EPJ Web of Conferences **345**, 01036 (2026) https://doi.org/10.1051/epjconf/202634501036
ICE3MT2025

Prediction of compressive strength of brick masonry using machine learning models

Bhumika MR[1], *Madhavi* K[1*], *Shanta* Rangaswamy[2], *S M* Basutkar[1], and *M V* Renuka Devi[1]

[1]Department of Civil Engineering, RV College of Engineering, Bengaluru-59, India
[2] Department of Computer Science and Engineering, RV College of Engineering, Bengaluru-59, India

Abstract. This paper explores predicting brick masonry strength using machine learning models, such as Linear Regression (LRM), Polynomial Regression (PRM), Exponential Regression (ERM), Support Vector Regression (SVR), Decision Trees (DTM), Random Forests (RFM), Artificial Neural Network (ANN) and Adaptive Neuro Fuzzy Inference System (ANFIS). The dataset comprises 245 data points on brick, mortar and masonry strength, split into training, validation and testing sets. Python and MATLAB were used to implement the models. DTM and RFM achieved highest determination coefficient (R^2) values of 0.97 and 0.965. SVR also demonstrated high R^2 value of 0.936. ANN, ANFIS and PRM achieved moderate to strong R^2 values, while LRM and ERM displayed limitations. Overall, DTM, RFM and SVR emerged as the top performers, exhibiting strong predictive capability and good generalization ability. These models are suggested for practical applications in construction projects for their superior accuracy in estimating the masonry compressive strength.

1 Introduction

Predicting the compressive strength (σc) of brick masonry is very important for making sure that buildings are safe, structurally sound, and cost-effective [1]. In the past, testing the strength of brick masonry has depended on experiments, which are expensive, take a long time, and require a lot of work[2,3].Several researchers have come up with empirical correlation of the strength of masonry, brick, and mortar although the correlation is limited to specific testing conditions that cannot be generalized to all masonry. Brick, mortar and masonry exhibit sophisticated nonlinearity when subjected to compressive forces that are difficult to represent using simple empirical models. Therefore, maybe, more sophisticated analytical and numerical models are needed. Machine learning (ML) systems have demonstrated the ability to make predictions using the large amounts of data and identifying relationships between various input characteristics. These tools use past data to make predictions that are more accurate and useful than those made by traditional methods. This makes it more cost-effective and captures complex relationships that traditional methods often miss, which leads to more accurate predictions. Engineers can confidently design structures with anticipated compressive strength, enhancing overall structural

[*] Corresponding author: madhavik@rvce.edu.in

integrity [4].Machine learning algorithms are employed in both regression and classification problems. Linear Regression (LRM), Polynomial Regression (PRM), Exponential Regression (ERM), Support Vector Regression (SVR), Decision Trees (DTM), Random Forests (RFM), ANN and ANFIS are supervised learning algorithm can be trained on datasets of masonry samples with known compressive strength values to learn relations between factors affecting the strength [5-7]. Mishra [8] investigated a range of ML tools, including ANN, SVR, decision trees, and random forests for durability and service life of RC structures predictions. ANN and SVR were the accurate techniques, with ANN achieving the highest accuracy. Garzón-Roca [9] and Zhou[10] explored ANN and ANFIS to estimate hollow concrete and clay brick masonry strength respectively, both models were predicted accurately. Reuter [11] in a comparative study, examined ANN, SVR, random forests, and k-nearest neighbors for predicting concrete failure surfaces. The findings showed that the random forest model surpassed the others, as it is capable of managing nonlinear and intricate relationships between variables. Despite the growing use of machine learning for predicting, there is still a limited number of comparative studies that evaluate the accuracy of models with traditional methods. The aim of this work is to examine and evaluate different ML models for estimating the compressive strength of brick masonry. All of the reviewed articles show that ML can produce accurate predictions, though the best method will depend on the specific problem and dataset. The articles also emphasize how important feature selection, data normalization, and model validation are to improving the precision and reliability of models.

2 Research significance

There are still few comparative studies that assess the accuracy of models using conventional techniques, despite the increasing use of ML for prediction. This paper aims to investigate and assess different ML models for estimating brick masonry compressive strength.

