World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(RESEARCH ARTICLE)

Check for updates

Gradient boosting regression approach for housing unit price prediction

Paul Boye [1,*] and Cynthia Borkai Boye [2]

[1] *Department of Mathematical Sciences, Faculty of Computing and Mathematical Sciences, University of Mines and Technology, Tarkwa, Ghana.*
[2] *Department of Geomatic Engineering, Faculty of Geosciences and Environmental Studies, University of Mines and Technology, Tarkwa, Ghana.*

World Journal of Advanced Research and Reviews, 2025, 26(03), 1393-1404

Publication history: Received on 04 May 2025; revised on 07 June 2025; accepted on 09 June 2025

Article DOI: https://doi.org/10.30574/wjarr.2025.26.3.2302

Abstract

To purchase a house is one of the biggest financial goals for everyone. However, accurate and prompt housing unit price (HUP) prediction is crucial for both the real estate industry and investors. This study proposes a HUP prediction model based on gradient boosting regression (GBR). The proposed GBR model was compared with the following investigating methods: adaptive boosting (AdaBoost), k-nearest neighbour (KNN), decision tree (DT), random forest (RF), and support vector machine (SVM). The proposed GBR method demonstrated superior predictive performance over five state-of-the-art methods (AbaBoost, KNN, DT, RF, and SVM) when evaluated using a real dataset. This was obvious from the mean absolute percentage error (MAPE), coefficient of determination (R2), correlation coefficient (R), and coefficient of variance root mean square error (CVRMSE) employed as model assessment metrics. The results revealed that the GBR had the lowest MAPE (0.017%), CVRMSE (1.968%), and highest R2 (0.993) and R (0.99649) values as compared with the other investigated methods. This confirms the proposed GBR method's strength for reliable and efficient HUP prediction.

Keywords: Real Estate; Artificial Intelligence; Gradient Boosting Regression; Housing Unit Price; Prediction

1. Introduction

The housing market is an essential component of any viable economy. In many countries, owning one's piece of real estate is regarded as a symbol of social standing, and achieving this status is a goal for many people just starting life. On the other hand, investors are driven to the housing market because they regard property as an investment opportunity rather than a mere commodity (Alzain *et* al., 2022; Soltani *et* al., 2022).

Accurate and timely housing unit price (HUP) predictions are important to prospective house owners, developers, investors, and other real estate market participants. Undoubtedly, house price trends are significant because they influence people's decisions on real estate investment (Terregrossa and Ibadi, 2021; Xu and Zhang, 2021).

When entering the housing market, prospective house owners and investors alike do so with the expectation of making a profit from further price increases. As it is known, property value is directly related to the house ownership rate in developing countries, especially. Nonetheless, in the housing market, the main element determining sustainability is affordability. Moreover, as an investment strategy, whether housing is cost-effective also impacts the sustainability of its usage (Alzain *et* al., 2022).

In the real estate sector, which has become a highly profitable investment source, profitability is determined by housing prices. Recently, the determination of housing prices has become one of the most important topics in the sector. Such a

* Corresponding author: Paul Boye

topic has prompted many market players, from residential investors to real estate investment trusts and from individual investors to government officials, to predict the movement of housing prices, with these players adopting various methods to achieve that (Alzain *et* al., 2022).

In the literature, the wide variance between the listing and sales of house prices has led to the inference that real estate prediction is imprecise because the methods employed often fail to capture the complexity and the variety of the assets in the real estate market. These inaccuracies are due to the result of inadequacies in the conventional prediction methods employed. In addition, the lack of consistency and accuracy of the traditional methods of prediction is a matter of utmost concern. Therefore, this calls for the need for either a paradigm movement from the conventional prediction practices or a reinforcement of the methods used. Hence, the inaccuracy of the traditional methods has led to a concerted effort towards enhancing the traditional prediction methods through the utilisation of Artificial Intelligence (AI) techniques in prediction, which have been affirmed to be unique as they are paradigmatic techniques with an incredible predictive ability. The increased prediction accuracy and efficiency provided by the AI methods enable a more efficient allocation of resources within the housing market, further enhancing its overall success. Moreover, the application of AI methods in HUP prediction can aid the real estate market in effectively adapting to market fluctuations, consequently, fostering its growth (Yakub *et* al., 2020; Rampini and Re Cecconi, 2022; Neves *et* al., 2024; Yang *et* al., 2023).

In recent years, fostered by the increasing amount of data availability and the advancement in Information Technology, AI techniques have been widely employed to solve complex and nonlinear problems. The advantage of using these systems is that a large number of accurate predictions can be performed promptly and at a lower cost (Rampini and Re Cecconi, 2022; Mora-Garcia *et* al., 2022). Therefore, the AI methodology has been applied extensively in many fields with promising results (Štubňová *et* al., 2020).

