International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. #, No.# (Mon-20..)

Enhancing Accuracy in Predicting Continuous Values through Regression

Ahmed Aljuboori[1,2*], M.M A Abdulrazzq[3]

[1]Computer Science Department, College of Education for Pure Science / Ibn Al-Haitham, University of Baghdad, Baghdad, Iraq.
[2]Department of Computer Science, Dijlah University College, Baghdad, Iraq
[3]Computer Science, Faulty of Innovation & Technology, Taylor's University, India
…

*E-mail address: *a.s.aljuboori@ihcoedu.uobaghdad.edu.iq, dr.obay@aic4all.com*

Received ## Mon. 20##, Revised ## Mon. 20##, Accepted ## Mon. 20##, Published ## Mon. 20##

Abstract: Enhancing the accuracy of predicting continuous values is essential in many fields. Regression is a practical approach in data mining, and machine learning can achieve this task. This study proposes a new framework of multiple regression models to obtain high accuracy using the Boston House Pricing Dataset (BHD). The examined models involve simple linear, multiple linear, Polynomial, Lasso, Ridge, Random Forest, Keras, and Gradient Boosting regression to seek a fair comparison with the best experimental result. The attempt is to select the best-predicting model using evaluation indicators such as R-squared Score (R2), Mean Squared Error (MSE) and Mean Absolute Error (MAE). Among the examined models, the first promising outcomes indicate that Random Forest and Ridge regressors scored a high level of R2 i.e. 89.9 and 88.3, respectively. In addition, The Gradient Boosting model offers the best result of R2 92 with MSE 0.72 and MAE 2.00. This research proposes two techniques to improve the accuracy of the best model. Re-sampling and optimization using the RandomizedSearchCV tuned hyper-parameter enhances the R2 score to 93.2 with a better MSE of 0.015 and MAE of 0.82. These findings prove a significant improvement in model performance and potential for practical application in real-world scenarios.

Keywords: Gradient Boosting, Keras, Lasso, Linear, Polynomial, Random Forest, Regression, Ridge.

1. INTRODUCTION

Regression algorithms are adopted in different fields, such as marketing, economics, finance, and healthcare. The accurate model is considered an effective tool for decision-making and data analysis. For many decades, improving the accuracy of regression in predicting continuous values has been a difficult task. The challenges would be related to either data or modelling issues. For instance, [1] claims that the ability to improve the accuracy in regression algorithms may sharply decrease if the number of training samples is minimal. Another challenge is that the problem of sampling imbalanced datasets could lead to low accuracy [2]. The second challenge lies in selecting a proper machine learning model (ML) that fits the dataset and achieves high accuracy. For instance, [3] tested regression approaches such as simple linear [4], polynomial [5], ridge [6] and lasso [7] to predict Boston house dataset BHD, but the best two results were Ridge and Lasso' Regression obtaining accuracy of R^2 (88.28 and 89.79) respectively. [8] has examined other different ML models such as XGBoost [9], support vector [10], Random Forest [11], multilayer perceptron and linear' Regression on the BHD. The Authors suffer from choosing a suitable model with high accuracy, i.e. R^2 of 0.920, 0.570, 0.860, 0.640 and 0.910. Although other researchers have progressed in predicting continuous value, they are limited to selecting the proper BHD model with the highest accuracy [12].

The current research proposes a new framework by experimenting with new approaches, i.e., deep learning using Keras [13] and enhanced Gradient Boosting [14]' algorithm. As the BHD has been challenging for many decades, the current research focuses on the proposed gradient-boosting model. Unlike traditional approaches, our model uses a hypermethod of re-sampling and optimization. Specifically, we introduce Tomek and SMOTE, which enhance the re-sampling of the training

E-mail:author's email

data. Additionally, our method optimizes the model, providing significant improvements using RandomizedSearchCV to determine the most optimal hyperparameter configuration.

Compared to the state-of-the-art [8] and as the best accuracy, the primary advantage of our method over existing ones is obtaining the highest accuracy using the best model on the BHD. Furthermore, the improved approach enhanced the performance of the gradient boosting algorithm, which addresses the shortcomings of classical ones.

This article is structured as follows: Section 1 presents the introduction. Section 2 states the literature review. Section 3 describes the methodology. Sections 4 and 5 present the results and discussion, respectively. Section 6 states the conclusion and future work.

2. RELATED WORK

Regression techniques have grown in machine learning and data mining fields because of their capacity to handle complex data patterns. In addition, BHD was a challenge for many researchers. Some researchers have used it to predict the price of housing, while others have utilized it to classify issues. Recent literature has shown many regression methods have been experimented on BHD. Some researchers have tried to fill the gap to choose the best model to perform a regression model in terms of accuracy as follows:

[3] Implemented simple linear regression, polynomial, lasso, and ridge regression on the BHD. The authors stated that Lasso regression outperformed simple linear regression in all evaluation metrics, with the highest accuracy. However, this can be computationally intensive and challenging because of cross-validation parameter tuning. Compared to this study, the prediction accuracy of the proposed method is higher, and dealing with the learning rate of hyper-parameters is better.

