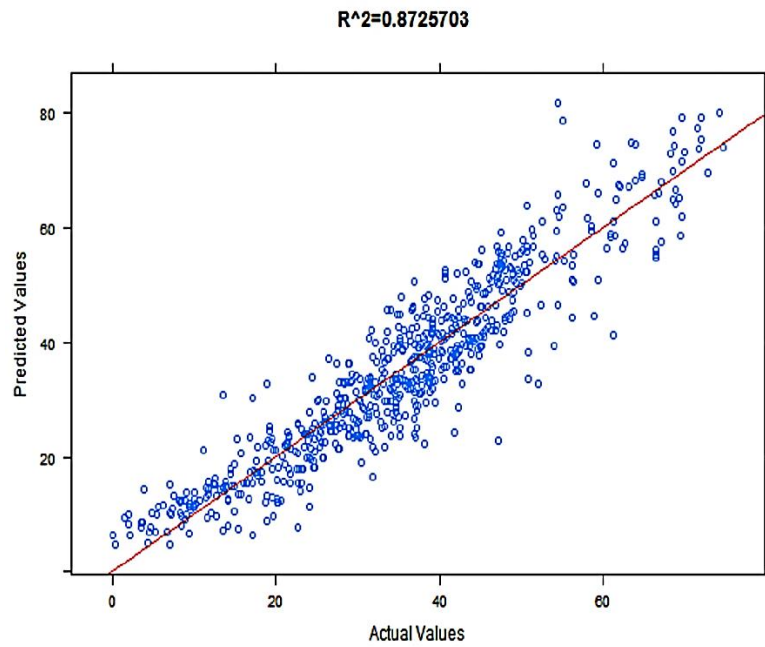


Fig. 5. Actual vs Predicted graph obtained from Train Data using 'MARS' model.

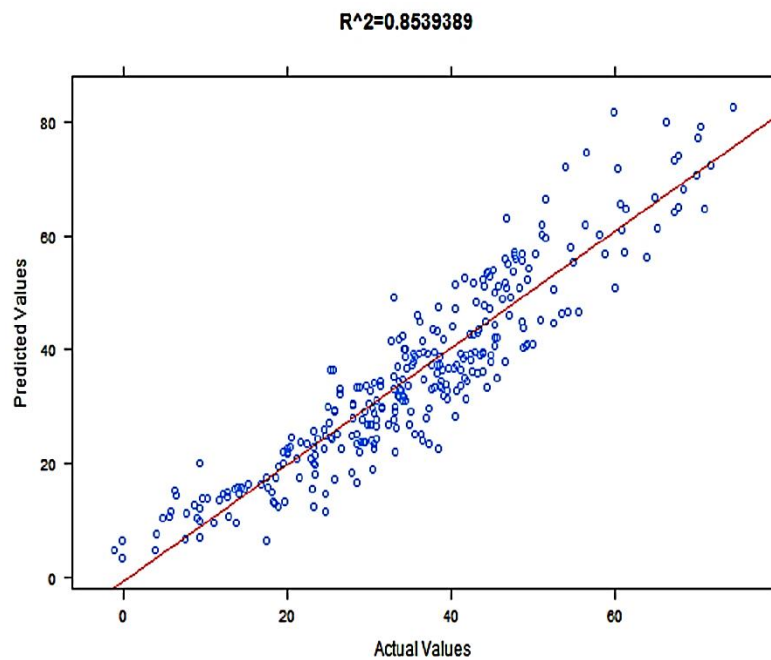


Fig. 6. Actual vs Predicted applied on Test Data.

The final equation of the output variable (i.e. maximum compressive strength) has been obtained as combinations of basis functions and is given below:

