*Petroleum Science and Engineering*
2025, Vol. 9, No. 1, pp. 38-47
https://doi.org/10.11648/j.pse.20250901.14

SciencePG
Science Publishing Group

Research Article

# Application of Machine Learning for Bit-formation Matching in Drilling Operations

**Shedrach Igemhokhai[1,*], Kelani Bello[1], Abiodun Olaoye[2], Ehigie Momodu[2], Abayomi Adejumo[3], Oladele Akindayini[4]**

[1]Department of Petroleum Engineering, Faculty of Engineering, University of Benin, Benin, Nigeria
[2]Gankco Energy Services Limited, Abuja, Nigeria
[3]Oriental Energy Resources Limited, Lagos, Nigeria
[4]Sterling Global Exploration and Production Company Limited, Lagos, Nigeria

## Abstract

Efficient bit formation matching is imperative for the success and cost-effectiveness of drilling operations and emissions reduction to provide energy solutions. Currently, drill bit selection predominantly depends on historical data and experiential knowledge. While machine learning, particularly Artificial Neural Networks (ANNs), has gained prominence in bit selection, other diverse and impactful algorithms such as XGBOOST and Random Forest (RF), are often overlooked. This paper involves the systematic application and comparative analysis of XGBOOST, RF, and ANN, alongside an optimization approach using Genetic Algorithm. The study comprehensively considers various influential factors including formation properties, drilling fluid characteristics, bit design, and operational parameters. In this study, we achieved promising results with the highest classification accuracy for bit selection recorded at 0.97 using the XGBOOST model, while RF and ANN yielded accuracies of 0.91 and 0.93 respectively. Additionally, we obtained impressive R squared values of 0.991, 0.975, and 0.953 for predicting the Rate of Penetration using the XGBOOST, ANN, and RF models respectively. These algorithms, coupled with the optimization techniques, aim to establish a robust framework for nuanced and accurate bit-formation matching. The results obtained hold significant potential for minimizing costs and optimizing resource allocation and utilization during the planning and execution of drilling projects in the oil and gas industry.

## Keywords

Machine Learning, Bit-Formation Matching, Drilling, Artificial Neural Networks, XGBOOST, Random Forest

## 1. Introduction

Efficient bit formation matching is a critical factor in the success and cost-effectiveness of drilling operations and emissions reduction to provide energy solutions in the oil and gas industry. The selection of an appropriate drill bit tailored to the geological formation being penetrated significantly influences drilling performance, including rates of penetration, tool wear, and overall operational efficiency. However, traditional methods of bit selection often rely on historical data

[*]Corresponding author: shedrach.igemhokhai@eng.uniben.edu (Shedrach Igemhokhai)

**Received:** 9 April 2025; **Accepted:** 27 April 2025; **Published:** 29 May 2025

and subjective expert judgment, which may lead to suboptimal outcomes due to the complex and dynamic nature of geological formations and drilling environments.

Moreover, the increasing diversity and heterogeneity of subsurface formations pose significant challenges to conventional decision-making processes. Variations in lithology, porosity, permeability, and other formation properties necessitate a more sophisticated approach to bit formation matching that can account for the intricate interplay between geological characteristics, drilling parameters, and tool design.

In light of these challenges, there is a pressing need for advanced methodologies that can leverage the wealth of available data and provide actionable insights into bit-formation compatibility. Machine learning (ML) techniques offer a promising avenue for addressing this need by enabling the analysis of large and complex datasets to uncover hidden patterns and relationships that may inform more informed bit selection decisions.

However, despite the growing interest in ML-driven approaches for bit formation matching, there exists a gap in the literature regarding the systematic evaluation and comparison of different ML algorithms and optimization techniques tailored specifically to this application.

Drilling a well requires careful consideration of several important factors, including bit selection. Consequently, choosing the right bits is a difficult task because bit performance is affected by a number of variables, including ROP, revolutions per minute (RPM), mud weight (MW), and depth (D) [1]. Cost per foot is typically the basis for the most basic type of bit selection. With this approach, the bit that will offer the lowest cost per foot over the next interval is simply selected. Furthermore, other variables like offset, journal angle, and other design elements are also taken into account. This distinguishes one bit from another based on the unique surrounding [2].

Using offset well performance data is the standard method for bit selection. The bit type with the highest penetration rate or the bit with the lowest cost per foot are the two most often utilised criteria for choosing the bit for the following interval. The selection procedure also takes into account elements like hydraulics, formation hardness, bit design, and operational characteristics. The selection process is a trial-and-error process because of the quantity of variables taken into account [3]. This method frequently overlooks several crucial factors influencing bit performance and cannot ensure the best bit type is chosen.

## 1.1. Types of Bits

Based on their design qualities, rotary drilling bits can be broadly categorized as either roller cutter bits or drag bits as shown in table 1. Fixed cutter blades are a frequent feature of drag bits. These blades are included into the bit's body. The drill string rotates as a single unit during the rotation. Conversely, the roller cutter bits typically feature two or more cones with fundamental cutting components. When the bottom-hole rotation occurs, these cutters revolve around the cone's axis [4].