[9] Discussed the use of different regression models to predict BHD prices, highlighting the advantages of XGBoost in capturing complex relationships and the disadvantages of SVM in interpretability. The authors aimed to find the best model for predicting BHD prices. The models used were (linear regression, random forest, XGBoost, and SVM) on BHD price. XGBoost Regression was the best model for predicting BHD, offering high performance and scalability, while the SVM model was the worst. Although [9] achieved reasonable accuracy using XGBoost, [8] scored butter R^2 of XGBoost with low accuracy of SVM. Both resources are limited to hyper-parameter tuning or improving the classical models to increase the accuracy.

[15] Stated various ML techniques, explaining the random forest algorithm, highlighting the ability to capture nonlinear relationships on BHD. He claimed that Random forest was the best-performing regression model out of the tested ones. The authors limit to accommodate specific variables. On the other hand, [16] compares the effectiveness of ridge regression and random forest methods enhanced by genetic algorithms in predicting Boston home values and discovers that the latter performs better and has strong stability and reliability. In predicting Boston estate prices, the random forest model enhanced by a genetic algorithm was better than the ridge regression model. Although [17] showed good stability and exceeded ninety accuracy, it was limited to scoring low R^2 compared to [9]. [12] Use the models of (linear regression, random forest repressor, SVM repressor) to analyze BHD. The analysis included the model's fitness with R2, MAE, and MSE as evaluation indicators. The authors also claimed that random forest regression was the best method for predicting BHD, outperforming linear regression and SVM. [12] suffer from assembling forests, which may lead to more training time, while the random forest is not the best accuracy compared to the current research. The methodology of [16] involved using the random forest ML technique with BHD, data analysis, exploration, feature selection, preprocessing, and model development. The authors improved the random forest accuracy compared to [12] effectively with an error margin of ±5, highlighting that deep learning models can be explored for better house price prediction. This research has tested Keras deep learning and achieved better accuracy than [16].

The literature review needs more advanced regression methods to discover the best model with high accuracy applied to the BHD. Therefore, this research seeks to fill the gaps by selecting the best model, providing new experiment arguments, and proposing new procedures applied to the best model to improve the performance of the tested regression algorithms.

3. METHODOLOGY

This research compares eight regression techniques: simple linear, multiple, linear polynomial, lasso, ridge, random forest, Keras, and gradient boosting' regression. Boston housing prices dataset (BHD) is used as a benchmark for this research. BHD contains n = 506 observations with p = 14 features. This experimental research focuses on the algorithm that scores high accuracy of R^2 and on improving its performance using integrated procedures. The experiment started with linear regression, followed by other regressors to seek the best accuracy on the benchmarked BHD. The accuracy of the proposed framework is calculated through evaluation indicators R^2, MSE and MAE. The best model with high accuracy is then compared to the rest of the experimented models until the best results are reached. Finally, two techniques are applied to the best model, i.e., re-sampling and optimization, to improve the accuracy and fulfil the

aim of this research, as shown in Figure 1. Each model is discussed as follows.

A. Linear Regression

In simple linear regression, a linear relationship is established between the dependent variable *y* and a single independent variable *X*. This relationship is modelled by fitting a regression line represented by Eq. (1).

$$y = \beta_0 + \beta_1 X + \epsilon \quad (1)$$

β_0, β_1 refers to the vector of coefficients, and ϵ is the error term [4].

Nevertheless, it is crucial to recognize that the simple linear regression model's predictions may not always be precise. The limitation of the model is overcome by utilizing the error term ϵ. In this study, Linear regression was examined first as a baseline for the relationship between variables. It was considered the starting point to determine a high level of accuracy. Still, due to the limit of a single predictor, a further method is tested to obtain the best model with the highest accuracy.

B. Multiple Linear Regression

Multiple Linear is an extension of simple linear Regression [18]. It models the relationship between several independent variables ($X1$, $X2$, ..., X_p) and the dependent variable *y*. It considers several features of the dependent variable compared to ordinary linear regression, as the latter only considers one independent variable. Eq. (2) shows the form of the MLR model.

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \epsilon \quad (2)$$

β_0 represents the intercept, while β_1, β_2, …, β_p are coefficient of each predictor. and ϵ is the error term coefficient most used to resemble the data.

This algorithm has the potential to offer a more precise understanding of the correlation between each aspect and the result. It also showed a better relative accuracy when compared to simple linear regression because of the use of multiple predictors, but further experiments are needed to fulfil the aim of this research.

C. Polynomial Features and Feature Scaling

Polynomial regression enhances the original features by including additional variables of higher order [5]. To identify and extend the simple linear regression with only one feature, X^2, it is added as an extra feature to express the general form of this regressor, as shown in Eq. (3).

$$y = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon \quad (3)$$

X^2 represents the n-degree of the polynomial feature. Including these polynomial features enables the model to deal with curves, bends, and the impact on the data. In addition, it improves its ability to detect complicated patterns.

In this study, the polynomial model has shown better performance when compared to the first two algorithms because an appropriate feature scaling procedure is employed to ensure the stability and accuracy of the tested model. Further experiments will be conducted to seek the best model with the best accuracy applied to the BHD.