$$\begin{aligned}
 & \text{Output}(\text{maximum compressive strength}) \\
 & = 103.0197 \\
 & - 0.1495273 \times \max(0, 236 - C) \\
 & + 0.1039996 \times \max(0, C - 236) \\
 & - 0.1011699 \times \max(0, 42.1 - B) \\
 & + 0.1092582 \times \max(0, B - 42.1) \\
 & - 0.1073434 \times \max(0, \\
 & B - 189) \\
 & - 0.05140282 \times \max(0, 174.2 - F) \\
 & - 0.3996787 \times \max(0, F - 174.2) \\
 & - 0.4039404 \times \max(0, 145.9 - W) \\
 & - 0.3215826 \times \max(0, W - 145.9) \\
 & + 0.7121183 \times \max(0, W - 210) \\
 & - 1.630315 \times \max(0, 2 - S) \\
 & - 0.4188873 \times \max(0, S - 2) \\
 & - 0.5888875 \times \max(0, 613.2 - FA) \\
 & - 0.981171 \times \max(0, \text{Age} - 14) \\
 & - 1.341536 \times \max(0, 56 - \text{Age}) \\
 & + 0.9884349 \times \max(0, \text{Age} - 56)
 \end{aligned}$$

### 3. Basic Sketch of LSSVM and Model Development

The second method used for concrete prediction is LSSVM which was developed by Suykens and Vapnik [4], [5]. This section will give a small description of the methodology of LSSVM. Estimation of non-linear and classification functions are determined by LSSVM with the combination of the kernel function for high dimensional space vector machines. Vapnik SVM classifiers have been modified to execute a set of linear equations which is related to various areas and it may be comparatively easier than solving quadratic programs. Properties of LS-SVM classifier are stated below. The below Fig. 7 has shown a generic comparison of SVM and LSSVM.

#### i. Kernel function

Mercer condition has to be satisfied and the positive value of the kernel function is chosen

#### ii. Global and unique solution

Unique solution is obtained by solving the two problems of nonlinear LSSVM to linear Karush –Kuhn-Tucker model

#### iii. Lack of Sparseness

Values of every data points will be equal to zero in the case of LSSVM classifier. The above diagram shows the comparison between SM and LSSVM classifier with the

absolute values and sorting the value in descending order. In the case of LSSVM the each and every data points contribute to the model. The data points that are larger are closer and far from the decision boundary. The following flowchart shows the basic steps of LLSVM in the below Fig. 8.

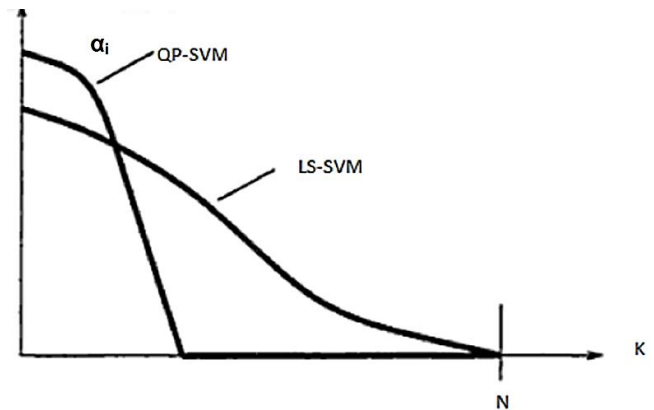


Fig. 7. Comparison graph of SVM and LSSVM.

Flowchart of LLSVM

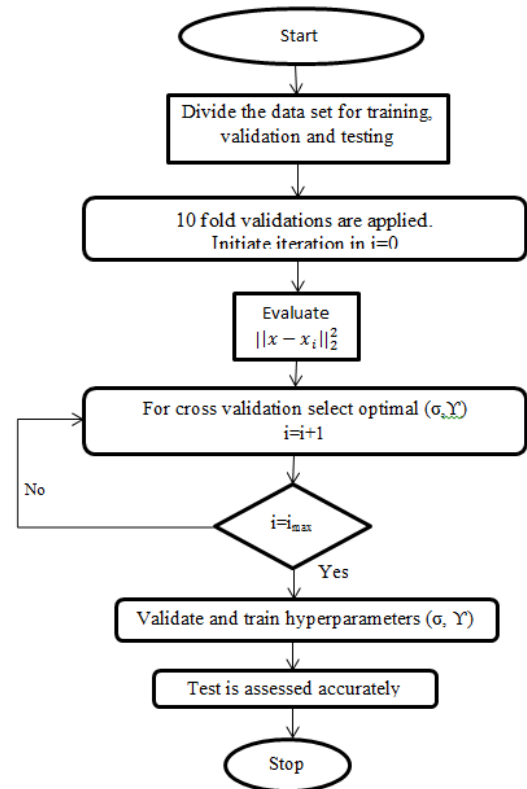


Fig. 8. Example of LSSVM Classifier to formulate grid search.