## 1.2. Iadc Bit Classification System

### 1.2.1. Roller Cutter Bit Classification

The world saw the introduction of the International Association of Drilling Contractors' (IADC) bit code categorization system in 1940. The IADC bit code categorization system for rolling cutter bits was created in 1987 and enhanced with additional features in 1992. An illustration of typical roller cone nomenclature is provided below in figure 1 and the standard nomenclature in table 2.

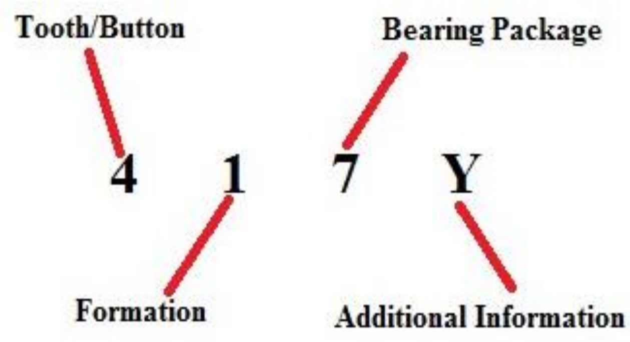


***Figure 1.*** *Standard Nomenclature for Roller cone bits [5].*

### 1.2.2. Classification of Polycrystalline Diamond Compact (PDC) Bits

One letter and three numbers make up the nomenclature for PDC drill bits. The letter, such as M for matrix, S for steel, or D for diamond, denotes the type of body. In the meantime, the type of formation that will be drilled is indicated by the first digit. The bit profile will be represented by digit number three, and the structure by digit number two. Table 3 shows the specifications for each of these digits.

## 1.3. Bit Selection Methods

Although bit performance may have a very small impact on the overall well cost (about 5 percent of the budget), bit cost may have a significant overall impact [6] Bit selection can be broadly divided into two groups:

1) Evaluation of Costs (Cost Analysis)
2) Analysis of Offset Well Logs.

### 1.3.1. Evaluation of Costs (Cost Analysis)

Typically, the cost analysis method is used to examine historical data from offset wells. Moreover, bit run monitoring uses it. An economical assessment of the bit's performance level will be made. But this approach takes up valuable time [8]. The estimation of bit cost is highlighted in the formula below.

*Table 1. Standard Nomenclature for Roller cone bits [5].*

| DIGIT 1 | | |
|---|---|---|
| **No** | **Type** | **Tooth/Button** |
| 1 | Soft | Tooth |
| 2 | Medium | |
| 3 | Hard | |
| 4 | Soft | TCI |
| 5 | Medium | |
| 6 | Medium Hard | |
| 7 | Hard | |
| 8 | Hardest | |

| DIGIT 2 | |
|---|---|
| **No** | **Formation** |
| 1 | Soft |
| 2 | Medium |
| 3 | Hard |
| 4 | Hardest |

| DIGIT 3 | |
|---|---|
| **No** | **Bearing** |
| 1 | NSRB |
| 2 | |
| 3 | SLRB |
| 4 | |
| 5 | |
| 6 | SLJB |
| 7 | |

| ALPHABET | |
|---|---|
| A | Air Application |
| B | Special Bearing |
| C | Center Jetted |
| D | Deviation Control |
| E | Extended Jets |
| G | Extra Gauge |
| H | Horizontal Application |
| J | Jet Deflection |
| L | Lug Pads |
| M | Motor Application |
| S | Standard Steel Tooth |
| T | Two Cone |
| X | Enhanced Cutting Structure |
| X | Chisel Inserts |
| Y | Conical Inserts |
| Z | Other Inserts |

*Table 2. Geological Formation Type Classification for Drilling [7].*

| S/N | Type | Description | PDC Cutting Structure |
|---|---|---|---|
| 1 | Soft and soft sticky | Highly drillable formations (Clay, marble, gumbo, unconsolidated sands) | Normal size |
| 2 | Soft-medium | Low compressive strength sands, shale and anhydrites with hard layers intermixed | |
| 3 | Medium | Moderate compressive strength sand, chalk, anhydrite and shale | |
| 4 | Medium hard | Higher compressive strength with non or semi sharp sand, shale, lime and anhydrite. | 19mm cutters |
| 5 | Hard | High compressive strength with sharp layers of sand or siltstone | 13mm cutters |
| 6 | Extremely hard | Dense and sharp formations, such as quartzite and volcanic rock | 8mm cutters |

$$Cost\ per\ feet = \frac{B+(T+t)R}{F} \quad (1)$$

Where $B$ is the Bit cost ($), R is the Rig cost per hour ($/hr), T, the trip time (hr), t, rotating time (hr), and F is the hole section.