In the literature, it has been revealed that AI methods are efficient for accurately predicting HUP in the housing industry (Štubňová *et* al., 2020). As a result, this study intends to employ the following AI methods for HUP prediction: Adaptive Boosting (AdaBoost), K-Nearest Neighbour (KNN), Decision Tree (DT), Random Forest (RF), Gradient Boosting (GB), and Support Vector Machine (SVM). Sharma and Gill (2024) developed seven machine learning models (linear regression, extreme gradient boosting (XGBoost), AdaBoost, Categorical Boosting (CatBoost), RF, KNN regressor, and multilayer perceptron (MLP)) for HUP prediction. Among the models, RF emerged with impressive accuracy. In a similar development, Tandon (2024) developed an AI model using RF, SVM, Light Gradient Boosting Machine (LightGBM), and Gradient Boosting Regressor (GBR) to predict HUP. Results showed that the GBR model was the best with the lowest mean square error (MSE), root mean square error (RMSE), and the highest R-squared (R^2) value compared to the other investigated models. Tchuente and Nyawa (2022) also developed seven machine-learning models for HUP prediction and compared their prediction capabilities. The experimental results revealed that neural network and RF models outperformed the other models when geocoding features were not accounted for, while RF, AdaBoost, and GBoost performed well when geocoding features were considered. Reddy and Chandra (2023), on the other hand, developed DT regression and RF regression models for HUP prediction. DT showed 71.63% accuracy in its prediction than RF, with a 62.11% prediction accuracy. Li *et* al. (2009) in another development created a Support Vector Regression (SVR) model and compared it with a backpropagation neural network (BPNN) model for HUP prediction. Analytical results showed SVR superiority over the BPNN based on its lowest mean absolute percentage error (MAPE) and RMSE values.

Because of the versatility and mathematical attractiveness of the AI methods in the reviewed literature relating to this study, it is extremely necessary to explore beyond the methods' application boundaries and assess their predictive strengths in the housing industry. Therefore, this study applied the following AI methods: AdaBoost, KNN, DT, RF, GB, and SVM to harness the power of AI for the prediction of HUP. Thus, this study fills a research gap by way of creating awareness and collaborating on the implications of using AI as a computational tool to solve problems in the housing industry. The study, which is worth a scientific investigation in the Ghanaian setting, can also provide reliable guidance to legislators and researchers. The methods mentioned were tested on a dataset obtained from Regimanuel Gray Estates Ltd., an estate developer in Ghana, West Africa. In the end, it was revealed that the GB regression is robust and superior to the other investigating methods and thus could be useful to legislators, managers, and investors in the housing industry to make knowledgeable decisions.

2. Methods

2.1. Adaptive Boosting (AdaBoost)

The Adaboost method was proposed by Freund and Schapire in 1997 (Dargahi-Zarandi et al., 2020). The characteristic of AdaBoost is to use the initial training data to generate a weak learner, and then adjust the training data distribution according to the prediction performance for the next round of weak learner training (Feng et al., 2020). Reweighted

data in this method, in which the weights are chosen based on the quality of the learner's performance, are considered sequentially in the weak learners (Dargahi-Zarandi et al., 2020).

The first of the shortcomings of the AdaBoost method is that the sample weight cannot be adjusted so that weak learners can learn more pertinently. Secondly, multiple weak learners cannot be integrated into a stronger learner through a boosting algorithm. The method has greatly enhanced the deficiencies of the two aspects, and its main idea is to train the weak regression with different prediction accuracies for the same set of training data many times and fuse multiple weak regressions to form a stronger regression with higher accuracy. Thus, it must be seen that the algorithm mainly changes the distribution of the dataset, and the computation results reveal that the weight of each weak regression is adaptively adjusted. Consequently, in large error samples in the weak regression, the weight of each is increased. Hence, new data updated by weight is input to the later weak regression training, and finally, the weak and dry weak regressions are merged (Li et al., 2022; Sai et al., 2023).

Considering the general regression, the training dataset is shown in Equation (1) (Feng et al., 2020).

$$\Phi = (X_i, Y_i) \quad \text{.....} \quad (1)$$

where Xi is the input data vector, and Yi is the output data value. $i = 1, \ldots, m$ is the ith sample in the training dataset, and m is the total number of samples. This is used to train a weak (base) learner G(X) using a specific learning algorithm, and the prediction error is shown in Equation (2).

$$e_i = \frac{\left| Y_i - G\left(\mathbf{X}_i\right) \right|}{E} \quad \text{.....(2)}$$

where $E = \max \left| Y_i - G\left(\mathbf{X}_i\right) \right|$ is the maximum absolute predicting error of all the samples.

AdaBoost produces a series of weak learners Gk(X), $k = 1, \ldots, N$ and combines them to construct a strong learner H(X) by a regression technique shown in Equation (3).

$$\text{H(X)} = \nu \sum_{k=1}^{N} \left(\ln \frac{1}{\alpha_k} \right) g\left(\mathbf{X}\right) \quad \text{……..(3)}$$

where α_k is the weight of the weak learner Gk(X), g(X) is the median of all the α_k Gk(X), $\nu \in [0,1]$ is the regularisation factor (learning rate) used to avoid overfitting issues.

The weak learner Gk(X) and its weight α_k is produced using the modified versions of the original training data, which is achieved by a reweighting approach, namely, adjusting the distribution weights of each sample according to the predicting error by the previous weak learner Gk-1(X). The mis-predicted samples will have their weight increased such that they will be more concentrated in the next training process. This is eventually an iterative process. For iteration $k = 1, \ldots, N$, the weak learner is Gk(X) and the relative predicting error e_{ki} is computed according to Equation (2), thus, the total error ratio ek of this step is expressed as

$$e_k = \sum_{i=1}^{m} e_{ki} \quad \text{.....(4)}$$

Therefore, the weight of the weaker learner is

$$\alpha_k = \frac{e_k}{1 - e_k} \quad \text{.....(5)}$$

The distribution weight of each sample for the next step of training is updated as

$$w_{k+1,i} = \frac{w_{k,i}\alpha_k^{1-e_{ki}}}{\sum_{i=1}^{m} w_{k,i}\alpha_k^{1-e_{ki}}} \quad \text{.....(6)}$$

It must be noted that there are two kinds of weight in the above derivations, $w_{k,i}$ and α_k. The first one ($w_{k,i}$) is in the sense of training data samples, which indicates that the mis-predicted samples will have their weights increased then they can be better learned in the next step. The second one (α_k) is in the sense of the weaker learners, which indicates that the more accurate weak learner will have a large influence on the final results.