The basic formula of LSSVM is shown below [4], [5]:

$$y(x) = w^T \phi(x) + b; \quad w \in \mathbb{R}^n \quad b \in \mathbb{R} \quad (5)$$

where  $\phi(\cdot)$  is the nonlinear mapping which plots the input data into a higher dimensional feature space; again  $w$  is

expressed as weight vector which is adjustable and  $b$  stands for scalar threshold.

Now the following optimization problem helps to find the value of  $w$  and  $b$ :

$$\text{Minimize: } \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{k=1}^N e_k^2 \quad (6)$$

$$\text{Subjected to: } y(x) = w^T \phi(x_k) + b + e_k, \quad k=1, \dots, N. \quad (7)$$

where,  $e_k$  is error variable and  $\gamma$  the regularization parameter which determines the trade-off between the minimization, smoothness and fitting error and  $e_k$  denotes error [25].

By applying Lagrange multipliers, the above optimization problem is written in the following way.

$$L = \frac{1}{2} \|w\|^2 + \gamma \sum_{k=1}^N e_k - \sum_{k=1}^N \alpha_k \{w^T \phi(x_k) + b + e_k - y_k\} \quad (8)$$

where

$$y_i = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \quad e_i = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} \quad \text{and} \quad \alpha_i = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix}$$

The optimum conditions are given by

$$\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum_{k=1}^N \alpha_k \phi(x_k)$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{k=1}^N \alpha_k = 0$$

$$\frac{\partial L}{\partial e_k} = 0 \Rightarrow \alpha_k = \gamma e_k, \quad k=1, \dots, N.$$

$$\frac{\partial L}{\partial \alpha_k} = 0 \Rightarrow w^T \phi(x_k) + b + e_k - y_k = 0, \quad k=1, \dots, N.$$

By putting value of  $w$  in Eq. (7) we get,

$$y(x) = \sum_{k=1}^N \alpha_k \phi^T(x_k) \phi(x) + b \quad (9)$$

By substituting kernel function  $K(x_k, x) = \phi^T(x_k) \phi(x)$  in the above equation.

$$y(x) = \sum_{k=1}^N \alpha_k K(x_k, x) + b \quad (10)$$

Below Fig. 9 is the graph which shows the change in training output with respect to  $\alpha$  values in case of LLSVM when it is applied upon training data.

Training dataset is used to build the model. Validation of the model is done by using testing dataset. This article uses 824 training instances and 206 datasets as test instances. The dataset is scaled between 0 and 1. In this study, radial basis function has been adopted as the kernel function.

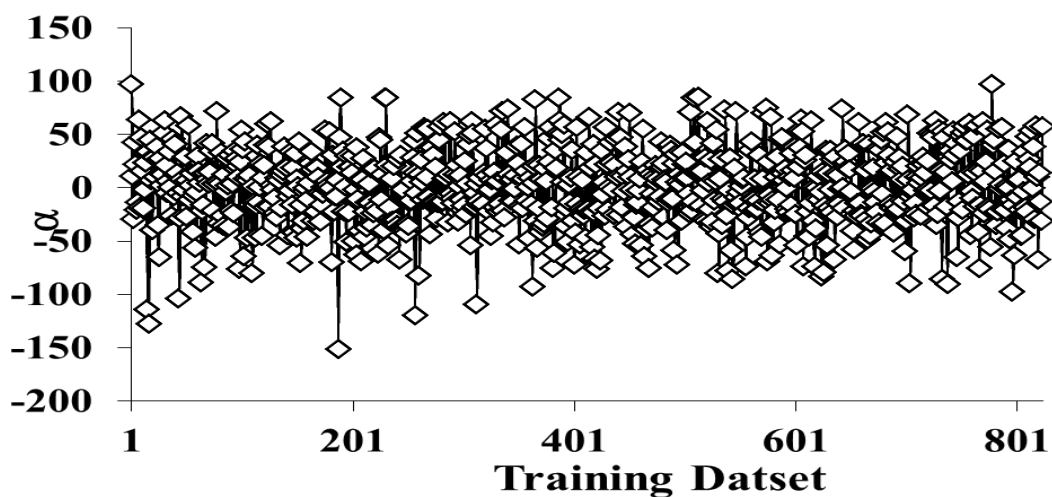


Fig. 9.  $\alpha$  value with respect to training data set using LLSVM.