D. Lasso Regression

Lasso algorithm [7] eliminates a fundamental challenge in regression analysis, namely Overfitting. When a model becomes complicated by fitting noise, this could lead to poor generalization. The Lasso overcomes this issue by incorporating a penalty term into the linear regression equation, encouraging the model to select a subset of the most pertinent features while reducing the coefficients of less significant ones toward zero. In contrast, the simple linear regression aims to minimize the mean squared error (MSE) between predicted and actual *y* values. Lasso presents a regularization term but conducts a selection of variables by shrinking some coefficients to zero. The objective function is in Eq. (4).

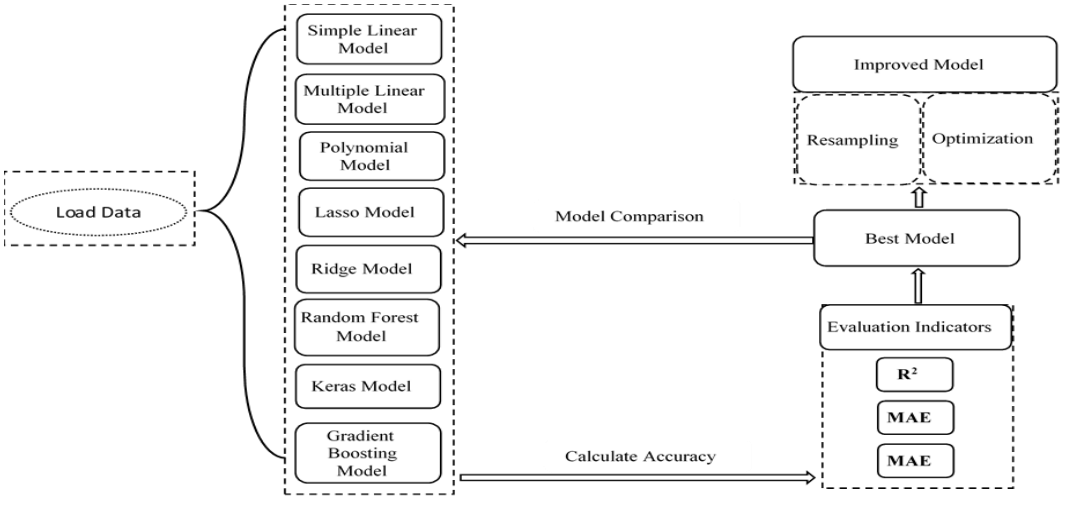

Figure 1 Model Framework

$$L(\beta) = \sum_{i=1}^{n}(y_i - X_i\beta)^2 + \sum_{j=1}^{p}|\beta j| \quad (4)$$

p is the number of predictors. λ is also known as *L1*, which refers to the regularization parameter controlling the shrinking degree. n is the number of observations. $|B_j|$ is the absolute value of the coefficient.

A drop in accuracy was noticed because the high λ resulted in an overfitting in the lasso model using BHD. In some experiments, however, under-fitting could occur because of missing significant features. Therefore, further experiments are required for better accuracy for the best model to be applied to the BHD.

E. Ridge Regression

Ridge regression modifies linear regression models by adding regularizing terms to stop the overfitting issues [6]. Because it reduces the influence of correlated features on coefficient estimates, it enhances the stability of the model and is especially helpful when handling multi-collinearity or highly correlated features.

Ridge regression can reduce the impact of less relevant features by reducing their coefficients closer to zero. It selects the optimal value for finding the *L2* regularization that balances model complexity and performance as described in Eq. (5).

$$L(\beta) = \sum_{i=1}^{n}(y_i - X_i\beta)^2 + \sum_{j=1}^{p}\beta_j^2 \quad (5)$$

Unlike L_1 regularization in Lasso Regression as a penalty term of the loss function, L_2, i.e. β^2_j term, reduces the coefficients while maintaining their inclusion in the model. Ridge regression lowers variance and increases model stability, especially with multi-collinearity. Ridge regression showed better accuracy than lasso but not the best in other experiments. Therefore, further research is required to fill the gap of stat-of-the-are.

F. Random Forest Regression

Random forest regression is an ensemble learning approach for regression applications [11]. It builds several decision trees during the training process. It produces the average prediction of all the individual trees to manage complicated datasets with high dimensionality and nonlinear correlations. It divides the feature space into areas recursively, giving each zone a constant value to reduce overfitting and de-correlating the different trees. The average of all the individual trees' forecasts makes up the prediction of a random forest regression model, as shown in Eq. (6).

$$\bar{Y} = \frac{1}{T}\sum_{t=1}^{T} h_t(X) \quad (6)$$

Where $\bar{Y}$ is the predicted value, T is the total number of trees. $H_t(X)$ is the prediction of the *t-th* tree.

In this research, random trees improved the predictive accuracy by controlling overfitting compared to previously examined approaches. This model has several benefits, such as better generalization, robustness to outliers, and parallelization training individual trees inside the forest.

G. Deep Learning with Keras Algorithm

This study uses the Keras library [13] to apply Neural Network regression. Keras's model usually includes one input layer with one or more hidden layers to incorporate the regression process. In the implemented Keras on BHD, medv was the target variable. The input layer contained 128 neurons, and the first input layer contained 64 neurons and ReLU activation. The model continues with a multilayer perceptron (MLP) design for one hidden layer followed by two hidden layers. This design lets the model learn complex, nonlinear relationships between the input features and the target variable. Values shown in Eq. (7).

$$\bar{Y} = f(x) = W_2 \cdot ReLU(W_1 \cdot x + b_1) + b_2 \quad (7)$$

W_1 *and* W_2 are the weights addressed to connections between layers. b_1 *and* b_2 are bias vectors that allow the model to fit better. ReLU stands for Rectified Linear Unit, which activates the functions applied to assist hidden to deal with non-linearity. The dropout is adjusted to (0.2) between the hidden layers. This mechanism encourages the model to not depend strongly on a particular feature during the training to promote generalization as deep learning. The applied Keras resulted in reasonable accuracy performance but indicated that it is not the best model to predict the continuous values of the BHD. Further, research is conducted to achieve the aim of this study.

