MACHINE LEARNING-BASED PREDICTION MODELS FOR BUDGET FORECAST IN CAPITAL CONSTRUCTION

Prashnna Ghimire [1]
Sudan Pokharel[1]
Kyungki Kim[1]
Philip Barutha[2]

[1] University of Nebraska-Lincoln | USA
[2] University of Notre Dame| USA
Corresponding author: pghimire3@huskers.unl.edu

Keywords

Machine Learning; Budget Forecast; Capital Construction; Decision Tree

Abstract

Forecasting the final budget at different phases of the design-build process is one of the most challenging aspects of capital construction project management. Predicting the construction budget is critical to ensuring the success of the project and avoiding delays and cost overruns. This paper presents data-driven machine learning models to predict the final budget of capital construction. Using the data of the City of New York's capital construction projects with twenty-five million or greater value, this study applied machine learning algorithms to identify the best model that predicts the final budget based on project type, project phase, budget changes, and schedule changes. Four Machine Learning (ML) models- Decision Tree (DT), Random Forest (RF), Gradient Boosting Regressor (GBR), and Multiple Linear Regression (MLR) - were trained and tested to compare and determine the best model to explain. The results of this study demonstrate how machine learning can effectively predict the project budget and its practical applicability. The outcome of this will help project stakeholders in decision-making by forecasting the final budget of the capital construction in any phase of the project lifecycle.

1. INTRODUCTION

There are many uncertainties associated with complex and large-scale capital construction projects, such as changes in scope, schedule, and cost, which makes it difficult to forecast final budgets for these projects. An inaccurate budget allocation often prevents organizations from effectively controlling the costs of a construction project, resulting in an unreliable forecast of project profits or losses after the construction is complete, which renders organizations unable to control the costs of the project[1]. It has been extensively discussed in the literature how changes can occur during capital construction projects at all stages, resulting from a variety of causes, with potentially negative impacts, such as increased project costs[2]–[5]. Design changes are a significant contributor to construction delays and cost overruns in most public projects[6]. Estimation is a crucial aspect of the success of construction projects, and it is typically performed at every phase of the project lifecycle [7]. The complexity of the built environment necessitates innovation, driving built environment professionals to work intelligently and develop new tools and methods [8].

Despite the fact that the construction industry generates a large amount of data, its value is underutilized. Machine learning (ML), a branch of Artificial Intelligence (AI), can extract hidden patterns from this data and transform them into explicit knowledge to solve construction industry problems[9]. There are some studies in the body of literature regarding cost estimation and budget estimation. As far as the authors are aware, the combination of qualitative and quantitative project variables is not taken into account when predicting the final budget. Moreover, cost forecasting is a crucial aspect of industry practice, necessary to prevent deficits that could have a negative impact on payment schedules and stakeholders' reputation. However, this process is typically long and manual, requiring a substantial amount of time and money[10]. Therefore, the development of reliable predictive models that account for key project variables such as project phase, project type, schedule changes, and budget changes that play together in a model is essential to support informed decision-making and minimize the risk of cost overruns and delays. This study implements and compares four machine learning models, namely DT, RF, GBR, and MLR, to determine the most effective model for accurately forecasting the final budget of a project. The analysis takes into account key variables such as project type, project phase, total schedule changes, and total budget changes.

2. METHODOLOGY

There are three primary steps involved in the overall process: (1) data preparation, which involves extracting data, conducting exploratory data analysis, and performing feature engineering, (2) constructing machine learning regression models, such as DT, RF, GBR and MLR, and (3) evaluating model results and performance to identify the best model to forecast the capital construction budget.

2.1. DATA PREPARATION

2.1.1. DATA

This study utilizes data from the NYC Open Data database updated in January 2023, comprising information on capital projects in the State of New York, USA with a budget of $25 million or more that are currently in the design, procurement, or construction phase [11]. The original data set includes 3157 data points and 16 variables for capital projects dating from 1995 to 2023. These projects are from the State of New York, USA. Each project has a series of variables including identifiers, in this study we have selected five variables: project category (coded as ‘cate’), project phase (coded as ‘phase’), budget forecast (coded as ‘bud get’), total schedule changes (coded as ‘tsc’), and total budget changes (coded as ‘tbc’). The project category represents the type of project that has multiple categories. The project phase during the reporting period has three categories: design, procurement, and construction. The budget forecast represents the total cost of the project estimated at the time of the reporting. Total schedule changes refer to the number of days the project is ahead or behind schedule since the design start date, with negati ve numbers indicating days ahead of schedule. And, the total budget changes are defined as the number of dollars the project is over or under budget since the design start date, with a negative number indicating dollars under budget.