3 Methods

Prediction of compressive strength encompasses several stages, such as gathering data, preprocessing it, and applying a machine learning algorithm for modeling. In this study, a variety of ML models are used to determine the brick masonry compressive strength, as detailed in Table 1.

Table 1. ML models and abbreviations.

Sl. No	MODEL	Abbreviations
1	Linear Regression	LRM
2	Polynomial Regression	PRM
3	Exponential Regression	ERM
4	Support Vector Regression Machine	SVR
5	Decision Tree	DTM
6	Random Forest	RFM
7	Artificial Neural Network	ANN
8	Artificial Neural Fuzzy Inference System	ANFIS

3.1 Data collection and data splitting

The study collected 245 data points on brick, mortar, and masonry strength from various sources as listed in Table 2 [12]. The dataset included burnt clay bricks. The ‘σc’ ranged from 3.5 to 25 MPa, as per the Indian Standard (IS) code guidelines and the mortar compressive strength ranged between 3 to 15 MPa. The ratio of brick to mortar strength ratio was restricted to 0.5 to 3 to ensure compatibility. Some datasets collected are given in the Table 2. The data was then split into training (145 samples), validation (50 samples), and testing sets (50 samples). Training allows the model to learn patterns, while validation fine-tunes the model and adjusts the hyper parameters. The testing set assesses the model's performance on new data and identifies over fitting or under fitting.

Table 2. Dataset comprising of brick, mortar and the corresponding masonry compressive strength[12].

No.	f_b	f_{mo}	f_m	No.	f_b	f_{mo}	f_m	No.	f_b	f_{mo}	f_m
Training Set				**Validation Set**				**Test Set**			
1	7.10	9.60	3.0	1	6.60	13.20	3.2	1	14.14	10.78	7.5
2	20.20	8.90	9.6	2	8.00	7.80	3.8	2	10.00	3.60	4.1
3	5.00	6.50	2.5	3	8.00	8.00	4.2	3	9.80	13.20	4.9
4	8.00	7.20	3.5	4	8.00	10.40	4.6	4	8.00	6.80	3.9
5	7.60	4.30	2.14	5	8.00	9.00	4.6	5	8.00	7.60	4
6	15.80	14.20	9.8	6	8.00	6.40	4.5	6	8.00	10.40	4.6
7	8.00	14.00	4.7	7	7.80	7.11	2.99	7	21.30	14.40	9.97
8	5.30	6.50	2.55	8	8.00	10.00	4.7	8	3.80	6.50	1.58
9	11.40	5.80	5.5	9	8.00	6.50	4.1	9	5.88	3.94	2.05
10	5.88	3.82	2.55	10	8.00	8.40	4.2	10	8.80	3.30	3.2
11	11.60	10.70	7.0	11	5.88	4.27	2.3	11	12.00	4.20	5.7
12	12.00	12.20	8.0	12	5.63	5.94	2.37	12	12.00	6.10	6.3
13	12.00	7	6.2	13	7.75	9.70	3.37	13	5.21	3.32	3.28
14	7.19	6.76	5.6	14	7.19	5.94	5.27	14	10.94	8.15	4.45
15	3.58	3.78	1.99	15	12.00	4.54	7.59	15	5.63	5.40	1.76
16	16.30	8.58	6.86	16	5.70	6.60	1.83	16	12.75	11.20	6.9
17	23.00	12.21	10.00	17	3.80	6.46	1.41	17	15.80	6.46	7.27
18	5.30	6.46	2.3	18	6.50	3.98	3.02	18	5.30	3.98	2.08
19	3.80	3.98	1.36	19	5.65	3.21	1.92	19	7.90	3.98	4.04
20	5.70	10.00	2.8	20	5.80	5.40	2.4	20	4.73	8.03	4.83
21	23.10	13.10	11.54	21	5.80	4.90	2.3	21	4.79	8.03	3.33
22	5.55	8.03	4.11	22	3	3.5	1	22	5.80	3.80	1.9
23	7.33	3.21	2.56	23	3.73	5.08	2.18	23	5.14	5.08	2.867
24	5.80	4.20	2	24	24.70	13.60	15.2	24	6.87	5.08	3.288
25	5	3.5	1.2	25	24.70	13.00	12.6	25	24.70	11.00	10.4
26	13.98	9.36	5.8	26	7.04	3.35	3.8	26	4.41	3.35	2.6
27	6.08	9.42	4.48	27	5.71	5.00	3.5	27	5.708	7.5	4.1
28	24.70	12.00	14.65	28	5.71	5.03	3.5	28	5.63	5.40	1.76