Because drilling characteristics are not taken into account, the aforementioned procedure is not regarded as an effective way to choose a bit. Thus, alternative approaches will be considered [9].

### 1.3.2. Analysis of Offset Well Logs

W. J. Hightower in 1964 [10] created an intricate graphical depiction of the formation under study with the aid of gamma ray and spontaneous potential log data. This method, which is based on the compressive strength of the rock, compares various drilling bit types, drilling conditions, and lithology using sonic log and other lithology log data to be used in the selection of the suitable bit [11]. But bit selection is still based solely on past offset bit records, and current drilling data is still not taken into account. But current drilling data is still not taken into account, and bit selection is solely based on historical offset bit records. But current drilling data is still not taken into account, and bit selection is solely based on historical offset bit records.

## 2. Methodology

### 2.1. Data Collection and Analysis

The dataset utilized in this study comprises offset data from 2 wells drilled within the Niger Delta Field. The data were divided as follows: 80% for training, 10% for validation, and 10% for testing the ROP functions and conclusions drawn from the modelled bit.

The first stage was using field data to determine the Targeted IADC bit code. Using the Targeted IADC bit code that was obtained in Stage 1, Stage 2 involved predicting the ROP values. The third stage involves reusing the predicted ROP value in the data set to obtain the optimized predicted IADC bit code value. The Pearson correlation heatmap, shown in Figure 2, illustrates the relationships among the numerical variables in the dataset used. Positive correlation values indicate that an increase in one variable corresponds with an increase in another, whereas negative values reflect an inverse relationship between variables.

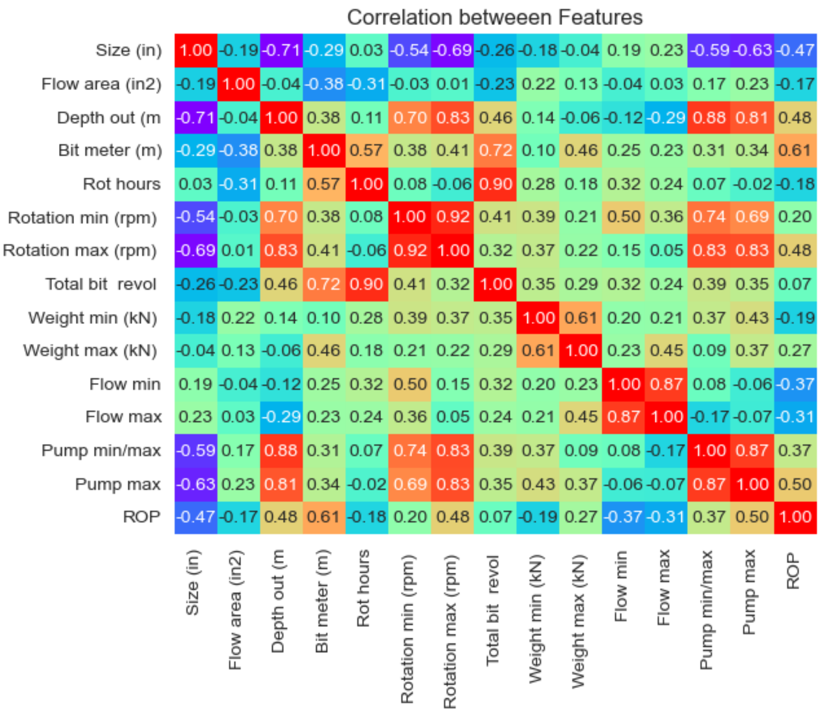


***Figure 2.*** *Correlation between numerical features in the dataset.*

### 2.2. Machine Learning Model Development

The machine learning workflow, representing the steps utilized for the development of the machine learning models used for this study is depicted in Figure 3. The data gathered was stored in an excel workbook format. The data was then exported to a python environment (Jupyter notebook), where all data processing and machine learning operations was carried out.

The models adopted were ANN, Random Forest (RF) and Extreme Gradient Boosting (XGB), which were all implemented in the python environment (Jupyter Notebook) using different libraries and their appropriate versions.