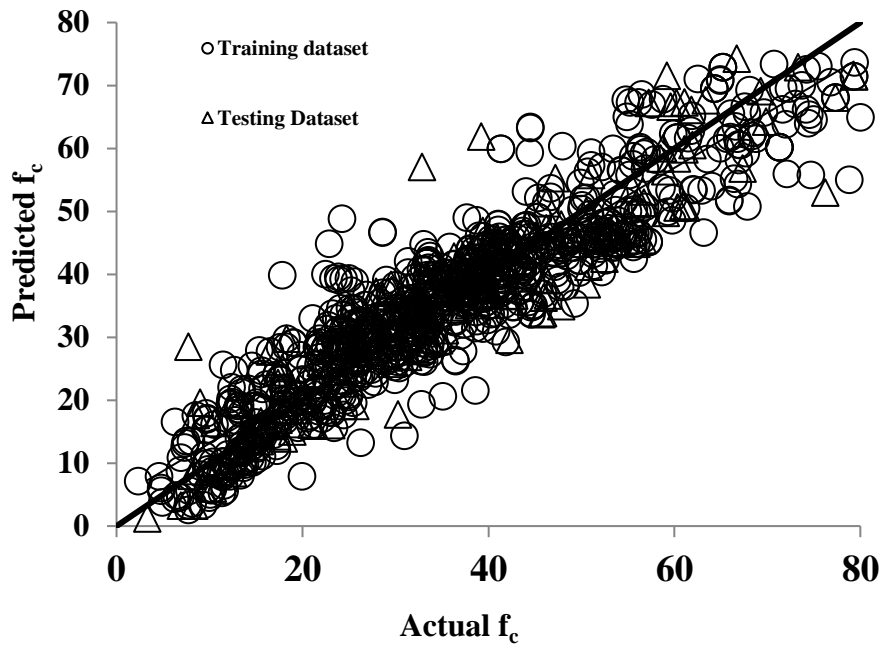


Fig. 10. Actual vs Predicted  $f_c$  of training and test data set in LSSVM.