2.2. K-Nearest Neighbour (kNN)

The kNN is a non-parametric method that operates in a vector space, and it is mostly employed in supervised learning methodologies in machine learning (Gbashi et al., 2024). The kNN operates on the principle that data points that are close to each other in feature space tend to share similar properties. The method predicts the output for an unknown data point by considering the majority class among its k-nearest neighbours from the training set, determined using a specific distance metric such as Euclidean, Manhattan, Chebyshev, or Minkowski. As a nonparametric method, kNN does not assume any specific distribution for the data, which enhances its robustness in dealing with noisy or incomplete observations. kNN is particularly effective for machine learning-based forecasting as it can identify influential patterns within noisy datasets. For continuous data, kNN matches data points based on computed distances to determine similarity, which directly influences its accuracy and performance. The algorithm involves the following two main steps: first, it computes the distance between the target data point and all points in the training set to find the closest neighbours. The Minkowski distance serves as the most comprehensive form of distance measurement. Secondly, it classifies the target data point based on the majority label of these neighbours or predicts a value based on their averaged output in the case of regression (Someetheram et al., 2025; Çetin and Büyüklü, 2025; Sai et al., 2023).

The two most critical hyperparameters that significantly impact the accuracy of the estimation and the model's performance of the kNN method are the distance metric and the optimum number of neighbours (k-value). The distance between two points (x_1, y_1) and (x_2, y_2) as shown in Equation (7) (Someetheram et al., 2025) is defined in the Euclidean space.

$$d = \sqrt{(x_1 - x_2)^2 - (y_1 - y_2)^2} \quad \text{.....(7)}$$

2.3. Decision Tree (DT)

A DT is a nonparametric supervised learning method that predicts a target value of a variable through simple decision rules inferred from measured training data and their features (San Millan-Castillo et al., 2019). The regression DT algorithm is an effective algorithm in machine learning that can be used to solve regression tasks. In decision analysis, the methodology can be employed to explicitly and visually show both decisions and decision-making. The principal objective of using the algorithm is to create a training model that can be used to forecast the value of the target variable with the help of learning modest judgment principles from the training data. As the DT name suggests, it has a simple tree-like structure of decisions as shown in Figure 1. In a DT, each node represents a conditional statement, and the branches of it show the outcome of the statement shown by the nodes. The algorithm iterates from the root node to the leaf nodes. The splitting process is then applied to each of the new branches. After executing all attributes in the above nodes, the leaf node shows the decision formed. The process continues until each node reaches a user-specified minimum node size and becomes a terminal node (Singh et al., 2021; Chen and He, 2018).

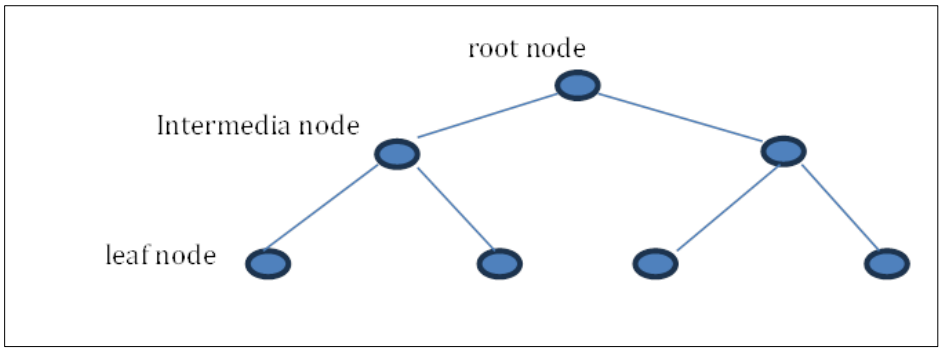

Figure 1 Schematic Diagram of Decision Tree

2.4. Random Forest (RF)

RF is the best and most adaptable supervised machine learning algorithm in the concept of ensemble technique and hybrid model for improving performance and prediction accuracy (Sai et al., 2023). Regression RF is an ensemble learning algorithm that creates many regression tree models built from bootstrap samples of the training dataset (see Figure 2). The methodology injects randomness into the tree-growing process by randomly selecting only a subset of predictor variables to consider for split-point selection at each node. This operation reduces the chance of the same strong predictor variables being selected when a split is to be carried out, thus preventing regression trees from becoming overly correlated (Johansson et al., 2014; Everingham et al., 2016). The multiple regression tree predictors are joined together to reduce the prediction variance and increase prediction accuracy. The method predicts the average value of all individual regression tree predictors (Čeh et al., 2018).

Similar to most machine learning methods, regression random forest has some free parameters that need to be optimised. There are, among others, the number of predictor variables randomly selected at each node, the number of trees, the minimum number of observations in a regression tree's terminal node, and the proportion of observations to sample in each regression tree. These free parameters are optimised through cross-validation. In practice, it is not necessary to fine-tune the number of decision trees. However, it is generally suggested to set it to a large number, allowing the convergence of the prediction error to a stable minimum (Zhang and Haghani, 2015; Bentéjac et al., 2021).