29	4.41	6.34	3.3	29	8.70	5.23	5.72	29	4.75	9.42	5.01
30	8.70	3.54	3.88	30	15.50	5.23	8.14	30	7.40	9.36	5.8

Using tools like pandas, numpy, scikit-learn, matplot lib, and seaborn, three regression models (linear, polynomial, and exponential) were created and assessed. R^2 and RMSE were used for evaluation after the linear regression model was constructed and fitted to the training set. To determine the ideal complexity for the polynomial model, different polynomial degrees were tested. For the exponential model, a linear regression model was fitted after the dataset was transformed using the logarithm. To learn more about predictions and residuals, visualisation techniques with Matplotlib and Seaborn were employed. Models of exponential, polynomial, and linear regression were built and assessed. In order to handle the data, create the models, and display the findings, the analysis required importing the necessary libraries.

The SVR model was refined with grid search CV to identify the set of optimal hyperparameters.Various hyperparameters to modify a model are contained in the param_grid. To find the balance between the margin maximisation and minimisation of errors, the parameter C is also tested using [0.1, 1, 10, 100, 1000]. The suitable test on the kernel parameter is “linear”, “rbf”, “poly”, and “sigmoid” which are suitable in the different types of relationships. In addition, the gamma parameter is also investigated with [scale, auto, 0.01, 0.1,1].These parameters are to determine the shape of the decision boundary of specific kernels such as “rbf”, “poly” and “sigmoid”. The best model that had been generated by the Grid SearchCV was then fitted to the training dataset.

The param_grid provided was applied to the hyperparameters of the Random Forest and the Decision Tree models to fine-tune them. Various values of “max depth” including “None”, “10”, “20” and “30” were tried with the Decision Tree to have the opportunity to observe the various levels of complexity and enable the tree to grow without depth limits. To manage the required samples to split internal node, “min samples split” parameter was experimented with 2, 5 and 10. This affected the granularity of splits and overfitting. Similarly, “min_samples_leaf” was probed with the values 1, 2 and 4 whereby samples in leaf-nodes were fixed to determine the amount of samples required to influence tree simplicity and avoid overfitting.

The “n estimators”, the number of trees in the ensemble, was mixed in the Random Forest with 100, 200 and 300. The same values of “max depth”, “min samples split” and “min samples leaf” used in Decision Tree allowed a direct comparison of the models. The combination that was best determined through optimising these hyperparameters with tools, such as the GridsearchCV which gave the best model performance and generalisation on the dataset provided. Optimal model was trained with grid search being cross-validated on the dataset.

The ANN architecture was optimised with the help of grid search and cross-validation. Differing setups of 1-5 hidden layers and 5-50 neurones per layer with 5 increment steps were experimented. The aim was to optimise the R-squared (R^2) value that is the extent to which the model accounts of the data by optimising the architecture of the model. The ANN was trained and evaluated using the LevenbergMarquardt back propagation algorithm and 5-fold cross-validation. The architecture with the best mean R2 value was the preferred architecture according to the cross-validation process. In the case of the best ANN model, the optimal value of R2 has been reported, the optimal hidden layers and neurones per layer.

The ANFIS model had two input variables, each, connected with two membership functions to signify fuzzy connection with the output. Four rules were used to merge these fuzzy sets to produce the final output. These rules were based on the Sugeno fuzzy inference system (sugfis), and it is available in MATLAB. To model the linear and non-linear dependencies between the inputs and the outputs, the system uses linear functions in the consequent section of every rule. All these methods made the model more flexible and effective in discovering complex relationships among the variables.

4 Results and discussion

Each model was assessed using the coefficient of determination (R^2) and the root mean square error (RMSE), with the results summarized in Table 3. The R^2 value indicates the extent to which the model accounts for the variance in the target variable, with larger values signifying more accurate predictions. Conversely, RMSE measures the average discrepancy between actual values and predicted, with least values reflecting greater precision. According to Table 3, the Decision Tree Model (DTM) exhibited the lowest R^2 and RMSE, recorded at 0.97 and 0.023, respectively.