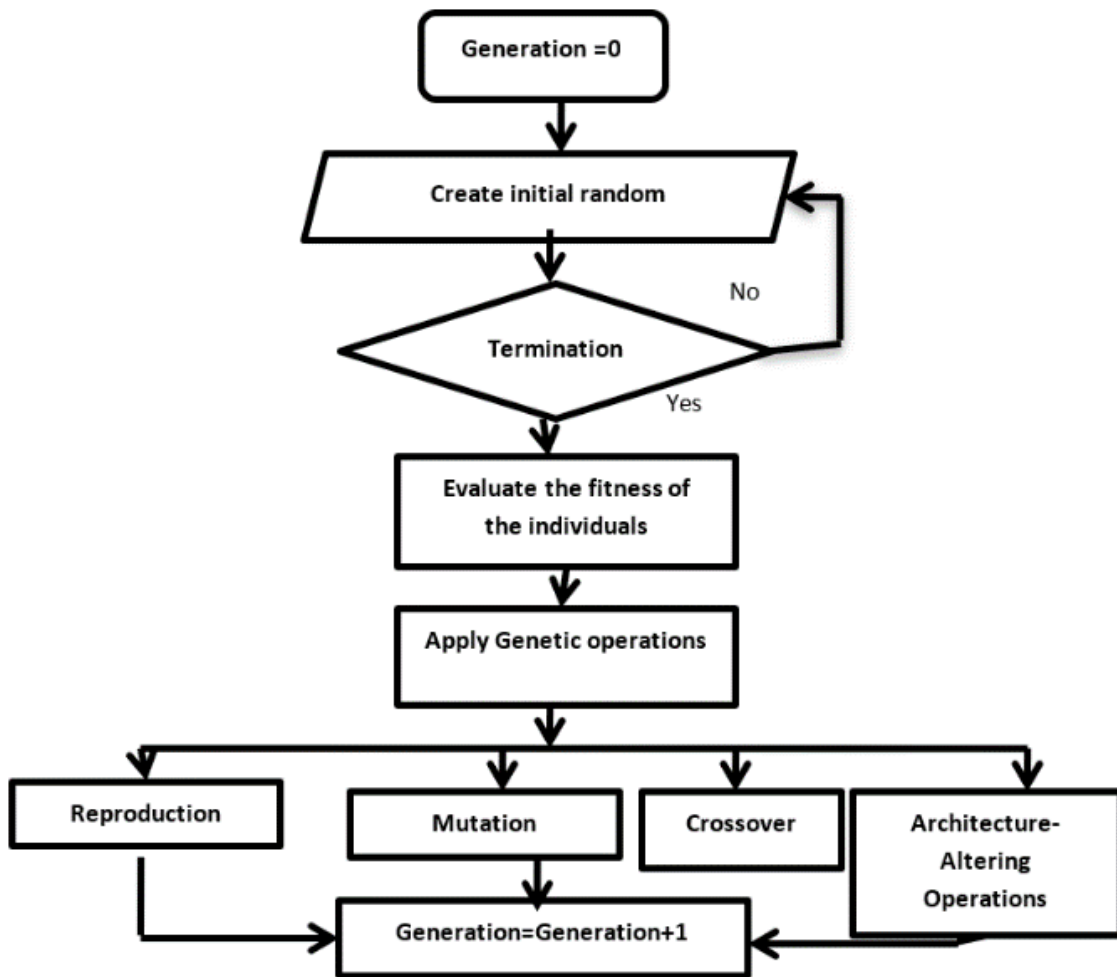


Fig. 11. Flow Chart of operations of Genetic Programming.



Figure 10 shows the plot between actual  $f_c$  and predicted  $f_c$  for training and testing datasets obtained by LSSVM. Table 1 shows the value of RMSE, MAPE, E, RSR, NMBE and VAF of the developed LSSVM. The following equation has been presented based on the developed LSSVM (by putting  $N=824$ ,  $b=-3.977$ ,  $\sigma=20$ , and

$$K(x_i, x) = \exp\left[\frac{-(x_i - x)(x_i - x)^T}{2\sigma^2}\right] \text{ in Eq. (8).}$$

$$f_c = \sum_{i=1}^{824} \alpha_i \exp\left[\frac{-(x_i - x)(x_i - x)^T}{800}\right] - 3.977$$

#### 4. Details of Genetic Programming and Formulation of Mathematical Equations

The third method used for the estimation of compressive strength is Genetic Programming (GP). Koza has developed this method in the year of 1992 [7]. Genetic programming is the evolution computing (ie) a change of transformation exists from one generation to another generation by the iterative process. Similarly, a fresh program is generated by stimulating the operations of genetic process to the specified computer programs. In this work, while predicting the compressive strength of

concrete simultaneously many equations are generated for the modelling. In the following Fig. 11, the steps of genetic programming have been shown.

The performance of GP depends on population size, number of generations, mutation frequency and crossover frequency. After using GP technique on test data, the final expression of GP is given below:

$$\begin{aligned} f_c = & 0.646C + 0.646B + 0.07CA \\ & + 0.07 \exp(\exp(w)) + 0.326 \cos\left(\frac{1}{\exp(8.624FA)}\right) \\ & + 0.840 \cos\left(\frac{1}{\exp(8.456a)}\right) + 0.646 \cos(w) \\ & - 0.194 \exp(B) - 0.344S(F+S) + 0.172Ba \\ & + 0.194 \exp(a)(F+S) - 1.465 \end{aligned}$$

The value of  $f_c$  has been foreseen by using the above equation for training and testing datasets.

Figure 12 depicts the performance of GP in terms of actual vs predicted graph. The calculated values like RMSE, MAPE, E, RSR, NMBE (%) and VAF of GP for the adopted three methods have been produced in Table 1.