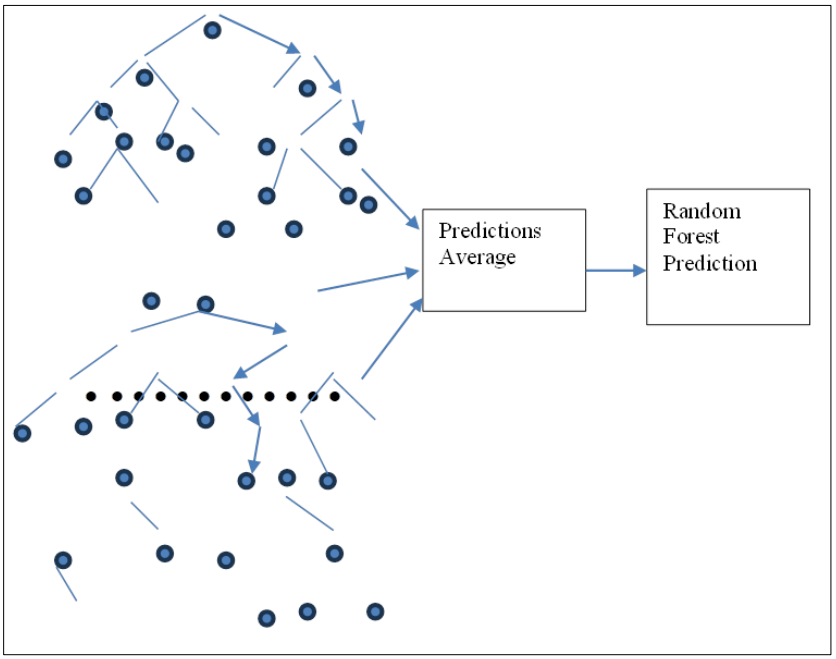

Figure 2 Schematic Diagram of Regression Tree Node

Let $\left\{\overline{\mathbf{Y}}_b(\mathbf{s})|\mathbf{s}\in D\right\}_{b=1}^{B}$ be the ensemble of regression tree predictors resulting from the training of the traditional regression random forest, where $\left\{\mathbf{Y}(s)|s\in D\subset R^d\right\}$ be a real-valued response variable defined on a spatial domain of interest D, and B is a real value. At this stage, individual regression tree predictors and their averages do not necessarily match the response variable's measured values at sampled locations. The next steps consist of using this ensemble of regression tree predictors $\left\{\overline{\mathbf{Y}}_b(\mathbf{s})|\mathbf{s}\in D\right\}_{b=1}^{B}$ to reconstruct individual regression tree predictors that perfectly match the response variable's observed values at sampled locations. By construction, their average also exactly fits the response variable's observed values at sampled locations (Fouedjio, 2020).

2.5. Gradient Boosting Regression (GBR)

The GBR methodology, which was first proposed by Friedman in 2001, was posed as a gradient descent method in which each step consists of fitting a nonparametric model to the residues of a previous model (Andrade et al., 2023). The algorithm involves an ensemble learning approach where robust forecasting models are formed by integrating several individual regression decision trees that are referred to as weak learners. Such an algorithm reduces the error rate of weakly learned models. Weakly learned models are those that have a high bias regarding the training dataset, with low variance and regularisation, and whose outputs are considered only somewhat improved when compared with arbitrary guesses (Singh et al., 2021). In this technique, the regression is dependent on the residuals of the previous iteration, where the impact of each feature is evaluated sequentially until a target accuracy is obtained. Consider a training dataset $D=\{(\mathbf{x}_i,y\ )|i=1,2,...,T\}$. The GBR goal is to find an approximation $G(\mathbf{x})$ of the function $G^*(\mathbf{x})$ which maps x to their output values y by minimising the expected value of a given loss function $\tau(y,G(\mathbf{x}))$. The residuals computed by the Loss function are optimised using the gradient descent method. The final result is obtained by the summation of the results of the T sequential regression functions g_t as shown in Equation (8) (Upadhyay et al., 2020; Zhang and Haghani, 2015).

$$y=\sum_{t=1}^{T}\beta_t g_t\left(\tau\left(y,G(\mathbf{x})\right)\right);\ \ \mathrm{g}_t\in G \qquad \text{.....(8)}$$

where G is a multilayer perceptron space, g_t is a decision tree (regression tree), β_t is the step size (boost rate), and T is the total number of iterations in the boosting algorithm. The ensemble of trees is constructed sequentially by estimating the new decision tree $g_{t+1}(\mathbf{x})$ with the help of Equation (9).

$$\arg\min\sum_{T}\tau\left(y_t.G_T(\mathbf{x})+G_{T+1}(\mathbf{x})\right) \qquad \text{.....(9)}$$

2.6. Support Vector Machine (SVM)

SVM is a machine learning algorithm based on statistical learning theory and the principal structural risk minimisation inductive principle presented by Vapnik and Cortes in 1995. Figure 3 shows the network structure of SVM. The main idea of the methodology is to transform the nonlinear input area into an area with high-dimensional properties to find a hyperplane via nonlinear mapping, and the procedure seeks to minimise an upper bound of the generalisation error consisting of the sum of the training error and confidence level (Zendehboudi et al., 2018). Based on such an induction principle, SVM usually achieves higher generalisation performance than the traditional neural networks that implement the empirical risk minimisation principle in solving many machine learning problems. Another key characteristic of SVM is that training SVM is equivalent to solving a linearly constrained quadratic programming problem, so that the solution of SVM is always unique and globally optimal, unlike other network training, which requires nonlinear optimisation with the danger of getting stuck in a local minimum. In SVM, the solution to the problem is only dependent on a subset of training data points, which are referred to as support vectors. Using only support vectors, the same solution can be obtained using all the training data points (Dong et al., 2005; Radhika and Shashi, 2009).