H. Gradient Boosting Regression

Gradient boosting regression is a powerful ML method that has gained widespread popularity in predictive modelling. It handles complex relationships in data and produces highly accurate predictions [14]. This regressor is an ensemble learning method that improves predictions by successively fitting numerous weak learners. It uses decision trees to create an additive model, as shown in Eq. (8).

$$\bar{Y} = \sum_{m=1}^{M} \boldsymbol{\gamma}_m h_m\ (x_i) \quad (8)$$

$\bar{Y}$ represents the predicted values of the iteration *i-th*. M is the total number of the trees. $\boldsymbol{\gamma_m}$ is the weight applied to trees. $h_m(x_i)$ is the prediction of the trees for the required observation.

In this research, gradient-boosting regression scored with the best accuracy because weak learners were added one after the other; this reduced the residual errors from the previous step until a strong predictor was created. This made the model perform better compared to all experiments in this research. The learning rate of the shrinking technique prevents overfitting for further

enhancement. In addition, it helps in feature selection and model interpretability.

*I. Improving Gradient Boost Regressor (**Re-sampling & Optimization**)*

The experiments of this study have proved that the Gradient Boosting algorithm has achieved higher accuracy on the BHD when compared to the state-of-the-art. The proposed model suggests adding the re-sampling and optimization techniques to the classic gradient algorithm to improve the accuracy.

First, the SMOTETOMEK technique is applied to balance the sampling of the dataset. SMOTE produces adequate samples of the minority class, whereas TOMEK eliminates the nearest neighbours of the borderline to balance and clean the dataset, as shown in Eq. (9).

$$x_{new} = x_i + \delta.(x_{nn} - x_i) \quad (9)$$

x_{new} represents the new samples, while x_i is the minority class. x_{nn} as nearest neighbor subtracts the minority class of x_i. δ denotes the random number of distribution between zero and one.

Second, RandomizedSearchCV is conducted to tune the hyper-parameters. This technique samples a specific number of parameters randomly. It sets the specified distributions and assesses them through validation, as presented in Eq. (10).

$$\widehat{\Theta} = argmin_{\theta \epsilon \Theta} \frac{1}{k} \sum_{k=1}^{K} L\ (y^K, f(X^K,)) \quad (10)$$

Integrating the two methods, SMOTETOMEK and RandomizedSearchCV, into a gradient-boosting algorithm improved the performance. The former in Eq. (9), as a preprocessing to the dataset, enhances the quality of the training to obtain a reliable model. The latter, in Eq. (10), is a tuning parameter that guarantees the tuning of the model optimally for better generalization, which improves accuracy. The following steps describe the new proposed model applied to the BHD.

Step 1: Call libraries needed.

Step 2: Load the dataset.

Step 3: Make an optional Skewed Target Variable.

Step 4: Divide the dataset into training sets and testing sets.

Step 5: Re-sample training data using Tomek and SMOTE.

Step 6: Set the GradientBoostingRegressor model's initialization.

Step 7: Establish the RandomizedSearchCV parameter grid.

Step 8: Conduct a random search.

Step 9: Fit the model.

Step 10: Predict the evaluation set.

Step 11: Model evaluation.

Step 12: Cross-Validation

The above steps describe a regression analysis approach that uses hyper-parameter optimization and re-sampling to address the high accuracy of the BHD.

Step 1 is to import libraries for data manipulation, such as sci-kit-learn and data handling tools.

Step 2 loads the regression dataset for analysis to indicate that one class is noticeably underrepresented compared to the others.

Step 3 to develop a skewed target variable (for testing objectives). This step is mainly utilized in experiments by purposefully distorting the target of the variable's distribution.

Step 4 Dividing dataset into testing and training: next, a random split function is used to split the dataset into training and testing sets. This division ensures the model is tested on untested data (testing set) and trained on a representative portion of the data (training set).

Step 5 was tested but did not achieve the highest accuracy. It starts by re-sampling training data to rebuild for imbalance: the training data's class imbalance is addressed by applying the SMOTETomek re-sampling technique. This strategy involves two approaches: SMOTE (Synthetic Minority Oversampling Technique): by producing artificial data points for the minority class, this technique corrects the imbalance. A re-sampled training dataset with a more balanced class distribution is created by using SMOTETomek. This step may enhance the performance in unbalanced regression issues.

Step 6, the GradientBoostingRegressor model is instantiated. This model is widely preferred for regression problems because of its versatility and capability to handle nonlinear correlations between data and the target variable.

In Step 7, the RandomizedSearchCV parameter grid is created to optimize the model's performance by adjusting hyperparameters. In this phase, a grid is designed to determine each hyper-parameter's potential values that must be adjusted. This grid defines the boundaries of the search space for the optimization technique.