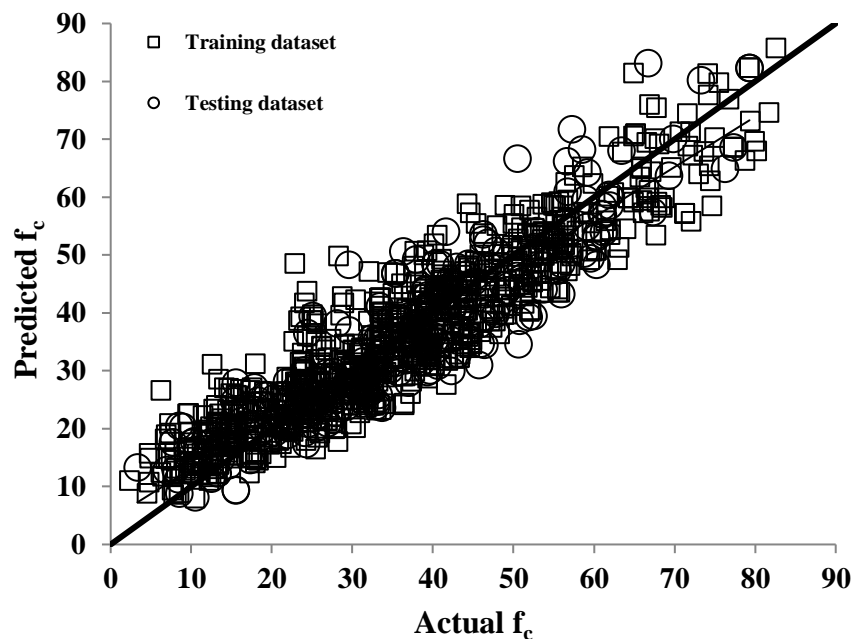


Fig. 12. Actual vs Predicted  $f_c$  values of training and test set using GP.

## 5. Comparative Study and Performance Evaluation of the Adopted Models

The performances of LSSVM depend on the proper choice of  $\gamma$  and  $\sigma$ . The best performance of LSSVM is obtained at  $\gamma = 100$  and  $\sigma = 20$ . The performances of three proposed methods of MARS, LSSVM and GP have been assessed in terms of Mean absolute percentage error (MAPE), Root Mean Square Error (RMSE), Root means square error to observation's standard deviation ratio (RSR), Coefficient of efficiency (E), normalized mean bias error (NMBE), and Variance account factor (VAF), which has been shown in Table 1.

The expression of RMSE, MAPE, E, RSR, NMBE (%) and VAF is given below [1].

$$RSME = \sqrt{\frac{1}{N} \sum_{i=1}^N (A_i - P_i)^2} \quad (11)$$

$$MAPE = \frac{1}{N} \sum_{i=1}^N \frac{|A_i - P_i|}{A_i} \times 100 \quad (12)$$

$$E = 1 - \frac{\sum_{i=1}^N (A_i - P_i)^2}{\sum_{i=1}^N (A_i - \bar{A})^2} \quad (13)$$

$$RSR = \frac{RMSE}{\sqrt{\frac{1}{N} \sum_{i=1}^N (A_i - \bar{A})^2}} \quad (14)$$

$$NMBE = \frac{1/N \sum_{i=1}^N (P_i - A_i)}{1/N \sum_{i=1}^N A_i} \times 100 \quad (15)$$

$$VAF = (1 - (var(A_i - P_i) / var A_i)) \times 100 \quad (16)$$

where  $A_i$  and  $P_i$  represents the actual and predicted values respectively.  $N$  is the total number of dataset.  $\bar{A}$  and  $\bar{P}$  are the average of actual and predicted values respectively.

The comparative analysis shown in Table 1, shows the performance of training data and testing data of LSSVM, GP and MARS. The values of RMSE, MAPE, E, RSR, NMBE (%) and VAF signifies that all the models has given a decent result as the predicted value of compressive strength were very close to the measured value. The value of RMSE of training for LSSVM, GP and MARS is 6.3931, 6.2436 and 6.1403 respectively, whereas the value RMSE for testing is 6.2775, 6.7797 and 6.6832, respectively. The value of MAPE and NMBE (%) for all the models for training and testing indicated very satisfied performance of the model. The value of E are very close to 1 for all the model, though for MARS model the value more close to 1 comparatively to others. Similarly, VAF value indicated a very good result, but the MARS model slightly outperformed the remaining two models as the result is more close to 100.

Hence the parameters used to evaluate the performance of the three models shows that all the model have performed very well and the results of the compressive strength are very close to the desired value. Whereas for overall comparison, it may be concluded that the MARS model very slightly out performed LSSVM and GP model results.

Table 1. Comparative study of LSSVM, GP and MARS.