Support vector regression is the SVM utilisation for function approximation and regression. Different basic kernel functions are used in SVM models. The functions can be classified as polynomial, exponential radial basis function, radial

basis function, sigmoid, and linear. A training dataset of input-output pairs is given as Equation (10) (Zendehboudi et al., 2018).

$$P = \left\{ \left(\mathbf{X}_i, Y_i \middle| i = 1, 2, \ldots, n \right) \right\} \quad \text{.....(10)}$$

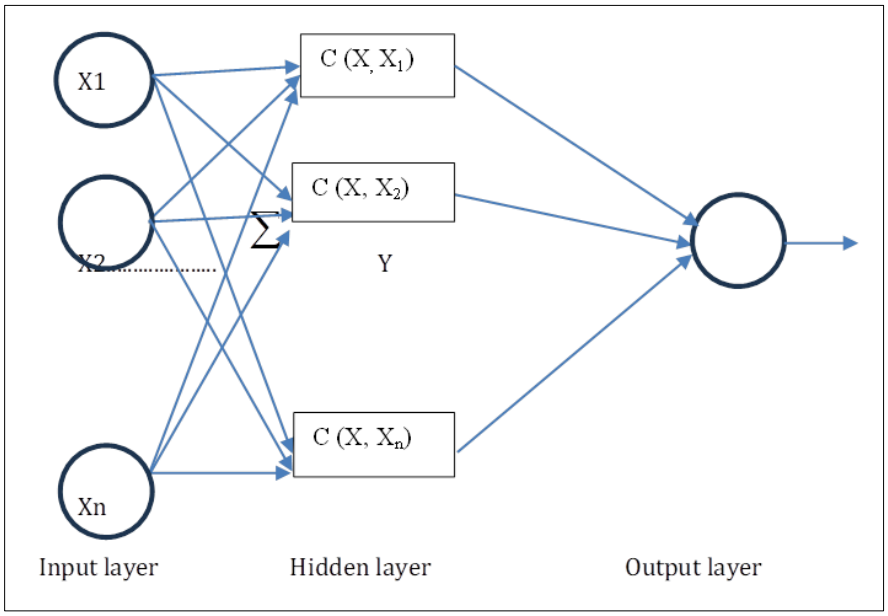

Figure 3 Schematic Diagram of SVM

where $\mathbf{X}_i \in R^{\tau}$, τ is the dimensional input vector, $Y_i \in R$ is the corresponding target, and n is the training data set size. The regression model is constructed as shown in Equation (11).

$$Y = W^T \phi(\mathbf{X}) + B \quad \text{.....(11)}$$

where W is the weight vector, B is the basis term, and $\phi(\mathbf{X})$ is a representative nonlinear mapping function, which maps X into higher dimensional feature space.

To obtain W, the regularisation function must be minimised, as shown in Equation (12), with the constraint shown in Equation (13).

$$\min \left\{ \frac{1}{2} W^2 + k \sum \left(\varphi_i + \varphi_i^{(*)} \right) \right\} \quad \text{.....(12)}$$

$$Y_i - \left\{ \left(W^T \phi(\mathbf{X}_i) \right) + B \le \psi + \varphi_i \right\}; \quad i = 1, 2, \ldots, n. \quad \text{.....(13)}$$

$$\varphi_i, \varphi_i^{(*)} \ge 0; \quad i = 1, 2, \ldots, n. \quad \text{.....(14)}$$

where ψ is equivalent to the function approximation accuracy placed on the training data samples, $\varphi_i^{(*)}$ and φ_i represent the positive slack variables, and k is the penalisation parameter of the error that is applied to control the trade-off between the regularisation term and empirical risk. The support vector regression is solved by introducing the Lagrange multiplier, ρ_i and ρ_i^*, and exploiting the constraints as shown in Equation (15).

$$f(\mathbf{X}) = \sum_{i=1}^{n} \left(\rho_i - \rho_i^* \right) C(\mathbf{X}, \mathbf{X}_i) + B \quad \text{.....(15)}$$

3. Model Performance Assessment

Four statistical metrics were used to assess the performance of the prediction models developed, as shown in Equations (16) to (20). The metrics used are: Mean Absolute Percentage Error (MAPE), Coefficient of Determination (R2), Correlation Coefficient (R), and Coefficient of Variance Root Mean Square Error (CVRMSE) (Dong et al., 2005; Someetheram et al., 2025; Lin et al., 2006; Feng et al., 2020).