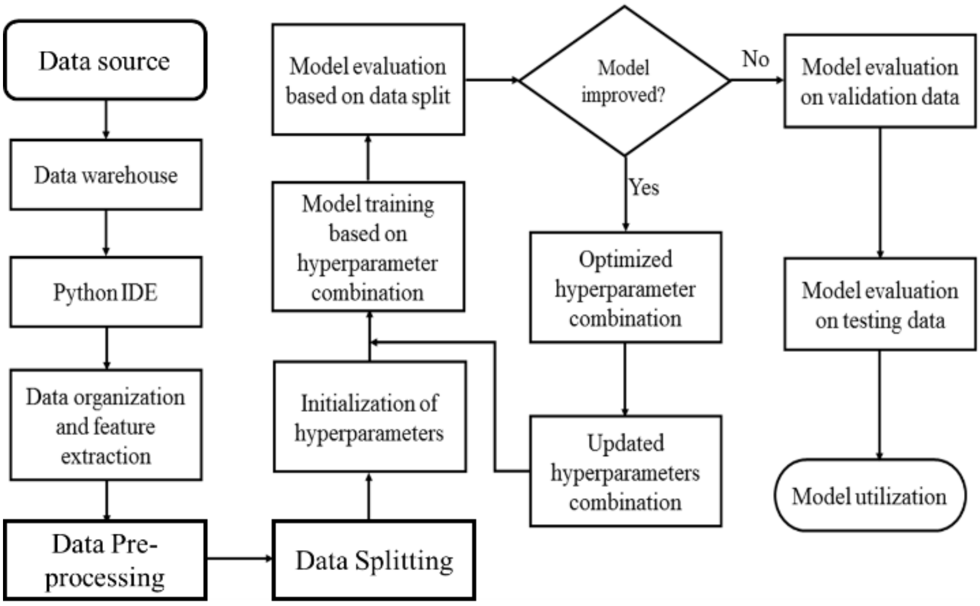


*Figure 3. Adopted Machine Learning Workflow [12].*

### 2.2.1. Random Forest (RF)

Random Forest is a machine learning technique widely used for building predictive models across various research domains. It operates on the principle of bootstrapping and aggregation, commonly referred to as bagging [13]. In predictive modeling, a common objective is to minimize the number of variables required for accurate predictions, thereby reducing data collection efforts and enhancing overall efficiency [14]. Within the Random Forest (RF) framework, each decision tree is trained on a randomly selected subset of the input data (a bootstrap sample), allowing it to grow independently. The model's high accuracy stems from the collective strength of combining predictions from multiple decision trees. The RF process comprises three main stages: first, generating *n* bootstrap samples from the input variables; second, building an unpruned regression tree for each sample using the optimal predictor split; and third, aggregating the predictions from all *n* trees to produce the final output [15].

For a random data subset $d(x, y)$, the decision tree iteratively partitions the variable space $x$ as samples with close targets are aggregated. The data at node m is represented by $d_m$ with $N_m$ samples. dm may be partitioned into subsets denoted as $d_m^{right}$ and $d_m^{left}$ using each candidate split $\theta(p, t_m)$ where $p$ and $t_m$ refer to a feature and threshold, respectively (Equation (2)).

$$\begin{cases} d_m^{left} = \{(x, y) |\, x_p \le t_m\} \\ d_m^{right} = d_m / d_m^{left} \end{cases} \tag{2}$$

The candidate split is defined as a loss function $H(\bullet)$, as shown in Equation (3).

$$\begin{cases} d_m^{left} = \{(x, y) |\, x_p \le t_m\} \\ d_m^{right} = d_m / d_m^{left} \end{cases} \tag{3}$$

$$\text{Where } \bar{y}_m = \frac{1}{N_m} \sum_{y \in d_m}$$

The minimization of Equation (3) yields the parameters $\theta(p, t_m)$.

$$G(d_m, \theta) = \frac{N_m^{left}}{N_m} H\left(d_m^{left}(\theta)\right) + \frac{N_m^{right}}{N_m} H\left(d_m^{right}(\theta)\right) \tag{4}$$

The recursion of Equation (4) continues for $d_m^{right}$ and $d_m^{left}$ until the maximum depth is achieved. The prediction of the RF model is thus obtained from Equation (5).

$$f(x) = \frac{1}{K} \sum_{K=1}^{K} DT_i(x) \tag{5}$$

where K is the number of decision trees (DT) in the random forest.

### 2.2.2. Extreme Gradient Boosting (XGB)

Gradient Boosting (XGBoost) is an efficient and scalable tree-based machine learning algorithm that has become widely adopted across numerous fields of data analysis. As an advanced form of gradient boosting, XGBoost is especially well-suited for both regression and classification problems. At its core, XGBoost leverages the boosting technique, where multiple weak learners are sequentially trained and combined using an additive model to produce a strong overall predictor. This iterative approach enhances predictive performance, reduces the risk of overfitting, and optimizes computational speed and resource usage [16].

The general function of the forecasting is set up at step $p$, as shown in Equation (6)

$$f_i^{(p)} = \sum_{k=1}^{p} f_k\,(x_i) = f_i^{(p-1)} + f_p\,(x_i) \qquad (6)$$

where $f_p\,(x_i)$ denotes the learner at step $p$, $f_i^{(p)}$denotes the prediction at $p$, $f_i^{(p-1)}$denotes the prediction at $p$−1, and $x_i$ denotes the input features.

To strike a balance between overfitting and computational efficiency, XGBoost employs a refined analytical formula—shown in Equation (7)—to assess the model's "goodness of fit" to the true underlying function. This method effectively regulates model complexity while enhancing predictive accuracy, thereby ensuring optimal performance.

$$Objective^{(p)} = \sum_{k=1}^{n} l\,(\bar{y}_i,\, y_i) + \sum_{k=1}^{p} \sigma(f_i) \qquad (7)$$

where $l$ presents the loss function, n presents the number of observations utilized, and σ presents the regularization term as represented in Equation (8).

$$(f) = \boldsymbol{\Theta}T{+}0.5\lambda\omega 2 \qquad (8)$$

where $\omega$ expresses vector scores in leaves, $\Upsilon$ expresses the minimal loss necessary to divide the leaf node further, and $\lambda$ expresses the regularization parameters.