Parameter	LSSVM		GP		MARS	
	Training	Testing	Training	Testing	Training	Testing
RMSE	6.3931	6.2775	6.2436	6.7797	6.1403	6.6832
MAPE	17.7194	16.5784	19.5454	20.3483	16.9751	18.6039
E	0.8557	0.8487	0.8623	0.8236	0.8678	0.8539
RSR	0.3799	0.3889	0.3710	0.4200	0.3732	0.4396
NMBE (%)	0.1482	-0.5301	0.0002	-0.1672	-0.0024	-0.0047
VAF	85.5662	84.8874	86.2325	82.4264	87.1542	86.0142

## 6. Conclusion

This article uses multivariate adaptive regression spline, least squares support vector machine and genetic programming for the prediction of concrete compressive strength. A database containing 1030 samples with eight variables is adopted to develop a prediction model of compressive Strength. To construct both models, the predictor data sets are selected randomly into 70% (training) and 30% (testing) subsets. The cement content,

percentage replacement blast-furnace, percentage replacement of FA, water, superplasticizer, coarse aggregate, fine aggregate and the age of concrete are the eight independent input variables used to the proposed model, whereas to authenticate the precision of the models RMSE, MAPE, E, RSR, NMBE (%) and VAF were obtained. The effectiveness of the MARS, LSSVM and GP are measured by the comparative study. From the results obtained experimentally, it can be easily established that MARS model performs slightly better than LSSVM

and GP models as the algorithm of MARS produces flexible models with simple linear regression and it also requires stepwise search, addition, and pruning which makes it simpler to interpret.

Though, all the model has provided satisfactory results. It can be concluded that the study suggests a very new technique in the field of civil engineering which can be utilized further to predict the value of compressive strength and the performance of all the models are promising. Also, Engineers like an equation for predicting design parameters. The developed LSSVM, MARS and GP give equations for prediction of compressive strength. Hence, researchers and engineers can use the established equations for prediction of compressive strength of the concrete.