2.1.2. EXPLORATORY DATA ANALYSIS

An exploratory data analysis (EDA), which is an important first step, was conducted to gain insights from the data before developing machine learning models. The purpose of this approach is to understand what the data can communicate to us and summarize the characteristics, and identify patterns of the data set through an analysis of the data[12]. Python was utilized in this study due to its open-source nature and the availability of rich libraries such as pandas, numpy, scikit-learn, and matplotlib, which are useful for performing EDA[13]–[17].

We loaded the required libraries and dataset in python. In this multivariate distribution of dataset, we only kept the variables we wanted to analyze that made the data shape (3157, 8). If missing values are present in only a few observations, the attribute is considered valid. However, if missing values occur in a significant number of observations, it is recommended to eliminate those observations[18]. We identified and dropped the rows which refer to missing values in the dataset using two different python functions. To regulate the cost-related variables, the values of the budget and tbc were transformed into million USD. Next, scatter plots were used to visualize the pattern and distribution between each feature variable and the target variable. Additionally, a correlation plot was generated to assess the strength and direction of the linear relationship between the va riables. Both plots showed evidence of a non-linear relationship between the variables.

2.1.3. FEATURE ENGINEERING

Feature engineering in machine learning includes the process of selecting and transforming input variables to train the model [19]. Based on the construction categories defined by Construction Industry Institute’s PDRI overview[20], all the projects are assigned to one of the three categories- industrial, infrastructure, and building. Categorical variables describe a particular category whereas quantitative variables contain numeric data. We have three quantitative variables and two nominal categorical variables in our final data set. It is recommended to perform a transformation on a categorical variable prior to utilizing it in a ML regression model for analysis. One approach to achieving this is one-hot encoding through the use of the get_dummies function. This allows the machine learning model to effectively interpret the categorical data and incorporate it into the analysis. This method involves creating new binary columns for each unique category in the original variable, with a value of 1 representing the presence of that category and 0 indicating its absence. Consequently, our project category variable generates three categorical variables: 'cate_building', 'cate_industrial', and 'cate_infrastructure', while the project phase variable is transformed into three categorical variables: 'phase_Construction', 'phase_Design', and 'phase_Procurement'.

2.2. MACHINE LEARNING MODEL

In machine learning, splitting data into training set and test set is a common approach. It is suggested that a 70/30 ratio for the training and testing datasets was optimal for training and validating the models [21]. The final data frame of size (2288, 9) was randomly divided into training (70%) and testing (30%). The variables are divided into feature and target variables and ther e are six feature variables (X): 'cate_building', 'cate_industrial', 'cate_infrastructure', 'phase_Construction', 'phase_Design', 'phase_Procurement', 'tbc', 'tsc', and one target variable (y): 'budget' in our data frame.

2.2.1. DECISION TREE

The Decision Tree, a supervised learning technique, is one of the most popular machine learning algorithms for the solution of classification and regression problems. It works by recursively splitting the data space into smaller subsets and fitting a basic prediction model in each partition. Classification trees are created for dependent variables that possess a finite number of values which are not ordered, and the misclassification cost is utilized to measure the prediction error. On the other hand, regression trees are intended for dependent variables that contain continuous or ordered discrete values, and the prediction error is generally calculated by the squared difference between the predicted and observed values [22]. A decision tree is comprised of decision nodes that test the value of an attribute, edges that indicate the outcome of a test and connect to the next node, and leaf nodes that make predictions about the outcome. When combined, these components form a comprehensive structure of a decision tree, which is depicted in Figure 1[23].