### 2.2.3. Artificial Neural Network (ANN)

ANN is a computational model inspired by the structure and function of the biological neural networks that make up the brain. The ANN model used for this study was a feed-forward neural network consisting of n hidden layers each with a set of neurons that can estimate a given output response (y) from provided input data (x) as shown in Figure 3.

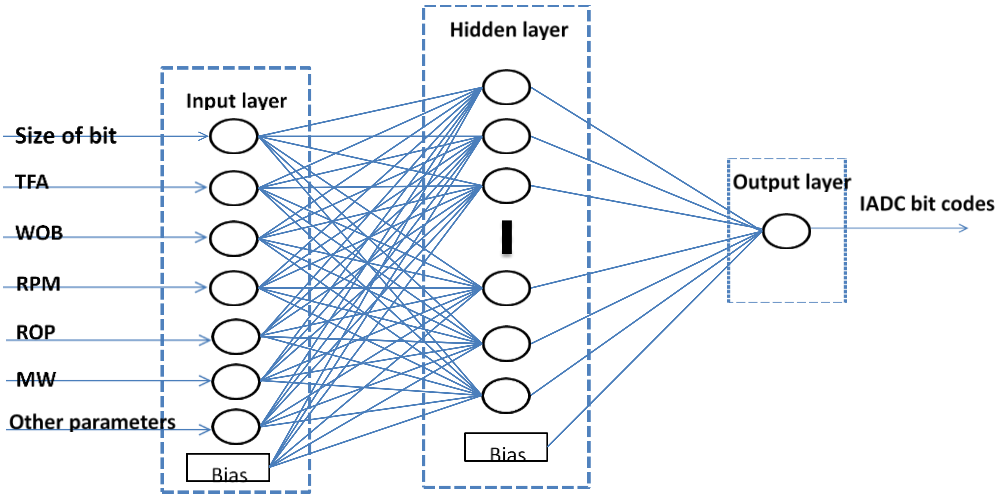


***Figure 4.*** *ANN Architecture for the implemented dataset [2].*

The mathematical concept of ANN involves a set of interconnected nodes or neurons that process and transmit information. Each neuron calculates a linear combination ($z$) of n input features ($x_i$) with corresponding weight ($w_i$) and bias ($b$) as shown in Equation 9. The weights of the connections between the neurons are adjusted during training to minimize the error between the predicted output and the true output. The selection of the right activation function is very essential in the ANN modelling process as it is chiefly responsible for the mapping of the inputs to the outputs.

$$z = \sum_{i=1}^{n} x_i w_i + b \qquad (9)$$

During the ANN process, the transformed version of the output data is forwarded from one hidden layer to another. The network is trained through an updating process that tunes the model weights to minimize model loss (L) which is typically taken as the root mean square error (RMSE) as shown in Equation 10 where $\hat{y}_i$ is the model prediction. The minimization is done through a gradient descent method where the weights are manipulated as a function of L and scaled using the learning rate (λ) as shown in Equation 11.

$$L = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (10)$$

$$w_{t+1} = w_t - \lambda\frac{\partial L}{\partial w_t} \quad (11)$$

### 2.2.4. Genetic Algorithm

The Genetic Algorithm (GA) is a population-based metaheuristic optimization technique inspired by the concepts of natural selection and genetics. It is particularly effective for addressing complex, nonlinear, and multivariable optimization problems where traditional methods may be inadequate. GA operates by evolving a population of potential solutions through genetic operations such as selection, crossover, and mutation, all driven by a fitness function that measures the effectiveness of each candidate solution [17]. In this study, GA was utilized to optimize the oil production rate predicted by the best-performing machine learning model, by fine-tuning input parameters such as gas injection rate, choke diameter, and other relevant variables. The overarching goal of GA in this context can be mathematically formulated as:

$$maximize \ subject \ to \ x \in \mathbb{R}^n \quad (12)$$

Where $f(x)$ is the fitness function—here, the predicted oil rate—and $x$ represents the set of controllable input parameters.

### 2.2.5. Model Evaluation

To evaluate the model's performance, two key metrics were employed: Classification Accuracy and the Coefficient of Determination ($R^2$).

Classification Accuracy: This metric measures the proportion of correct predictions made by the model out of all predictions. It is calculated as:

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \quad (13)$$

Coefficient of Determination ($R^2$): Commonly used in regression analysis, $R^2$ quantifies the proportion of variance in the dependent variable that is predictable from the independent variables. It is defined as:

$$R^2 = 1 - \frac{Sum\ of\ Squares\ of\ Residual\ (SSR)}{Total\ Sum\ of\ Squares\ (TSS)} \quad (14)$$

An $R^2$ value closer to 1 indicates that a greater proportion of variance is accounted for by the model, signifying a better fit.