Step 8 incorporates RandomizedSearchCV to determine the most optimal hyper-parameter configuration. This method efficiently analyses the specified parameter grid by randomly selecting a subset of hyper-parameter combinations and assessing their efficacy. The technique identifies the combination that produces the optimal performance on a validation set, a subset of the training data utilized for adjusting hyper-parameters.

Step 9 involves fitting the model using the hyper-parameters determined by the RandomizedSearchCV algorithm on the re-sampled training data, if applicable.

In step 10, the model is applied to the previously unseen testing data from step 4 to provide predictions. This step allows the regressor to assess the model's ability to make accurate predictions.

In Step 11, the model's efficacy is assessed using metrics appropriate for regression tasks. R2 and (MSE, MAE) are examples of standard metrics. These metrics provide information on the accuracy and goodness-of-fit of the model by quantifying the gap between the predicted values and the actual target values.

Step 12. K-fold or stratified k-fold cross-validation is applied using several random data splits, and this technique iteratively repeats stages 4 through 11 of the process. Unlike a single split technique, each iteration offers an independent assessment of the model's performance, resulting in a more reliable and generalizable evaluation.

To improve the accuracy of the Gradient Boost Regressor, a re-sampling method of SMOTETomek is used in Step 5. It generates synthetic samples for the minority class and removes instances close to the majority class. The first high accuracy is reached 0.92.

The second fundamental part of the procedure in step 6 is using RandomizedSearchCV to carry out a thorough hyperparameter optimization. A predetermined grid of hyper-parameters, including the number of estimators, maximum depth, learning rate, subsample ratio, minimum samples needed for a split, minimum samples required for a leaf, and maximum features considered for a split, are searched across by this method. The search type can find the ideal hyper-parameters through hundreds of iterations to tune the model with the perfect configuration. The performance of this strategy is evaluated by calculating the (MSE), (MAE), and (R^2). This approach has shown a noticeable enhancement in predicting the accuracy, resulting in better accuracy.

J. Evaluation and Performance Metrics

This research uses three metrics to examine the best model performance. i.e. MSE, MAE and R2. The metrics used are used to evaluate the performance of the examined models. MSE calculates the average squared difference between observed and predicted values, which provides information on the variance of prediction errors. Conversely, MAE computes the average absolute distinctions between actual and predicted values, providing a simple description of prediction accuracy. R-squared measures the percentage of variance in the dependent variable compared to the independent ones. It is usually considered a model with a high R2 value) and low error metrics (a low MSE and MAE) should be viewed as a better performance when compared to other models.

4. RESULTS

The results in this section are based on the examination of different regressors. The attempt was to select the best model with the best accuracy in predicting continuous values.

The experiments that were conducted started with linear regression and were subsequently followed by a series of several regressors to achieve the most optimal model. Linear regression is evaluated as a baseline. The correlation of Random features is applied to the BHD, as shown in Figure 1. The outcomes in Table 1 show considerable scores in terms of R^2 (74.9), (MAE) (0.09), and (MSE) (0.02). It was noticed that there is a limit to using a linear approach when complex data is utilized. It became apparent that the linear model scores the undesirable performance of R^2. Therefore, it is necessary to experiment with other regressors.

Multiple Linear Regression (MLR) uses a broader selection of features is examined. This strategy enabled this research to provide a better score using diverse features, as shown in Figure 2. An enhancement in the model's performance was observed, R^2 is increased to 0.67. The model demonstrated improved predictive capabilities by encompassing a more comprehensive range of factors influencing the target variable.