## References

- [1] A. H. Gandomi and D. A. Roke, "Intelligent formulation of structural engineering systems," in *Seventh M.I.T. Conference on Computational Fluid and Solid Mechanics Focus: Multiphysics & Multiscale*, Massachusetts Institute of Technology, Cambridge, MA., 2013.
- [2] J. H. Friedman, "Multivariate adaptive regression splines," *Ann. Stat.*, vol. 19, no. 1, pp. 1–67, 1991.
- [3] P. Saha, M. L. V. Prasad, and P. RathishKumar, "Predicting strength of SCC using artificial neural network and multivariable regression analysis," *Comput. Concr.*, vol. 20, no. 1, pp. 31–38, 2017.
- [4] V. Vapnik and A. Lerner, "Pattern recognition using generalized portrait method," *Autom. Remote Control*, vol. 24, no. 6, pp. 774–780, 1963.
- [5] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Process. Lett.*, vol. 9, no. 3, pp. 293–300, 1999.
- [6] M. Espinoza, J. A. K. Suykens, and B. De Moor, "Least squares support vector machines and primal space estimation," in *Proceedings of the IEEE Conference on Decision and Control*, 2003, vol. 4, pp. 3451–3456.
- [7] J. Koza, *Genetic Programming: On the Programming of Computers by Means Natural Selection*, Cambridge, MA: MIT Press, 1992.
- [8] P. Samui and D. P. Kothari, "A multivariate adaptive regression spline approach for prediction of maximum shear modulus (G max) and minimum damping ratio ( $\zeta$  min)," *Engineering Journal*, vol. 16, no. 5, pp. 69–78, 2012.
- [9] P. Samui, "Determination of compressive strength of concrete by statistical learning algorithms," *Engineering Journal*, vol. 17, no. 1, pp. 111–120, 2013.
- [10] L. Chen and T. S. Wang, "Modeling strength of high-performance concrete using an improved grammatical evolution combined with macrogenetic algorithm," *J. Comput. Civ. Eng.*, vol. 24, no. 3, pp. 281–288, 2010.
- [11] T. Islam, P. K. Srivastava, Q. Dai, M. Gupta, and L. Zhuo, "Rain rate retrieval algorithm for conical-scanning microwave imagers aided by random forest, RRelieff, and multivariate adaptive regression splines (RAMARS)," *IEEE Sens. J.*, vol. 15, no. 4, pp. 2186–2193, 2015.
- [12] W. Zhang, A. T. C. Goh, Y. Zhang, Y. Chen, and Y. Xiao, "Assessment of soil liquefaction based on capacity energy concept and multivariate adaptive regression splines," *Eng. Geol.*, vol. 188, pp. 29–37, 2015.
- [13] S. Kuter, G. W. Weber, Z. Akyürek, and A. Özmen, "Inversion of top of atmospheric reflectance values by conic multivariate adaptive regression splines," *Inverse Probl. Sci. Eng.*, vol. 23, no. 4, pp. 651–669, 2015.
- [14] S. D. Oduro, S. Metia, H. Duc, G. Hong, and Q. P. Ha, "Multivariate adaptive regression splines models for vehicular emission prediction," *Vis. Eng.*, vol. 3, no. 1, pp. 1–12, 2015.
- [15] S. Emamgolizadeh, S. M. Bateni, D. Shahsavani, T. Ashrafi, and H. Ghorbani, "Estimation of soil cation exchange capacity using Genetic Expression Programming (GEP) and Multivariate Adaptive Regression Splines (MARS)," *J. Hydrol.*, vol. 529, pp. 1590–1600, 2015.
- [16] O. Kisi, "Pan evaporation modeling using least square support vector machine, multivariate adaptive regression splines and M5 model tree," *J. Hydrol.*, vol. 528, pp. 312–320, 2015.
- [17] P. Taylan, G.-W. Weber, and F. Y. Ozkurt, "A new approach to multivariate adaptive regression splines by using Tikhonov regularization and continuous optimization," *Top*, vol. 18, no. 2, pp. 377–395, 2010.
- [18] Q. Li, B. Yang, Y. Li, N. Deng, and L. Jing, "Constructing support vector machine ensemble with segmentation for imbalanced datasets," *Neural Comput. Appl.*, vol. 22, no. SUPPL.1, pp. 249–256, 2013.
- [19] J. De Andrés, P. Lorca, F. J. De Cos Juez, and F. Sánchez-Lasheras, "Bankruptcy forecasting: A hybrid approach using fuzzy c-means clustering and multivariate adaptive regression splines (MARS)," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 1866–1875, 2011.
- [20] S. Jiao, G. Zeng, L. He, G. Huang, H. Lu, and Q. Gao, "Prediction of dust fall concentrations in urban atmospheric environment through support vector regression," *J. Cent. South Univ. Technol.*, vol. 17, no. 2, pp. 307–315, 2010.
- [21] Y. Zhang, F. Cao, and C. Yan, "Learning rates of least-square regularized regression with strongly mixing observation," *Int. J. Mach. Learn. Cybern.*, vol. 3, no. 4, pp. 277–283, 2012.
- [22] B. Baesens, S. Viaene, T. Van Gestel, J. A. Suykens, G. Dedene, B. De Moor, and J. Vanthienen, "An empirical assessment of kernel type performance for least squares support vector machine classifiers," in *KES'2000. Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies. Proceedings (Cat. No.00TH8516)*, 2000, vol. 1, pp. 313–316.

- [23] S. Sekhar Roy, R. Roy, and V. E. Balas, “Estimating heating load in buildings using multivariate adaptive regression splines, extreme learning machine, a hybrid model of MARS and ELM,” *Renewable and Sustainable Energy Reviews*, vol. 82. pp. 4256–4268, 2018.
- [24] I.-C. Yeh, “Modeling of strength of high-performance concrete using artificial neural networks,” *Cem. Concr. Res.*, vol. 28, no. 12, pp. 1797–1808, 1998.
- [25] C. Lu, T. Van Gestel, J. A. K. Suykens, S. Van Huffel, I. Vergote, and D. Timmerman, “Preoperative prediction of malignancy of ovarian tumors using least squares support vector machines,” *Artificial Intelligence in Medicine*, vol. 28, no. 3, pp. 281-306, 2003.



**Rahul Biswas**, photograph and biography not available at the time of publication.

**Baboo Rai**, photograph and biography not available at the time of publication.

**Pijush Samui**, photograph and biography not available at the time of publication.

**Sanjiban Sekhar Roy**, photograph and biography not available at the time of publication.