## 3. Results and Discussion

The data was used to train the machine learning (ML) models, optimizing their respective hyperparameters to enhance predictive performance. Table 3 presents the optimal hyperparameters obtained for each model through a combination of 3-fold cross-validation and a full-factor experimental grid search. The performance of these models was evaluated using key statistical metrics, which are discussed in Section 3.2.

### 3.1. Hyperparameter Tuning

To ensure optimal model performance, GridSearchCV with 3-fold cross-validation was employed for hyperparameter tuning. The best-performing hyperparameters for each model are summarized in Table 3.

***Table 3.*** *Optimized Hyperparameters of Classification Models.*

| Model | Hyperparameters | Optimized value |
|---|---|---|
| ANN | Activation function | 5 |
| | Hidden layer sizes | 2 |
| | Epoch | 100 |
| | Learning rate | 0.005 |
| RF | n estimators | 500 |
| | max depth | 5 |
| | max features | 3 |
| | random_state | 9 |
| XGB | max depth | 5 |
| | gamma | 0.01 |
| | n estimators | 500 |
| | subsample | 1 |
| | random state | 9 |

### 3.2. Performance Evaluation

The classification task involved predicting IADC bit codes based on formation and operational features. Evaluation was based on classification accuracy and confusion matrix analyses. These metrics assess how well each model matches bit types to formation characteristics.

***Table 4.*** *Classification Accuracies of the Various Models.*

| Model | Accuracy |
|---|---|
| ANN | 0.91 |
| Random forest | 0.90 |
| XGBoost | 0.94 |

The confusion matrix offers a more detailed evaluation of classification performance beyond simple accuracy metrics. Figures 5 and 6 display the confusion matrices for the ANN and XGBoost models, respectively. In the ANN confusion matrix, classes 1, 2, 3, and 4 correspond to bit categories II5M, Category A, Category C, and M323. These visualizations reveal how frequently each model misclassified specific bit categories, providing valuable insight into their predictive reliability.

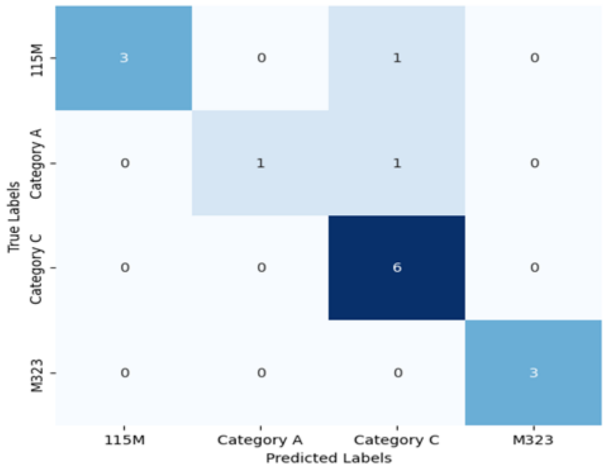


*Figure 5. Confusion Matrix of the ANN Classifier.*

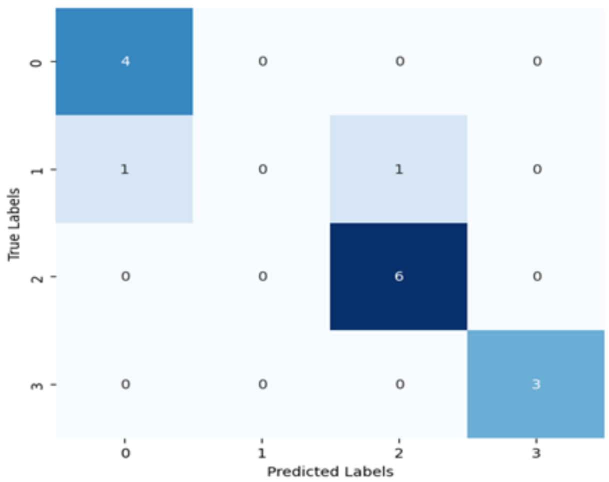


*Figure 6. Confusion matrix of the Xgboost Model.*

During preprocessing, it was observed that the target feature (IADC bit codes) included both alphanumeric and purely numeric codes. To enhance learning and reduce classification complexity, the numeric bit codes were grouped into broader categories. Grouping similar IADC codes into discrete categories allows the models to capture generalizable drilling characteristics associated with specific bit types.

*Table 5. Numerical IADC Bit Code Categories.*

| Numeric range | Category | IADC bit codes |
|---|---|---|
| 0-200 | Category A | 115, 135 |
| 201-400 | Category B | - |
| 401-600 | Category C | 415, 445 |

Category A represents softer formations or milder conditions. Category B represents Intermediate formation hardness, while Category C is suited for harder formations requiring robust bits.