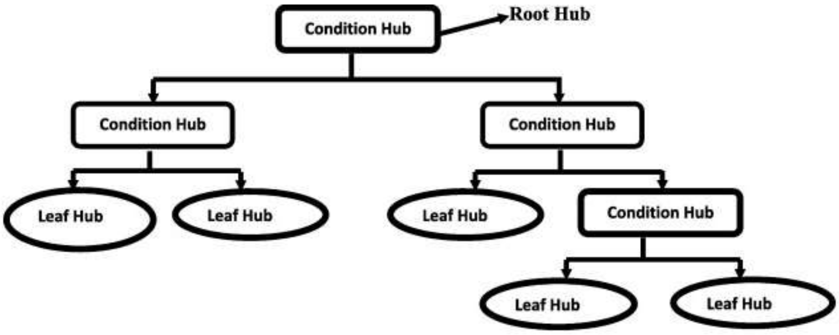

Figure 1. Decision Tree Structure [14].

To develop our model, we utilized the DecisionTreeRegressor library from the sklearn package. The selection of the model parameters play a crucial role in the model learning process. Hyperparameters tuning involves selecting the most optimal parameters for a particular learning algorithm. Typically, these hyperparameters must be determined prior to training, and their selection can significantly impact the performance of the resulting model [24]. Initially, we employed the commonly used GridSearchCV package to identify the optimal hyperparameters for all models, and subsequently compared its performance with that of random parameter tuning. We observed that GridSearchCV did not outperform our random parameter tuning approach in our specific case. Upon fine-tuning, we determined that the optimal hyperparameters were splitter='best', max_depth=None, min_samples_leaf=1, and max_features='auto'.

2.2.2. RANDOM FOREST

Random forest algorithm is a powerful nonparametric machine learning algorithm for classification and regression problem. A random forest consists of multiple tree predictors, where each tree's predictions depend on the values of a random vector. This vector is independently sampled with the same distribution for all trees in the forest[25]. During training, the random forest regressor generates multiple decision trees and combines their predictions to produce a final outcome. By randomly selecting a subset of features for each tree, the algorithm partitions the data into smaller subsets, with each tree making predictions for its subset. The final prediction is obtained by averaging the outputs of all the trees. This approach is widely adopted in various fields due to the model's robustness against noise and outliers in the data, which makes it highly accurate. While it is commonly acknowledged that random forest often performs well using default hyperparameters provided in software packages, optimizing the hyperparameters can lead to even better performance [26]. The number and depth of decision trees, as well as the minimum number of samples required to split an internal node, are among the most crucial hyperparameters of the Random forest model [18]. The optimal parameters determined through the random search were n_estimators = 100, max_depth =None, random_state=seed, criterion='absolute_error', min_samples_split=2.

2.2.3. GRADIENT BOOSTING REGRESSOR

Gradient Boosting Regressor is an ensemble machine learning algorithm that combines multiple weak learners, usually decision trees, to create a more robust and accurate model. Unlike Random Forest, GBR builds trees sequentially, with each tree aiming to correct the errors of the previous tree. This iterative approach results in a model that continually improves its predictions as more trees are added. GBR is a resilient algorithm that can effectively handle noisy data and is highly resistant to overfitting, making it an attractive option for modeling complex datasets. Boosting on successive subsets of data can also be employed in situations where there is insufficient main memory with random access capabilities to store the complete dataset[27]. Parameters such as learning_rate=0.05, n_estimators=100, max_depth=5 were identified as the best parameters in our GBR model.

2.2.4. MULTIPLE LINEAR REGRESSION

Multiple Linear Regression is a machine learning algorithm that establishes a linear relationship between one or more independent variables and a dependent variable. This technique is useful for regression problems in supervised learning where the goal is to predict a quantitative variable. The coefficients of the independent variables are estimated by the model and can be utilized to predict the value of the dependent variable. Linear regression has a drawback that it may not be an appropriate model when working with non-linear relationships, and it oversimplifies many real-world problems[28].

2.3. MODEL PERFORMANCE EVALUATION

Model performance evaluation metrics provide a quantitative way to assess the performance of prediction models. There are various ways to evaluate the model performance of machine learning regression models. We used three commonly used evaluation metrics such as Coefficient of Determination (R^2), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE).

2.3.1. COEFFICIENT OF DETERMINATION (R^2)

It is a widely used metric in machine learning regression, indicating the proportion of target variable variance explained by the model. The range of R^2 is 0 to 1, with higher values indicating better performance. The R^2 of 1 indicates a perfect fit, while 0 means the model can't explain the variance. The metric represents the percentage of the total variation in the target variable accounted for by the model, useful for evaluating the model's goodness of fit and comparing regression model performance.