Table 3. Comparison of performance of all models.

MODEL	TRAINING		VALIDATION		TEST		OVERALL	
	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE
LRM	0.821	0.109	0.796	0.126	0.817	0.110	0.813	0.112
PRM	0.864	0.072	0.861	0.155	0.860	0.103	0.856	0.057
ERM	0.860	0.062	0.873	0.126	0.845	0.112	0.855	0.050
SVR	0.925	0.046	0.961	0.069	0.934	0.068	0.936	0.034
DTM	0.968	0.030	0.979	0.052	0.967	0.050	0.970	0.023
RFM	0.963	0.032	0.981	0.049	0.958	0.060	0.965	0.025
ANN	0.906	0.059	0.929	0.097	0.906	0.088	0.910	0.044
ANFIS	0.923	0.047	0.938	0.090	0.911	0.093	0.923	0.038

4.1 Regression model

All three models demonstrated reasonably good fit to the data, as evidenced by R^2 values ranging from 0.813 to 0.856 for the overall dataset. PRM exhibited the highest R^2 as compared to LRM and ERM. However, PRM had marginally higher RMSE values on the test setas compared to ERM and LRM. Equations 1-3, are proposed for estimating the brick masonry compressive strength, where f_m is the predicted masonry compressive strength, f_b is strength of brick, and f_{mo} is the strength of mortar. Figure 1.a-1.c exhibits the comparison between the actual and predicted values of compressive strength of masonry for the test dataset.

$$f_m = -0.1315 + 0.4048f_b + 0.1906f_{mo} \quad (1)$$

$$f_m = 0.0047 + 0.2771f_b + 0.3225f_{mo} - 0.0018f_b^2 + 0.0096f_bf_{mo} - 0.0141f_{mo}^2 \quad (2)$$