After classification, the predicted bit category (or code) was used as an input feature for predicting ROP through regression models. The idea is that the selected bit has a significant influence on drilling performance, and hence, including it improves the ROP prediction.

Model performance was assessed using Root Mean Squared Error (RMSE) and R-squared ($R^2$) values. The hyperparameters of each regression model were tuned for optimal results.

*Table 6. Performance Metrics of Regression Models.*

| Model | RMSE | R squared |
|---|---|---|
| XGBoost | 1.236 | 0.975 |
| Random forest | 1.954 | 0.953 |
| ANN | 1.537 | 0.991 |

Figures 7 to 9 show the Actual vs. Predicted ROP plots for the three regression models. Visually, the ANN model tracks the actual data points most closely, with the least deviation, reflecting its superior predictive capability.


*Figure 7. Actual vs Predicted ROP using Xgboost Model.*


***Figure 8.*** *Actual vs Predicted ROP using ANN Model.*

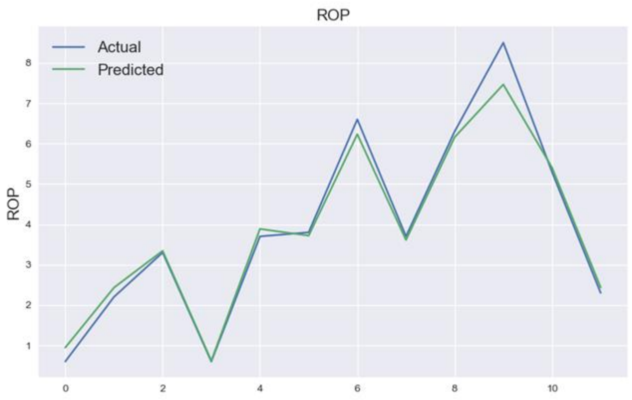


***Figure 9.*** *Actual vs Predicted ROP value using the RF model.*

The $R^2$ value of 0.991 achieved by XGBoost confirms its robustness in modeling complex nonlinear relationships between input parameters and ROP. Minor prediction errors are likely attributable to formation heterogeneity and real-time sensor noise.

It is evident from the aforementioned graphs that we can optimize the ROP value through the modelling process by utilizing all of the available data. We can then utilize the optimized ROP value to get the optimized IADC bit code. The accuracy of the R value is roughly 99 percent based on the graph's numerical values. A tiny error still exists, and it has specific causes.

## 4. Conclusions

This study successfully demonstrated the application of supervised machine learning techniques for bit selection and Rate of Penetration (ROP) prediction in drilling operations. By framing bit classification as a multiclass problem and coupling it with a regression task for ROP estimation, the workflow provided a comprehensive solution aimed at improving drilling efficiency.

Three machine learning algorithms—Artificial Neural Networks (ANN), Random Forest, and XGBoost—were implemented and compared across both classification and regression tasks. XGBoost consistently outperformed the other models, achieving the highest classification accuracy (97%) and the lowest root mean squared error (1.236) in ROP prediction, with an R-squared score of 0.991. This demonstrates the suitability of gradient-boosted ensemble methods for capturing the nonlinear relationships inherent in drilling data.

Despite these promising outcomes, the study is not without limitations. A key constraint was the size and scope of the dataset—data from only two wells were used. While the models achieved strong performance metrics, their generalizability across more diverse field conditions and formations remains untested. As such, care must be taken when extrapolating these results to broader drilling scenarios.

To build upon the insights of this research, the following directions are recommended:

Future studies should incorporate data from a larger number of wells across different geological settings. This would improve model robustness and enable better generalization.

Incorporating real-time data streams from drilling operations could enable the development of intelligent systems capable of dynamic bit selection and real-time ROP optimization.

Additional drilling and formation parameters—such as lithology type, mud properties, and bit wear indices—can be explored to further enhance predictive accuracy. Coupling machine learning predictions with advanced drilling simulators can provide a more holistic decision-support framework for well planning and execution.

## Abbreviations

| | |
|---|---|
| ROP | Rate of Penetration |
| XGBoost | Extreme Gradient Boosting |
| ANN | Artificial Neural Network |
| IADC | International Association of Drilling Contractors |

## Author Contributions

**Shedrach Igemhokhai:** Conceptualization, Methodology, Formal analysis, Writing original draft, Software, Visualization, Data curation, Review and Editing

**Kelani Bello**: Conceptualization, Review and Editing, Data curation

**Abiodun Olaoye:** Conceptualization, Review and Editing, Data curation

**Ehigie Momodu:** Conceptualization, Review and Editing, Data curation

**Abayomi Adejumo**: Review and Editing, Supervision

**Oladele Akindayini:** Review and Editing

## Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.


## Data Availability Statement

Due to confidentiality agreements with the data provider, the field data used in this study are not publicly available. However, a summary of the data analysis and results is included in this published article. Specific data requests may be considered on a case-by-case basis and subject to approval from the data provider. For inquiries, please contact the corresponding author.