3.1. Mean Absolute Percentage Error (MAPE)

This metric provides prediction accuracy measures as a percentage. A lower value of MAPE is an indication of accurate predictions compared to the actual values. Hence, the developed model's accuracy is high.

$$MAPE = \sum_{i=1}^{\tau} \left| \frac{A_i - B_i}{A_i} \right| \times \frac{100\%}{\tau} \quad \text{.....(16)}$$

3.2. Coefficient of Determination (R2)

R2 measures the proportion of the variance explained by the developed model. An R2 value close to 1 is an indication of how close the predicted values are to the actual values because the model captured the patterns in the dataset very well. It can further be said that the developed model explained most of the variations in the HUP data, while a value close to 0 shows how the model failed to explain most of the variability.

$$R^2 = 1 - \frac{\sum_{i=1}^{\tau} \left(A_i - B_i \right)^2}{\sum_{i=1}^{\tau} \left(A_i - \overline{A_i} \right)^2} \quad \text{.....(17)}$$

3.3. Correlation Coefficient (R)

The R-value is a measure of the strength of the developed model or linear relationship between two variables. Here, the metric explains the correlation between the HUP and the predicted HUP. The metric value lies between 0 and 1. The model is said to have maximised the goodness-of-fit if the R-value is close to 1, else the model is said to fit the data poorly.

$$R = \frac{\frac{1}{\tau} \sum_{i=1}^{\tau} \left(A_i - \overline{A} \right) \left(B_i - \overline{B} \right)}{\left(\sqrt{\frac{1}{\tau} \sum_{i=1}^{\tau} \left(A_i - \overline{A} \right)^2} \right) \left(\sqrt{\frac{1}{\tau} \sum_{i=1}^{\tau} \left(B_i - \overline{B} \right)^2} \right)} \quad \text{.....(18)}$$

3.4 Coefficient of Variance Root Mean Square Error (CVRMSE)

The CVRMSE is derived by normalising the RMSE with the mean of the data. As a nondimensional measure, it has the advantage of being expressed as a unitless and it is usually presented as a percentage. A developed model is considered accurate as the metric gets closer to 0%.

$$CVRMSE = \frac{RMSE}{\overline{A}} \times 100\% \quad \text{.....(19)}$$

where $$RMSE = \sqrt{\frac{\sum_{i=1}^{\tau}(A_i - B_i)^2}{\tau}} \quad \text{.....(20)}$$

where Ai and Bi are the ith actual and predicted HUP respectively, $\overline{A}$ and $\overline{B}$ are the averages of the actual and predicted HUP respectively, and τ is the sample size.

4. Numerical Application

4.1. Data Used

A 15-year dataset for a one-bedroom housing unit from Regimanuel Gray Estates Ltd., an estate developer in Ghana, West Africa, was used for the model development. The fifteen-year dataset for a one-bedroom housing unit from Regimanuel Gray Estates Ltd., an estate development corporation in Ghana, West Africa, was used for the model development. The dataset (cement, sand, iron rods, roofing, paint, and wood), which spans from 2003 to 2017, was used as the independent variables, and the half-yearly HUP served as the dependent variable. In total, 30 observations were applied. Table 1 shows the statistical summary of the dataset used for the model development.

Table 1 Statistical Summary of One-Bedroom HUP Dataset

Parameter	Mean Value ($)	Median Value ($)	Minimum Value ($)	Maximum Value ($)	Standard Deviation ($)
Cement	2486.25	2109.89	60.59	6119.39	2034.56
Sand	8949.44	10328.17	1467.16	15339.66	4180.64
Iron Rods	560.53	560.83	319.52	819.10	151.08
Roofing	3395.84	3418.23	1266.55	5464.05	1301.38
Paint	801.46	802.13	5.75	1592.75	1592.75
Wood	970.71	970.79	280.33	1661.63	424.99
HUP	55225.48	52602.50	31455.00	83600.00	15011.80

4.2. Developed Model Efficiency Test Results

The acquired dataset was partitioned into an 80% training set (24 data points) and a 20% independent testing set (6 data points) to develop the following AI models (AdaBoost, KNN, DT, RF, GBR, and SVM). The training set was used for the model fit, while the independent data served as the testing set to validate the models' developed forecasting capabilities. The intercomparison among the methods applied is shown in Table 2.

Table 2 Performance Indices Testing Results

Model	MAPE%	CVRMSE%	R2	R
AdaBoost	0.024	3.216	0.981	0.99045
kNN	0.016	2.271	0.991	0.99549
DT	0.036	5.352	0.949	0.97417
RF	0.022	2.956	0.984	0.99197
GBR	0.017	1.968	0.993	0.99649
SVM	0.044	4.39	0.965	0.98234

The measure R in Table 2 shows the extent of the linear relationship between the predicted and observed values. The closer the R-value near 1, the stronger the relationship between the two variables mentioned above. From Table 2, GBR had the highest R = 0.996 value than the other models. This is an indication of the strong linear relationship between the predicted and the observed HUP values. On the other hand, the R2 value shows the proportion of the HUP that the developed model failed to account for. From Table 2, the GBR model had the highest R2 = 0.993. This implies that the GBR model failed to explain only 0.07% of the proportion in the HUP than any of the contending models. Once again, the GBR method had the least CVRMSE = 1.968% as shown in Table 2. This means that the GBR model failed to explain only 1.968% of the mean variation in the HUP than any of the methods under consideration. The MAPE value of 0.017% in Table 2 indicates that the GBR model was able to explain 99.983% variability in the dataset.

Overall, the graph in Figure 4 demonstrates that while all the models attempt to capture the trends in the dataset, the GBR model achieved the highest accuracy and exhibited the least computed error. This conclusion was drawn from the model's prediction values' ability to closely align with the observed dataset. Exhibiting the smallest spikes indicates the model's superior capacity for precise HUP prediction.