$$f_m = 0.5443 * f_b^{0.74} * f_{mo}^{0.294} \quad (3)$$

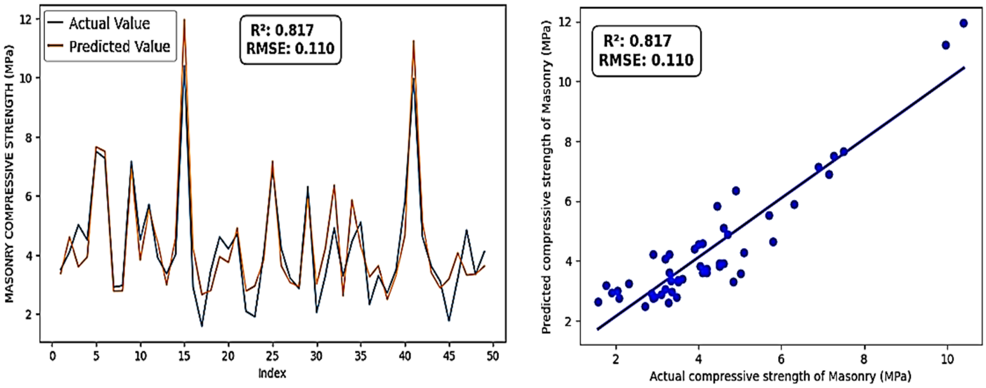

Fig. 1.a LRM

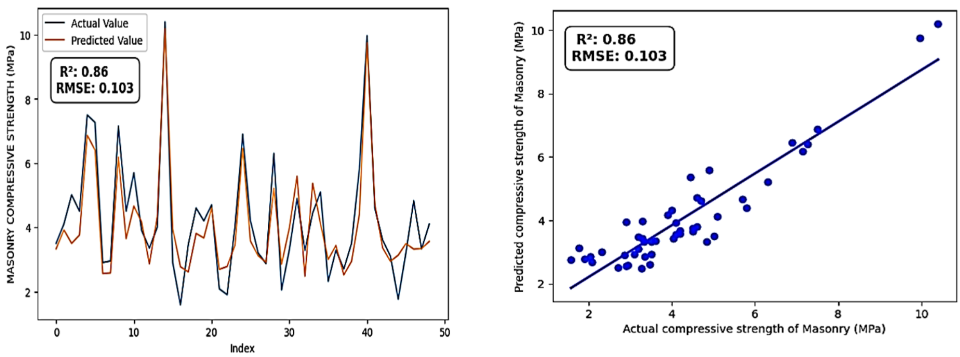

Fig. 1.b PRM

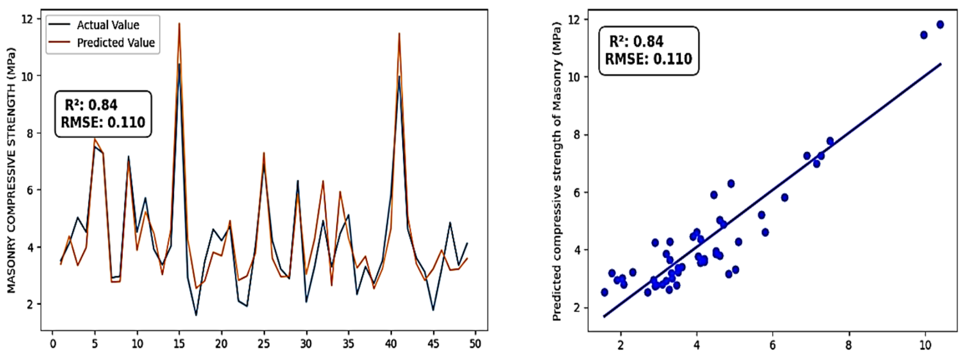

Fig. 1.c ERM

Fig 1.a to 1.c Comparison of actual and predicted compressive strength LRM, PRM and ERM.

4.2 Support vector regression machine (SVR)

The best combination of hyper parameters, C=100, gamma=0.01, and the 'sigmoid' kernel, was determined through the Grid Search CV process for the SVR model. These parameters were selected as they demonstrated the most optimal performance in terms of fitting the data, generalization, and predictive accuracy.The model had high values of R2 of 0.925, 0.961 and 0.934 of the training, validation and test sets respectively. The RMSE values were low at 0.046, 0.069 and 0.068. The train set performance was relatively strong. However, the performance on the test set experienced a minor decrease, which is an indication of possible overfitting. The training and validation dataset outcomes of the model could have led to poor generalization of the model to new data. Further fine-tuning and regularization may be necessary to come up with a more robust and generalizable model. Figure 2.a illustrates its performance on the test dataset.

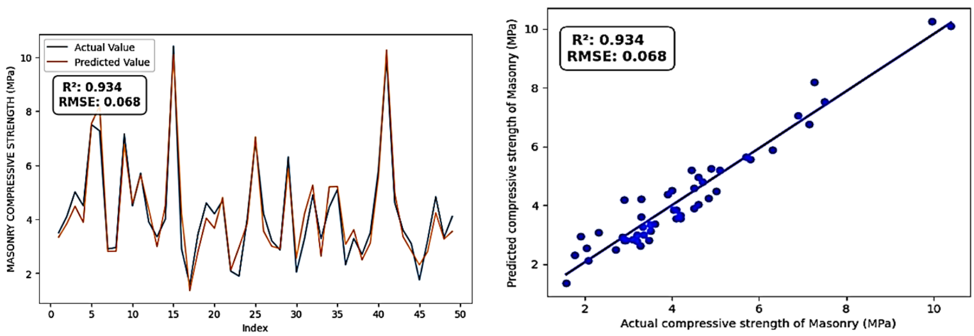

Fig. 2.a SVR

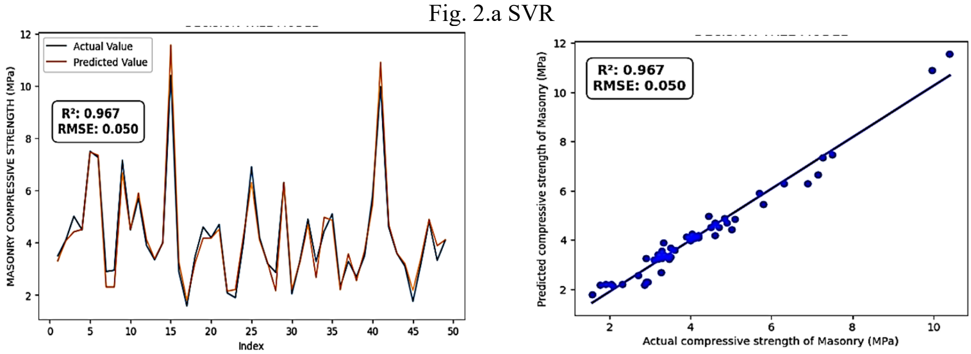

Fig. 2.b DTM

Fig. 2. Comparison of actual and predicted compressive strength for SVR and DTM.