## Conflicts of Interest

The authors declare no conflicts of interest.

## References


[1] Momeni, M., Hosseini, S. J., Ridha, S., Laruccia, M. B., and Liu, X. (2018). An optimum drill bit selection technique using artificial neural networks and genetic algorithms to increase the rate of penetration. *Journal of Engineering Science and Technology*, *13*(2), 361–372.

[2] Watalingam, P. (2014). *Bit Selection Using Drilling Data By Artificial Neural Networks*. *May*. http://utpedia.utp.edu.my/id/eprint/14233

[3] Bilgesu, H. I., Al-Rashidi, A. F., Aminian, K., and Ameri, S. (2001). An Unconventional Approach for Drill-Bit Selection. *Proceedings of the Middle East Oil Show*, 198–203. https://doi.org/10.2523/68089-ms

[4] Fasheum, M. (1997). *THE UNIVERSITY OF NOTTINGHAM DEPARTMENT OF MINERAL RESOURCES ENGINEERING by*. *April*.

[5] McGehee, D. Y., Dahlem, J. S., Gieck, J. C., Kost, B., Lafuze, D., Reinsvold, C. H., and Steinke, S. C. (1992). *The IADC Roller Bit Classification System*. https://doi.org/10.2118/23937-MS

[6] Jamshidi, E., and Mostafavi, H. (2013). Soft computation application to optimize drilling bit selection utilizing virtual inteligence and genetic algorithms. *Society of Petroleum Engineers - International Petroleum Technology Conference 2013, IPTC 2013: Challenging Technology and Economic Limits to Meet the Global Energy Demand*, *1*(October), 357–371. https://doi.org/10.2523/16446-ms

[7] Brandon, B. D., Cerkovnik, J., Koskie, E., Bayoud, B. B., Colston, F., Clayton, R. I., Anderson, M. E., Hollister, K. T., Senger, J., and Niemi, R. (1992). *Development of a New IADC Fixed Cutter Drill Bit Classification System*. https://doi.org/10.2118/23940-MS

[8] Rabia, H., Farrelly, M., and Barr, M. V. (1986). A New Approach to Drill Bit Selection. *Society of Petroleum Engineers of AIME, (Paper) SPE*, 421–428. https://doi.org/10.2118/15894-MS

[9] Bataee, M., Edalatkhah, S., and Ashna, R. (2010). Comparison between Bit Optimization Using Artificial Neural Network and Other Methods Base on Log Analysis Applied in Shadegan Oil Field. *Society of Petroleum Engineers - International Oil and Gas Conference and Exhibition in China 2010, IOGCEC*, *4*, 3097–3103. https://doi.org/10.2118/132222-MS

[10] Hightower, W. J. "Proper Selection of Drill Bits and Their Use." Paper presented at the SPE Mechanical Engineering Aspects of Drilling and Production Symposium, Fort Worth, Texas, March 1964. https://doi.org/10.2118/794-MS

[11] Onyia, E. C. (1988). Relationships Between Formation Strength, Drilling Strength, and Electric Log Properties. *Society of Petroleum Engineers of AIME, (Paper) SPE, OMEGA*. https://doi.org/10.2118/18166-MS

[12] Igemhokhai, S., Bello, K., and Adejumo, A. (2025). A machine learning approach to Intelligent Artificial lift method selection. *International Journal of Frontiers in Engineering and Technology Research*, *8*(2), 001–015. https://doi.org/10.53294/IJFETR.2025.8.2.0030

[13] M. Ali, R. Prasad, Y. Xiang, Z. M. Yaseen, Complete ensemble empirical mode decomposition hybridized with random forest and kernel ridge regression model for monthly rainfall forecasts, J. Hydrol. 584 (2020) 124647, https://doi.org/10.1016/J.JHYDROL.2020.124647

[14] Speiser, J. L., Miller, M. E., Tooze, J., and Ip, E. (2019). A comparison of random forest variable selection methods for classification prediction modeling. Expert Systems with Applications, 134, 93–101. https://doi.org/10.1016/j.eswa.2019.05.028

[15] Nosakhare, A., Igemhokhai, S., Aimhanesi, S., Ugbodu, F., and Iyore, N. (2024). Heliyon Data-driven intelligent modeling, optimization, and global sensitivity analysis of a xanthan gum biosynthesis process. Heliyon, 10(3), e25432. https://doi.org/10.1016/j.heliyon.2024.e25432

[16] Alshboul, O., Shehadeh, A., Almasabha, G., and Almuflih, A. S. (2022). Extreme Gradient Boosting-Based Machine Learning Approach for Green Building Cost Prediction. Sustainability, 14(11), 6651. https://doi.org/10.3390/su14116651

[17] Michell, M. (1998). An introduction to genetic algorithms [electronic resource]. Mit Press. http://books.google.com/books?id=0eznlz0TF-IC