$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$, Where, y_i= actual value, $\hat{y}_i$= predicted value, $\bar{y}_i$= mean of y_i, and n= the sample size. (1)

2.3.2. MEAN ABSOLUTE ERROR (MAE)

It provides a measure of the average absolute difference between the predicted and actual values of the target variable. A lower MAE value signifies superior model performance, indicating closer agreement between the model's predictions and actual values. The formula for MAE is:

$MAE = \frac{1}{n} . [\sum_{i=1}^{n} |y_i - \hat{y}_i \,|]$, Where, y_i= actual value, $\hat{y}_i$= predicted value, and n= the sample size. (2)

2.3.3. ROOT MEAN SQUARED ERROR (RMSE)

RMSE evaluates the average magnitude of prediction errors, accounting for both their direction and direction. The lower the RMSE value, the higher the accuracy of the model's predictions. The RMSE is calculated as follows:

$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$, Where, y_i= actual value, $\hat{y}_i$= predicted value, and n= the sample size. (3)

3. RESULTS AND DISCUSSION

The comparison of model efficiency of four different machine learning algorithms- DT, RF, GBR, and MLR- used in this study to predict the capital construction budget is shown in Table 1 below.

Table 1. Comparison of Four Machine learning Models.

MODEL NAME	R^2	MAE	RMSE
DT	0.96	17.55	43.82
RF	0.95	19.27	48.75
GBR	0.90	40.41	70.22
MLR	0.28	102.39	184.34

The evaluation metrics used in the study demonstrated that the DT algorithm had the most accurate prediction capability, as reflected by its superior performance values. The results showed that DT had the highest R^2 value of 0.96, indicating that it can account for 96% of the total variability in the budget. Additionally, it had the lowest MAE value of 17.55, signifying that the average absolute difference between the predicted and actual budget was the smallest compared to other models. Also, DT had the RMSE value of 43.82, indicating that the average magnitude of the errors in the predictions was the lowest among the four models. In our analysis, it appeared that the predictive performance of DT and RF models is quite similar, with little variation between them. In contrast, GBR seems to be less effective in making accurate predictions. Additionally, our findings indicated that the MLR model is not performing poorly in terms of predictive capacity.

Our analysis has revealed that tree-based models-DT, RF, and GBR- excel in capturing nonlinearities and variable interactions without explicit specifications. This is in contrast to MLR, which failed to capture such non-linear relationships between feature variables and target variable, resulting in poor performance in this particular case. Thus, the DT model represents the most accurate budget prediction model that can enhance the efficiency of the decision-making process during any phase of capital construction projects. The authors are highly confident that the proposed DT model can be utilized by the project stakeholders such as capital project owners, project management organizations, contractors, designers, and project participants, to accurately predict the final project budget. The model takes into account various factors such as project category, project phase, total budget changes, and total schedule changes. Figure 2 represents the scatter plots between the target and predicted budget in four different models. A model that most accurately predicts the outcome, DT, in the scatterplot has points clustered tightly around the diagonal line.