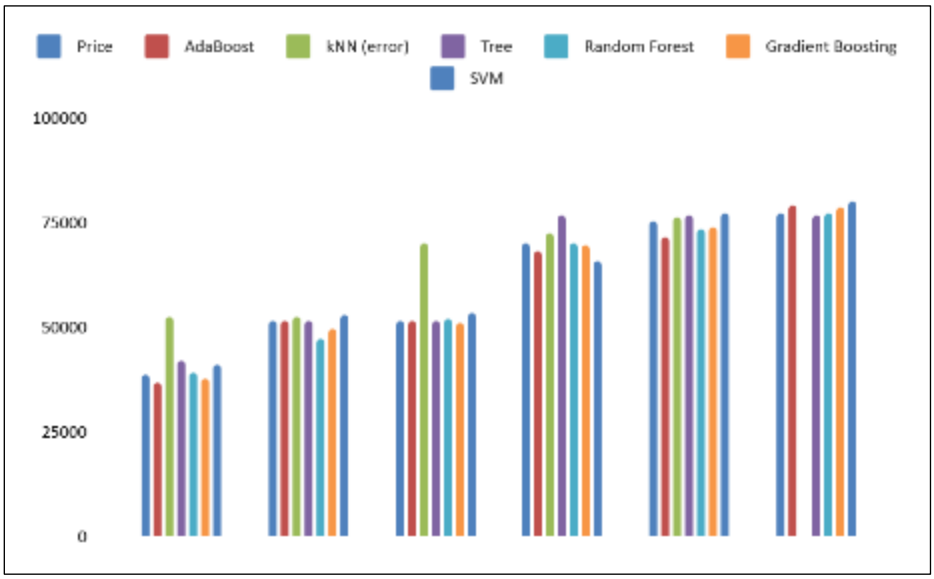

Figure 4 Line Graph of Model Errors

5. Conclusion

AI learning is a promising approach for prediction problems since the method produces more satisfactory results than the traditional methods in the literature. In this study, a GBR method was developed to predict HUP. The developed model has been compared with five standard AI methods (AdaBoost, KNN, DT, RF, and SVM). The GBR method produced superior results as compared to the other contending methods. This was evident from the performed statistical analysis, where GBR had the lowest MAPE (0.017%), CVRMSE (1.968%), and highest R^2 (0.993) and R (0.99649) values as compared with the other competing methods. Based on the results, the overall performance of the GBR approach in comparison with the five state-of-the-art methods has been demonstrated to have the ability to be used to predict HUP with some level of certainty. It can therefore be concluded that the presented paper can be useful for housing industrial players and researchers who will be modeling to use AI to predict the HUP.

Compliance with ethical standards

Data Availability

The housing unit price data used to support the findings of this study are available from the corresponding author upon request

Disclosure of conflict of interest

The authors have no conflicts of interest to declare.

References

[1] Alzain, E., Alshebami, A. S., Aldhyani, T. H., and Alsubari, S. N. (2022). Application of artificial intelligence for predicting real estate prices: the case of Saudi Arabia. Electronics, 11(21), 3448.

[2] [2] Soft Computing, 141, 110283.

[3] Bentéjac, C., Csörgő, A., and Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. Artificial Intelligence Review, 54, 1937-1967.

[4] Čeh, M., Kilibarda, M., Lisec, A., and Bajat, B. (2018). Estimating the performance of random forest versus multiple regression for predicting the prices of the apartments. ISPRS International Journal of Geo-Information, 7(5), 168.

[5] Çetin, A. İ., and Büyüklü, A. H. (2025). A new approach to K-nearest neighbors distance metrics on sovereign country credit rating. Kuwait Journal of Science, 52(1), 100324.

[6] Chen, E., and He, X. J. (2018). Crude Oil Price Prediction with Decision Tree-Based Regression Approach. Journal of International Technology and Information Management, 27(4), 1-16.

[7] Dargahi-Zarandi, A., Hemmati-Sarapardeh, A., Shateri, M., Menad, N. A., and Ahmadi, M., 2020. Modeling minimum miscibility pressure of pure/impure CO2-crude oil systems using adaptive boosting support vector regression: Application to gas injection processes. Journal of Petroleum Science and Engineering, 184, 106499.

[8] Dong, B., Cao, C., and Lee, S. E. (2005). Applying support vector machines to predict building energy consumption in tropical regions. Energy and Buildings, 37(5), 545-553.

[9] Everingham, Y., Sexton, J., Skocaj, D., and Inman-Bamber, G. (2016). Accurate prediction of sugarcane yield using a random forest algorithm. Agronomy for sustainable development, 36, 1-9.

[10] Feng, D. C., Liu, Z. T., Wang, X. D., Chen, Y., Chang, J. Q., Wei, D. F., and Jiang, Z. M. (2020). Machine learning-based compressive strength prediction for concrete: An adaptive boosting approach. Construction and Building Materials, 230, 117000.

[11] Fouedjio, F. (2020). Exact conditioning of regression random forest for spatial prediction. Artificial Intelligence in Geosciences, 1, 11-23.

[12] Gbashi, S. M., Adedeji, P. A., Olatunji, O. O., and Madushele, N. (2024). Optimal feature selection for weighted k-nearest neighbors for compound fault classification in wind turbine gearbox. Results in Engineering, 103791.

[13] Johansson, U., Boström, H., Löfström, T., and Linusson, H. (2014). Regression conformal prediction with random forests. Machine learning, 97, 155-176.