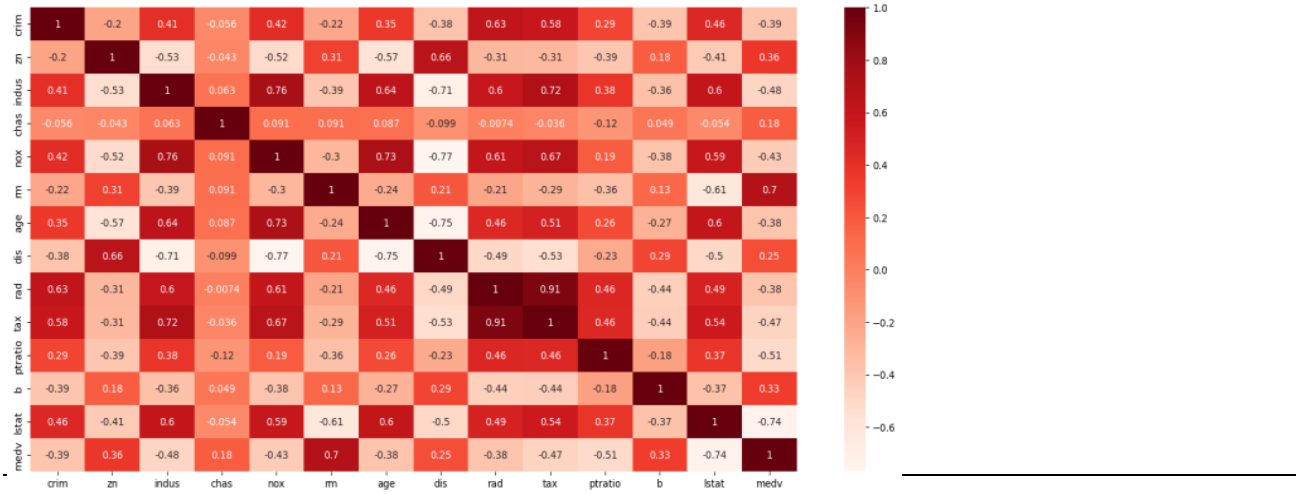

Figure 2 Correlation of Each Feature

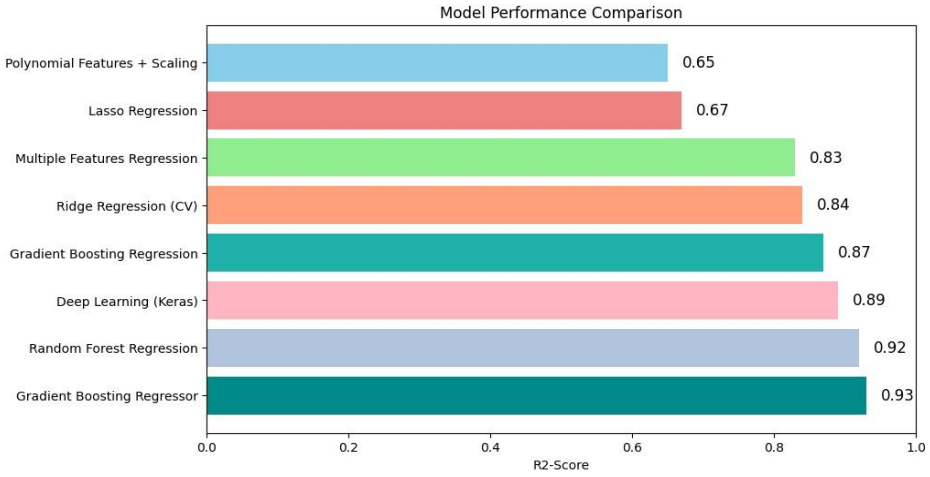

Figure 3 Model Performance

Polynomial Features are examined to Recognize the existence of nonlinear relationships in the dataset. This technique produced a substantial improvement in the model's predictive ability, concluding in a remarkable R^2 of 0.83. By incorporating Polynomial Features, the model gained the capacity to capture intricate data patterns beyond linear relationships. Concurrently, Feature Scaling guaranteed that each feature made a meaningful contribution to the predictions, thereby enhancing the overall effectiveness of the model.

Lasso Regression experiments were tested and led to a slight reduction in accuracy (R^2 = 0.65). This decline emphasized the challenge of striking the right balance between feature selection and model performance. While lasso's feature pruning capabilities were evident, an excessive reduction in feature dimensions may have led to the loss of essential information.

Ridge Regression is implemented with cross-validation, ending in an impressive R^2 of 88.3. This regularization approach introduced a reasonable mechanism for controlling complexity, thereby mitigating the model's susceptibility to overfitting. By incorporating cross-validation, the optimal selection of regularization parameters was ensured, refining the model's predictive performance and bolstering its ability to generalize well to unseen data.

Random Forest Regression is used to pursue enhanced predictive accuracy, and ensemble methods are embraced and implemented random forest regression, resulting in a notable R^2 of 89.9. This strategic approach highlighted the efficacy of amalgamating multiple decision trees to generate resilient predictions capable of capturing complex relationships within the data.

Deep learning with the Keras algorithm is examined. It offered an environment of deep learning, which could score the highest accuracy. While deep learning offered intricate complexity and scalability, its R^2 was 0.87. Nonetheless, the deep learning model presented challenging competition, showcasing its promising potential in regression tasks.

Gradient Boosting Regression was experimented with and reached first high accuracy with an impressive R^2 of 0.92, which increased to 0.932 When the optimization technique was used. This iterative ensemble method meticulously refined the best model, showcasing its adeptness in handling intricate regression tasks characterized by complex relationships within the data. The progressive refinement facilitated by gradient boosting regression underscores its efficacy in continuously enhancing predictive performance, solidifying its position as a formidable tool in regression analysis.

5. DISCUSSION

The performance of several regression models assessed by different authors' provides a better understanding of each model. For this evaluation, the (R^2), (MSE), and (MAE) metrics are used, as shown in Table 1.

[9] stated that XGBoost scores robust performance on the BHD, as demonstrated by its low MSE of 0.628, MAE of 2.936, and R^2 of 85.8%. With 82%, MSE of 0.81, and MAE of 1.348, Random Forest comes second with predictive solid accuracy. The superiority of ensemble methods in this situation is demonstrated by the poorer performance of linear regression and SVM, which noticeably have more errors.

[15] states that the Random Forest model scores an astounding 91.3%, with an MSE of 0.49 and an MAE of 1.115. The strong R^2 indicates excellent model fit, and the comparatively low error metrics further support its effectiveness in handling this dataset.

[17] claims that the Ridge Regression and GA-RF Model produce different outcomes for the Boston dataset. The performance of Ridge Regression is indicated by its R^2 of 69%, MSE of 17.882, and MAE of 2.793. With an of 91%, MSE of 3.599, and MAE of 1.196, on the other hand, the GA-RF Model performs remarkably well, highlighting the value of integrating genetic algorithms and random forests.