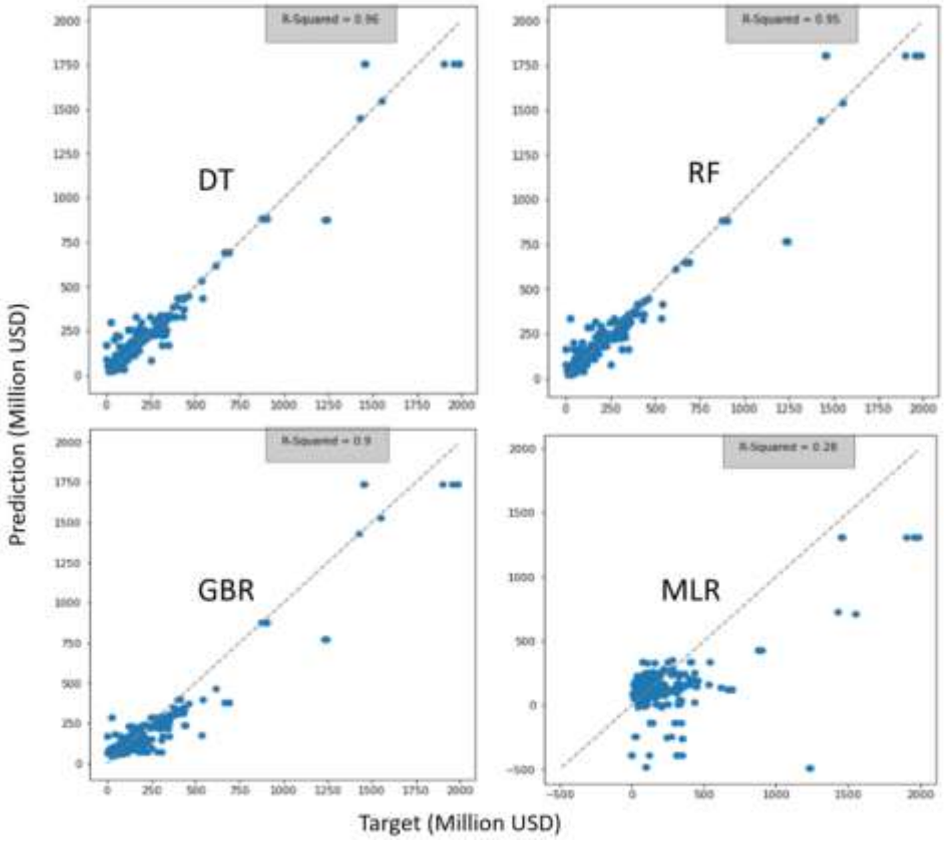

Figure 2. Scatter Plots of Four Machine Learning Models.

4. CONCLUSION AND RECOMMENDATIONS

In this study, four machine learning models, DT, RF, GBR, and MLR, were developed and compared using 2288 project data points. The analysis revealed a noticeable disparity in the performance of tree-based models and the MLR model. It was observed that the tree-based models were more effective in capturing the nonlinearities between variables than the MLR model. Evaluation based on three metrics, R^2, MAE, and RMSE, indicated that the DT model performed the best, with respective values of 0.96, 17.55, and 43.82. The majority of construction-related datasets in the real world include both categorical and quantitative variables. In such cases, the DT model has shown to be particularly effective in producing accurate predictions. This model can be used by stakeholders in capital construction projects, including the owner, public authority, designer, contractor, and other project participants to produce accurate budget predictions. For example, there is a medical facility project in the construction phase that needs to meet its schedule despite the fact that there have been many changes in budget and schedule, but the final cost is not certain what it will be, then this model can help to predict the final cost in a matter of seconds. It is anticipated to be a highly beneficial tool throughout the lifecycle of any capital construction project. Further research can be done exploring other machine learning models as well as deep neural networks to identify and compare the efficacy, and computational complexities in this type of data with more variables.

References

[1] J.-B. Yang and C.-C. Chen, “Causes of Budget Changes in Building Construction Projects: An Empirical Study in Taiwan,” *The Engineering Economist*, vol. 60, no. 1, pp. 1–21, Jan. 2015, doi: 10.1080/0013791X.2013.879972.

[2] T. Hsieh, S. Lu, and C. Wu, “Statistical analysis of causes for change orders in metropolitan public works,” *International Journal of Project Management*, vol. 22, no. 8, pp. 679–686, Nov. 2004, doi: 10.1016/j.ijproman.2004.03.005.

[3] B.-G. Hwang and L. K. Low, “Construction project change management in Singapore: Status, importance and impact,” *International Journal of Project Management*, vol. 30, no. 7, pp. 817–826, Oct. 2012, doi: 10.1016/j.ijproman.2011.11.001.

[4] I. A. Motawa, C. J. Anumba, S. Lee, and F. Peña-Mora, “An integrated system for change management in construction,” *Automation in Construction*, vol. 16, no. 3, pp. 368–377, May 2007, doi: 10.1016/j.autcon.2006.07.005.

[5] M. Sun and X. Meng, “Taxonomy for change causes and effects in construction projects,” *International Journal of Project Management*, vol. 27, no. 6, pp. 560–572, Aug. 2009, doi: 10.1016/j.ijproman.2008.10.005.

[6] C. Wu, T. Hsieh, and W. Cheng, “Statistical analysis of causes for design change in highway construction on Taiwan,” *International Journal of Project Management*, vol. 23, no. 7, pp. 554–563, Oct. 2005, doi: 10.1016/j.ijproman.2004.07.010.