4.3 Decision tree (DTM)

The result of fine-tuning the hyper parameters for the Decision Tree model yielded optimal configuration of 'max_depth' set to None with an unrestricted tree depth, the model can explore complex relationships in the data; 'min_samples_leaf' set to 4 promotes more generalized tree structures, and 'min_samples_split' set to 2 allows for fine-grained splits, capturing detailed patterns in the data. The model produced accurate predictions on the training set, with an R^2 value of 0.968 and a low RMSE of 0.03. With an R^2 value of 0.979 and an RMSE of 0.052, the validation set further demonstrated the DTM's strong predictive abilities. With an R^2 value of 0.967 and an RMSE of 0.050, the test set performance remained consistently high, confirming the robustness of the model. The actual and predicted compressive strength of masonry for the test dataset are compared in Figure 2.b.

Its efficacy is further supported by the low RMSE values on all sets. With high R^2 values and low RMSE values, DTM performed exceptionally well overall.

4.4 Random forest (RFM)

Fine-tuning of the hyperparameters of the RFM produced the optimal hyperparameter configuration. The data did not have intricate relationships to explore since the max depth was set to be "none" meaning that each decision tree in the ensemble would grow unrestrained by any depth limit. As the value of "min_samples_leaf" is used to encourage more generalized tree structures and to eliminate chances of overfitting, the parameter has been modified to 2. The minimum split of samples was also made 2 to allow detailed splits and more complicated patterns to be captured in the data.

The method was reliable and consistent. It produced a very good R^2 of 0.963 and a small RMSE of 0.032 in the training set indicating that it is very adaptive to the data besides being able to predict accurately. This was enhanced even more as the R^2 value was 0.981 and the RMSE was 0.049 in the validation set which means that it can predict new information. The RMSE and R^2 was 0.06 and 0.958 respectively indicating that the RFM was competent despite having a small decrease in the performance in the test set. In Figure 3, the real and predicted masonry compressive strengths of the test sample are compared. With a R^2 of 0.958, and RMSE =0.06, RFM can be said to possess a good predictive power and can explain a good portion of data variance and make good predictions.

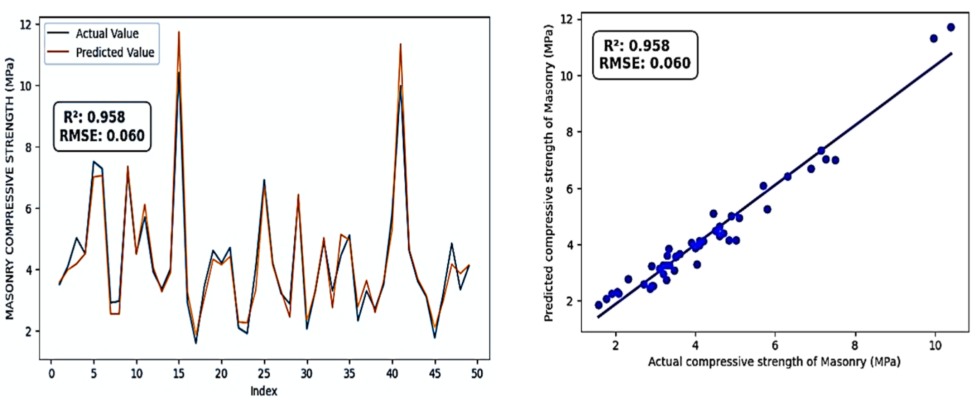

Fig. 3. Comparison of actual and predicted compressive strength (RFM).