The outcomes in [12] highlight the benefits of ensemble methods once more. With an 86.41%, Random Forest performs noticeably better than SVM-Regressor 59% and Linear Regression 74.66%. The predictive accuracy of Random Forest is further demonstrated by its reduced MSE and MAE (2.55 and 0.94, respectively).

The only numbers provided by the [3] are those for Ridge Regression (88.28%) and Lasso Regression (88.79%), which are considered to be strong competitors. Linear and Polynomial Regression have comparable, if somewhat lower, values of 73.66% and 74.27%.

[16] argues that Random Forests of 90%, MSE of 0.702, and MAE of 1.900 confirm its reliable performance in many investigations For the BHD.

Table 1 Comparison of Regressors

Authors	Regression	R^2	MSE	MAE	Dataset
[9]	XGBoost	85.799520	0.628	2.936	Boston
	Random Forest	81.971735	0.81	1.348	
	Simple Linear	71.218184	3.090	19.074	
	SVM	59.001585	0.0001	0.009	
[15]	Random Forest	91.3	0.49	1.115	Boston
[17]	Ridge	69	17.882	2.793	Boston
	GA-RF Model	91	3.599	1.196	
[12]	Simple Linear	74.66	19.07	3.09	Boston
	Random Forest	86.41	2.55	0.94	
	SVM	59.00	26.95	2.94	
[3]	Simple Linear	73.66			Boston
	Polynomial	74.27			
	Ridge	88.28			
	Lasso	88.79			
[16]	Random Forest	90	0.702	1.900	Boston
[8]	Simple Linear	91	0.017	0.075	Boston
	Multilayer	64	0.066	0.179	
	Random Forest	86	0.025	0.112	
	SVM	57	0.079	0.211	
	XGBoost	92	0.015	0.84	
The Experiments of this Study	Simple Linear	74.9	0.09	0.02	Boston
	Multiple Linear	67	2.50	10.0	
	Polynomial	83	5.50	1.80	
	Lasso	65	10.5	2.61	
	Ridge	88.3	5.01	1.71	
	Random Forest	89.9	3.51	1.21	
	Keras	87	1.99	2.74	
	Gradient Boosting	92	0.72	2.00	
	Improved Gradient Boosting	93.2	0.015	0.82	

[8] states that the 91% and 92% scores, respectively, Linear Regression and XGBoost stand out among the several models this study examines using the Boston dataset. Their low MAEs (0.075 and 0.84) and MSEs (0.017 and 0.015) indicate their excellent predictive performance. Lower values are shown in Multilayer Perceptron, Random Forest, and SVM-Regressor, indicating less predictive accuracy.

The contrast in Table 1 shows that the results of R^2 of SVM are similar to those reported by the [9], [12] and [8]. Therefore, SVM was excluded from the proposed model because of the low level of R^2. [9], [12] and [3] share around 75% of R^2, similar to the proposed model. However, no significant increase was detected in the proposed model because the proposed model scored R^2 of 74.9. Further analysis shows that the R^2 in [9] and [12] are similar to those reported in [8]. The random forest reveals that (80-86) % of R^2 shows promising results. However, in [15] and [16], reassuring results encouraged the author in this research to obtain a reasonable level of $R^{2,}$ i.e. 89.9. Both results in [17] and [3] reveal that the R^2 in Ridge regression scores of (69 and 88.28) respectively, whereas this research registered an increase of 88.3 with better performance. No rise of R^2 was detected in this research of Lasso, while [3] obtained notable performance. However, the experiments of the Polynomial model of the current study show a better level of R^2 83 when compared to [3].

Compared to the current research results in [8] and [9], Gradient Boosting outperforms the score of R^2 in XGBoost, as shown in Table 1. The R^2 of 0.92 in the Gradient Boosting technique using **SMOTETomek** scored a competitive level of accuracy. With optimization technique, the best obtaining of the greatest R^2 0.932%, lowest MSE (0.015) and MAE (0.82). This demonstrates how well sophisticated boosting techniques work when combined with a re-sampling approach. These results provide the importance of the Gradient Boosting model over the tested model in the current suggested framework.

6. CONCLUSION

The ability of different regression approaches to accurately predict continuous values differs significantly, as can be seen by comparing them. Gradient Boosting performed better than all the other tested models, especially after optimization, with the lowest MSE of 0.015, the highest R2 score of 93.2, and the lowest MAE of 0.82. It shows how well Gradient Boosting handles complex data patterns and generates accurate predictions. Random Forest and Ridge Regression also showed an outstanding performance, demonstrating that these models are appropriate for tasks requiring high prediction accuracy. However, the effectiveness of Lasso Regression and Linear Regression was comparatively lower, highlighting the necessity for more advanced techniques in specific situations.

The findings highlight the necessity of selecting and optimizing appropriate regression algorithms to improve

the accuracy of continuous value predictions, providing functional visions for future research and application in various domains.

Further research will examine the Early Stopping approach in the training process to reduce the errors in the validation and prevent overfitting. In addition, more advanced regression methods will be experimented with using different datasets to enhance the accuracy.

ACKNOWLEDGEMENT

Any organization did not support this work.

REFERENCES

[1] B. Dou *et al.*, "Machine learning methods for small data challenges in molecular science," *Chem. Rev.*, vol. 123, no. 13, pp. 8736–8780, 2023.