[7] V. Chandanshive and A. Kambekar, “Estimation of Building Construction Cost Using Artificial Neural Networks,” *J. Soft Comput. Civ. Eng.*, vol. 3, no. 1, Jan. 2019, doi: 10.22115/scce.2019.173862.1098.

[8] J. Kim, J. Liu, and P. Ghimire, *The Categorization of Virtual Design and Construction Services*. 2019.

[9] D. Chakraborty, H. Elhegazy, H. Elzarka, and L. Gutierrez, “A novel construction cost prediction model using hybrid natural and light gradient boosting,” *Advanced Engineering Informatics*, vol. 46, p. 101201, Oct. 2020, doi: 10.1016/j.aei.2020.101201.

[10] “Construction Cost Forecasting | Risk Analysis & Management,” *Spire Consulting Group*. https://www.spireconsultinggroup.com/services/construction-advisory-consulting/risk-management/cost-forecasting/ (accessed Mar. 30, 2023).

[11] “Capital Projects | NYC Open Data.” https://data.cityofnewyork.us/City-Government/Capital-Projects/n7gv-k5yt (accessed Mar. 30, 2023).

[12] Dr. S. Pani, “Exploratory Data Analysis using Python,” Nov. 2019, doi: 10.35940/ijitee.L3591.1081219.

[13] “Welcome to Python.org,” *Python.org*, Mar. 23, 2023. https://www.python.org/ (accessed Mar. 30, 2023).

[14] “pandas - Python Data Analysis Library.” https://pandas.pydata.org/ (accessed Mar. 30, 2023).

[15] “NumPy.” https://numpy.org/ (accessed Mar. 30, 2023).

[16] “scikit-learn: machine learning in Python — scikit-learn 1.2.2 documentation.” https://scikit-learn.org/stable/ (accessed Mar. 30, 2023).

[17] “Matplotlib — Visualization with Python.” https://matplotlib.org/ (accessed Mar. 30, 2023).

[18] Y. Wang, Z. Shao, and R. L. K. Tiong, “Data-Driven Prediction of Contract Failure of Public-Private Partnership Projects,” *Journal of Construction Engineering and Management*, vol. 147, no. 8, p. 04021089, Aug. 2021, doi: 10.1061/(ASCE)CO.1943-7862.0002124.

[19] H. Liu, *Feature Engineering for Machine Learning and Data Analytics*, 1st ed. CRC Press, 2018. doi: 10.1201/9781315181080.

[20] “CII - Project Definition Rating Index Overview.” https://www.construction-institute.org/resources/knowledgebase/pdri-overview (accessed Mar. 30, 2023).

[21] Q. H. Nguyen *et al.*, “Influence of Data Splitting on Performance of Machine Learning Models in Prediction of Shear Strength of Soil,” *Mathematical Problems in Engineering*, vol. 2021, p. e4832864, Feb. 2021, doi: 10.1155/2021/4832864.

[22] W.-Y. Loh, “Classification and regression trees,” *WIREs Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 14–23, 2011, doi: 10.1002/widm.8.

[23] J. Singh Kushwah, A. Kumar, S. Patel, R. Soni, A. Gawande, and S. Gupta, “Comparative study of regressor and classifier with decision tree using modern tools,” *Materials Today: Proceedings*, vol. 56, pp. 3571–3576, Jan. 2022, doi: 10.1016/j.matpr.2021.11.635.

[24] M. Claesen and B. De Moor, “Hyperparameter Search in Machine Learning.” arXiv, Apr. 06, 2015. Accessed: Mar. 24, 2023. [Online]. Available: http://arxiv.org/abs/1502.02127

[25] L. BREIMAN, “Random Forests”, [Online]. Available: https://link.springer.com/content/pdf/10.1023/A:1010933404324.pdf

[26] P. Probst, M. N. Wright, and A.-L. Boulesteix, “Hyperparameters and tuning strategies for random forest,” *WIREs Data Mining and Knowledge Discovery*, vol. 9, no. 3, p. e1301, 2019, doi: 10.1002/widm.1301.

[27] J. H. Friedman, “Greedy Function Approximation: A Gradient Boosting Machine,” *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[28] S. Ray, “A Quick Review of Machine Learning Algorithms,” presented at the International conference on machine learning, big data, cloud and parallel computing (COMITCon), Feb. 2019.