4.5 Artificial neural network (ANN)

The ANN model had a "mapminmax" function that scaled the input features in the data sets to 0 and 1. Normalization makes effective learning possible and eradicates the dominant features that are based on the scale. Since the activation function was the purelin, the value of output was continuous and the activation function was the non-linear logsig that made the hidden layer to be non-linear. The best performing network had four hidden layers with each having fifteen neurons. The RMSE of ANN when fitted to the training set was 0.059 and the R2 was 0.906 and hence, it is observed that ANN fits the data and it predicts well. The validation set also indicated the generalization ability of ANN with a R^2 of 0.929 and a RMSE of 0.097. The ANN competency did not reduce to considerable extent despite the performance beginning to reduce marginally when implemented to the test set with an R^2 of 0.906 and RMSE of 0.088.The actual and estimated masonry compressive strengths of the test dataset are compared in Figure 4.a. The total R^2 of the ANN was 0.910 and RMSE = 0.044.

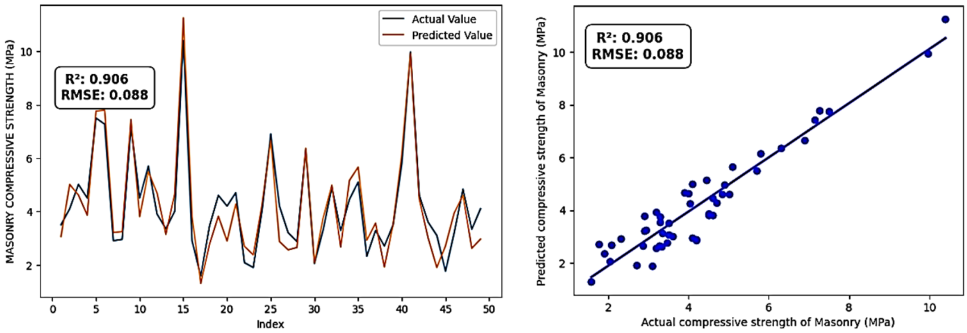

Fig. 4.a ANN model

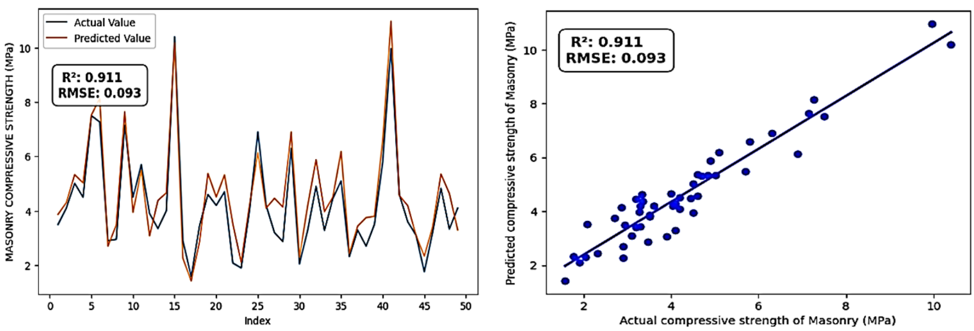

Fig. 4.b ANFIS model

Fig. 4. Comparison of actual and predicted compressive strength ANN and ANFIS.

4.6. ANFIS

The ANFIS model showed good learning and prediction accuracy with high R^2 values of 0.923 and 0.938 on training and validation sets respectively. It did underperform on the test set though not as much due to an R^2 value of 0.911. The test sets, training and validation sets had RMSE of 0.047, 0.09 and 0.09 respectively. The expected-observed masonry strength comparison of the testing dataset is presented in Figure 4.b with the R^2 being 0.923 and RMSE being 0.038 across the entire dataset indicating the model to be very predictive.

5 Conclusions

In this research, estimation of the strength of brick masonry was made using a range of machine learning models.

- ANN, PRM and ANFIS models give correct predictions at a fairly reasonable level, whereas LRM and ERM models are weak. The moderate and strong R2 values of the ANN and PRM models indicate that the two methods can identify underlying patterns.
- The models that had the highest R^2, namely DTM and RFM, have shown a good sense of capturing the relationship between variables and desired values.