[2] A. Kulkarni, D. Chong, and F. A. Batarseh, "5 - Foundations of data imbalance and solutions for a data democracy," F. A. Batarseh and R. B. T.-D. D. Yang, Eds. Academic Press, 2020, pp. 83–106. doi: https://doi.org/10.1016/B978-0-12-818366-3.00005-8.

[3] S. Sanyal, S. K. Biswas, D. Das, M. Chakraborty, and B. Purkayastha, "Boston house price prediction using regression models," in *2022 2nd International Conference on Intelligent Technologies (CONIT)*, 2022, pp. 1–6.

[4] G. James, D. Witten, T. Hastie, R. Tibshirani, and J. Taylor, "Linear regression," in *An introduction to statistical learning: With applications in python*, Springer, 2023, pp. 69–134.

[5] L. Tabelini, R. Berriel, T. M. Paixao, C. Badue, A. F. De Souza, and T. Oliveira-Santos, "Polylanenet: Lane estimation via deep polynomial regression," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 6150–6156.

[6] B. M. Kibria and A. F. Lukman, "A new ridge-type estimator for the linear regression model: Simulations and applications," *Scientifica (Cairo).*, vol. 2020, 2020.

[7] J. H. Lee, Z. Shi, and Z. Gao, "On LASSO for predictive regression," *J. Econom.*, vol. 229, no. 2, pp. 322–349, 2022.

[8] H. Sharma, H. Harsora, and B. Ogunleye, "An Optimal House Price Prediction Algorithm: XGBoost," *Analytics*, vol. 3, no. 1, pp. 30–45, 2024.

[9] Y. Chen, "Research on the Prediction of Boston House Price Based on Linear Regression , Random Rorest , Xgboost and SVM Models," vol. 21, pp. 27–37, 2023, [Online]. Available: https://drpress.org/ojs/index.php/HBEM/article/view/13600/13209

[10] Z. Liao, S. Dai, and T. Kuosmanen, "Convex support vector regression," *Eur. J. Oper. Res.*, vol. 313, no. 3, pp. 858–870, 2024, doi: https://doi.org/10.1016/j.ejor.2023.05.009.

[11] D. A. Zema, M. Parhizkar, P. A. Plaza-Alvarez, X. Xu, and M. E. Lucas-Borja, "Using random forest and multiple-regression models to predict changes in surface runoff and soil erosion after prescribed fire," *Model. Earth Syst. Environ.*, vol. 10, no. 1, pp. 1215–1228, 2024.

[12] Z. Li, "Boston House Price Prediction Based on Machine Leaning Methods," *BCP Bus. Manag.*, vol. 38, pp. 2883–2887, 2023, doi: 10.54691/bcpbm.v38i.4204.

[13] J. Ott, M. Pritchard, N. Best, E. Linstead, M. Curcic, and P. Baldi, "A Fortran-Keras Deep Learning Bridge for Scientific Computing," *Sci. Program.*, vol. 2020, p. 8888811, 2020, doi: 10.1155/2020/8888811.

[14] J. T. Vieira, R. B. D. Pereira, C. H. Lauro, L. C. Brandão, and J. R. Ferreira, "Multi-objective evolutionary optimization of extreme gradient boosting regression models of the internal turning of PEEK tubes," *Expert Syst. Appl.*, vol. 238, p. 122372, 2024.

[15] S. Sharma, D. Arora, G. Shankar, P. Sharma, and V. Motwani, "House Price Prediction using Machine Learning Algorithm," in *2023 7th International Conference on Computing Methodologies and Communication (ICCMC)*, 2023, pp. 982–986. doi: 10.1109/ICCMC56507.2023.10084197.

[16] A. B. Adetunji, O. N. Akande, F. A. Ajala, O. Oyewo, Y. F. Akande, and G. Oluwadara, "House price prediction using random forest machine learning technique," *Procedia Comput. Sci.*, vol. 199, pp. 806–813, 2022.

[17] L. Ye, "Comparison of Ridge Regression and GA-RF Models for Boston House Price Prediction," *Int. J. Math. Syst. Sci.*, vol. 6, no. 4, 2023.

[18] Q. Zhang, "Housing Price Prediction Based on Multiple Linear Regression," *Sci. Program.*, vol. 2021, p. 7678931, 2021, doi: 10.1155/2021/7678931.

Dr. Ahmed Aljuboori is a Lecturer in the Computer Science Department, College of Education for Pure Science / Ibn Al-Haitham, University of Baghdad, currently on secondment to Dijlah University College. Dr Aljuboori received his PhD in Data Mining from the University of Salford. Their research interests include [Data Mining, Data Science and Machine Learning. Dr. Aljuboori has published extensively in [Case-Based Reasoning, Fuzzy Logic, Classification and Security]. Please visit his Scopus profile for more information about Dr Aljuboori 's work,

Dr. M. M. A. Abdulrazzaq is an Adjunct Associate Professor in the Computer Science Faculty of Innovation & Technology at Taylor's University, India. He received his PhD in AI from Universiti Kebangsaan Malaysia. His interests include artificial intelligence engineering, data science, machine learning, and big data. For more information, please visit https://www.linkedin.com/in/drobay/.