[14] Li, D. Y., Xu, W., Zhao, H., and Chen, R. Q. (2009). An SVR-based forecasting approach for real estate price prediction. In 2009 International Conference on machine learning and cybernetics, 2, 970-974). IEEE.

[15] Li, R., Li, W., and Zhang, H. (2022). State of Health and Charge Estimation Based on Adaptive Boosting integrated with particle swarm optimization/support vector machine (AdaBoost-PSO-SVM) Model for Lithium-ion Batteries. International Journal of Electrochemical Science, 17(2), 220212.

[16] Lin, J. Y., Cheng, C. T., and Chau, K. W. (2006). Using support vector machines for long-term discharge prediction. Hydrological Sciences Journal, 51(4), 599-612.

[17] Mora-Garcia, R. T., Cespedes-Lopez, M. F. and Perez-Sanchez, V. R. (2022). Housing price prediction using machine learning algorithms in COVID-19 times. Land, 11(11), 2100.

[18] Neves, F. T., Aparicio, M., and De Castro Neto, M. (2024). The Impacts of Open Data and eXplainable AI on Real Estate Price Predictions in Smart Cities. APPLIED SCIENCES-BASEL, 14(5), 1- 40.

[19] Radhika, Y., and Shashi, M. (2009). Atmospheric temperature prediction using support vector machines. International journal of computer theory and engineering, 1(1), 55.

[20] Rampini, L., and Re Cecconi, F. (2022). Artificial intelligence algorithms to predict Italian real estate market prices. Journal of Property Investment and Finance, 40(6), 588-611.

[21] Reddy, P. S. M., and Chandra, J. P. (2023). Decision tree regressor compared with random forest regressor for house price prediction in Mumbai. Journal of Survey in Fisheries Sciences, 10(1S), 2323-2332.

[22] Sai, M. J., Chettri, P., Panigrahi, R., Garg, A., Bhoi, A. K., and Barsocchi, P. (2023). An ensemble of Light Gradient Boosting Machines and adaptive boosting for the prediction of type-2 diabetes. International Journal of Computational Intelligence Systems, 16(1), 14.

[23] San Millan-Castillo, R., Morgado, E., and Goya-Esteban, R. (2019). On the use of decision tree regression for predicting vibration frequency response of handheld probes. IEEE Sensors Journal, 20(8), 4120-4130.

[24] Sharma, S., and Gill, S. S. (2024). Advanced Machine Learning Models for Real Estate Price Prediction. In Applications of AI for Interdisciplinary Research,103-121. CRC Press.

[25] Singh, U., Rizwan, M., Alaraj, M., and Alsaidan, I. (2021). A machine learning-based gradient boosting regression approach for wind power production forecasting: A step towards smart grid environments. Energies, 14(16), 5196.

[26] Soltani, A., Heydari, M., Aghaei, F., and Pettit, C. J. (2022). Housing price prediction incorporating spatio-temporal dependency into machine learning algorithms. Cities, 131, 103941.

[27] Someetheram, V., Marsani, M. F., Kasihmuddin, M. S. M., Jamaludin, S. Z. M., Mansor, M. A., and Zamri, N. E. (2025). Hybrid double ensemble empirical mode decomposition and K-Nearest Neighbors model with improved particle swarm optimization for water level forecasting. Alexandria Engineering Journal, 115, 423-433.

[28] Štubňová, M., Urbaníková, M., Hudáková, J., and Papcunová, V. (2020). Estimation of residential property market price: comparison of artificial neural networks and hedonic pricing model. Emerging Science Journal, 4(6), 530-538.

[29] Tandon, R. (2024). The Machine Learning Based Regression Models Analysis for House Price Prediction. International Journal of Research and Analytical Reviews (IJRAR), 11(3), 296-305.

[30] Tchuente, D., and Nyawa, S., 2022. Real estate price estimation in French cities using geocoding and machine learning. Annals of operations research, 308(1), 571-608.

[31] Terregrossa, S. J., and Ibadi, M. H. (2021). Combining housing price forecasts generated separately by hedonic and artificial neural network models. Asian Journal of Economics, Business and Accounting, 21(1), 130-148.

[32] Upadhyay, D., Manero, J., Zaman, M., and Sampalli, S. (2020). Gradient boosting feature selection with machine learning classifiers for intrusion detection on power grids. IEEE Transactions on Network and Service Management, 18(1), 1104-1116.

[33] Xu, X., and Zhang, Y. (2021). House price forecasting with neural networks. Intelligent Systems with Applications, 12, 200052.

[34] Yakub, A. R. A., Hishamuddin, M., Ali, K., Achu, R. B. A. J., and Folake, A. F. (2020). The effect of adopting micro and macroeconomic variables on real estate price prediction models using ANN: A Systematic Literature. Journal of Critical Reviews, 7(11), 492-498.

[35] Yang, Y., Dai, H. M., Chao, C. H., Wei, S., and Yang, C. F. (2023). Training a Neural Network to Predict House Rents Using Artificial Intelligence and Deep Learning. Sensors and Materials, 35.

[36] Zendehboudi, A., Baseer, M. A., and Saidur, R. (2018). Application of support vector machine models for forecasting solar and wind energy resources: A review. Journal of cleaner production, 199, 272-285.

[37] Zhang, Y., and Haghani, A. (2015). A gradient boosting method to improve travel time prediction. Transportation Research Part C: Emerging Technologies, 58, 308-324.