- LRM, PRM, and ERM were found to have lower R^2 and higher RMSE 11-16% and 2.2-4.9 times, respectively, than DTM. The other models reflected improvement of RMSE between 10 and 90% and less than 10 variation in R^2.
- High R^2 values are also observed in SVR, which is their ability to effectively capture the complex relationships and their ability to work with new data.
- The generalization capability, low RMSE values, and good predictive capabilities all are good in overall the DTM, RFM, and SVR models.

Funding
No external sources of funding were provided for this research.

Conflicts of interest
The authors state that there are no conflicts of interest related to this work.

Authors contribution statement
Bhumika M R: Methodology, Original draft, Visualization, Software, Resources, Validation, Analyzed the key findings, review & editing; Madhavi K: Conceptualization, Formal analysis, Supervision; Shanta Rangaswamy: Software, Supervision; S M Basutkar: Analyzed the key findings. M V Renuka Devi: Review &Editing. All authors have read and agreed to the published version of the manuscript.

References

1. M. Mishra, Machine learning techniques for structural health monitoring of heritage buildings: a state-of-the-art review and case studies. *J. Cult. Herit.* **47**, 227–245 (2021). https://doi.org/10.1016/j.culher.2020.09.005
2. J.C. Morel, A. Pkla, P. Walker, Compressive strength testing of compressed earth blocks. *Constr. Build. Mater.* **21(2)**, 303–309 (2007). https://doi.org/10.1016/j.conbuildmat.2005.08.021
3. N.M. Mubarak, M. Anwar, S. Debnath, I. Sudin, *Fundamentals of biomaterials*. Springer Nature Singapore, Singapore (2023). https://doi.org/10.1007/978-981-19-9300-8
4. K.T. Nguyen, Q.D. Nguyen, T.A. Le, J. Shin, K. Lee, Analyzing the compressive strength of green fly ash based geopolymer concrete using experiment and machine learning approaches. *Constr. Build. Mater.* **247**, 118581 (2020). https://doi.org/10.1016/j.conbuildmat.2020.118581
5. F. Deng, Y. He, S. Zhou, Y. Yu, H. Cheng, X. Wu, Compressive strength prediction of recycled concrete based on deep learning. *Constr. Build. Mater.* **175**, 562–569 (2018). https://doi.org/10.1016/j.conbuildmat.2018.04.169
6. D.C. Feng, Z.-T. Liu, X.-D. Wang, Y. Chen, J.-Q. Chang, D.-F. Wei, Z.-M. Jiang, Machine learning based compressive strength prediction for concrete: an adaptive boosting approach. *Constr. Build. Mater.* **230**, 117000 (2020). https://doi.org/10.1016/j.conbuildmat.2019.117000
7. W.Z. Taffese, E. Sistonen, Machine learning for durability and service-life assessment of reinforced concrete structures: recent advances and future directions. *Autom. Constr.* **77**, 1–14 (2017). https://doi.org/10.1016/j.autcon.2017.01.016
8. M. Mishra, A.S. Bhatia, D. Maity, Predicting the compressive strength of unreinforced brick masonry using machine learning techniques validated on a case study of a museum through nondestructive testing. *J. Civ. Struct. Health Monit.* **10(3)**, 389–403 (2020). https://doi.org/10.1007/s13349-020-00391-7

9. J. Garzón-Roca, C. Obrer Marco, J.M. Adam, Compressive strength of masonry made of clay bricks and cement mortar: estimation based on neural networks and fuzzy logic. *Eng. Struct.* **48**, 21–27 (2013). https://doi.org/10.1016/j.engstruct.2012.09.029
10. Q. Zhou, F. Wang, F. Zhu, Estimation of compressive strength of hollow concrete masonry prisms using artificial neural networks and adaptive neuro-fuzzy inference systems. *Constr. Build. Mater.* **125**, 417–426 (2016). https://doi.org/10.1016/j.conbuildmat.2016.08.064
11. U. Reuter, A. Sultan, D.S. Reischl, A comparative study of machine learning approaches for modeling concrete failure surfaces. *Adv. Eng. Softw.* **116**, 67–79 (2018). https://doi.org/10.1016/j.advengsoft.2017.11.006.
12. Bhumika M R, M Tech thesis submitted to Department of Civil Engineering, RV College of Engineering,Bangalore.