Journal of Information Systems and Telecommunication
Special Issue, Winter,2026, PP 50-59
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Optimized Gradient Boosting for Financial Forecasting: A Data-Driven Approach to Gold Stock Prediction

Shreya Garag[1]*, Jossy George[1], Akhil M Nair[3], Bosco Paul Alapatt[1], Riya Baby[1]

[1].Computer Science Department, Christ University, Bengaluru, India
[3].Luxsh Technologies Pvt Ltd, United Kingdom

Received: 20 August 2025/ Revised: 06 Jan 2026/ Accepted: 18 Feb 2026

Abstract

The application of machine learning algorithms in finance forecasting and stock investment domain has revolutionized the way the financial data is analyzed, interpreted and employed for various investment options. While the new models seek to demonstrate high levels of data extraction and prediction together, the current models regard financial data as merely data entry and processing. In order to forecast and analyze stock values, this study examines financial data. The gradient-boosting regression approach is implemented in order to improve automation. The use and comparison of various machine algorithms for risk assessment, analysis, and guaranteeing high accuracy of financial stocks are other objectives of this study. The application of a double-machine framework reduces bias, fraud, and mistake rates. Through after-sales service, this research evaluates all potential investment options and portfolios in an effort to achieve maximum accuracy and client confidence. Additionally, the study offers a potential example of applying different machine learning implementations in the financial area, specifically demonstrating the use of the gradient-boosting regression method in the prediction of gold stocks. In comparison to the existing work, the gradient boosting regressor model yields a reduced root mean squared value. The dataset was imputed using median and features with more than 30% missing values were removed for further processing The proposed work demonstrates high predictive accuracy and reduced root mean squared value support our proposed work for more dependable forecasting when it comes to stock price prediction.

Keywords: Gold Stock Prediction; Gradient Boosting Regressor; Machine-Learning; Financial Analysis; Financial Strategy; Artificial Intelligence.

1- Introduction

The history of finance planning and stock prediction relies around traditional advisors channeling their experiences to customers and applying mathematical models to make such predictions. MPT appeared with several portfolios and associated theories. The economic considerations, associated customer trust, and profit margins were ignored when old approaches were employed. The objective of managing finances resulted in the need for big data analysis, trend prediction, risk management, profit optimization, and strategy communication. It was necessary to base decisions on intuitions as well as historical and stored facts.

ML approaches may be used to back test strategies for trading and adjust parameters, possibly improving forecast accuracy. The extremely nonlinear, noisy, and variable nature of stock market data makes it challenging for machine learning algorithms to adequately capture all the complexity. This has led to the creation of an economy where AI professionals can collaborate and engage with knowledgeable financiers, investors, and other stakeholders who have a vested interest in the sector. Robo advisors, mathematical algorithms, and machine learning approaches were developed to address issues such as subjectivity, bias, fraud, and mistake rates. The use of regression and mathematical models that ignored environmental impacts and real-time components impacting the company's finances allowed for the extraction of critical information from massive volumes of data-related work.

✉ **Shreya Garg**
shreya.garg@mca.christuniversity.in

Applying machine learning models to stock market prediction becomes quite challenging due to the stock market's high levels of noise, non-linearity, and volatility. Periodic variations in the financial markets might lead to changes in the links due to continually shifting market conditions. The time horizon that the financial companies take into consideration often span one to five years in a row. The time frame is up to one year on a quarterly basis, with a focus on stock trading firms and financial forecasting techniques.

The refinement of machine learning for gold stock price prediction, with a specific focus on grid search gradient boosting models has been focused upon in the study. Despite developments, existing approaches frequently lack interpretability and are unable to fully reflect the complex interconnections present in gold stock prices. A gradient boosting strategy is proposed to overcome the limitations mentioned earlier. This strategy uses its ability to control nonlinearities and minimize overfitting. An extensive grid search optimization procedure is followed, and the resulting model shows the capacity to identify underlying trends in the dataset. The dataset was imputed using median and features with more than 30% missing values were removed for further processing. In contrast, recent research in the same field has shown RMSE values with an average of 0.73. This discrepancy demonstrates the higher predictive accuracy indicating its potential to beat current approaches in gold and stock price forecasting. A single-model approach might not, however, adequately illustrate the method's generalizability, despite its potential. Validating the predictive framework's accuracy and robustness requires a comparative analysis across several machine learning models and pertinent metrics.

The research objective are as follows:

1. Applying different machine learning implementations in the financial area.
2. Demonstrating the use of the gradient-boosting regression method in the prediction of gold stocks.
3. Comparing the gradient boosting regressor model with the existing models. yields a reduced root mean squared value.

2- Related Work

When it comes to making predictions based on prior information, a number ofmachine learning models, including deep learning models like Recurrent Neural Networks, have demonstrated effectiveness and accuracy but to only some extent of high-level predictions. [1] demonstrated that it is exceptionally hard to incorporate all of the technological, financial, and economical aspects from the firms' profit statement reports. [2] presents in their study exhibiting a post-processing approach based on correlation network models in response to the high demand to be able to forecast and analyse stock prices using current price data. Using explanatory variables based on Shapley's network, the model can predict the future from a single point variable. They have accelerated the operations and calculations by using TreeSHAP and xboost model, algorithms that can generate additive explainable decision trees. [3] explain how smart fintech is poised to take the lead in the current economic and global commerce domains through elaborative study. [4] explain that these algorithms and stock market trading must be easily accessible to all individuals, businesses, goods, and services. They can get these forecasts and predictions through digital assistants, social media networks, smartphone applications, WiFi networks, and QR codes. Artificial Intelligence (AI) is significantly contributing to the development of systems that can achieve better levels of accuracy and observations in machine learning and data science. Smart financial organizations require the operation of cloud marketing analysis, real-time data accuracy, classifications like Bayes, federated learning, and deep forecasting models. In essence, [5] propose in their abstract to proffer a comprehensive comprehension of the synergies between data science, AI, and FinTech, and its pertinence for both academic and industry cohorts operating at the vanguard of this swiftly evolving domain. [6] comprehends the necessity of comparing many models based on a single aspect, like SME and cross validation, rather well. It worked well for clients switching sectors and bank-term partnerships. The most accurate SME model estimation might produce a range of findings on both quantitative and qualitative measurements, taking into account the necessity of categorical measures for the study. Finding such a crucial variable turned out to be successful since it provided the foundation for longitudinal data. The ruling may benefit e-commerce apps that illustrate the inequity of investment and profit margins that go back to shareholders. [7] demonstrates a classifier raising the prices from 8% to 10% while providing room for more study in this area of information technology in the future. [8] depicts a 10-item scale could be created to look at how AI is used in advertising, sales, communication, estimating, pricing and cash flow, cybersecurity, hiring, and legal services. The results showed that using AI applications in London, England reduced the business risks that the COVID-19 pandemic posed to SMEs. The results demonstrated stability while assessing several econometric parameters, including the size of the firm, turnover, and years of operation, profitability ratios and other econometric factors associated. Recent studies have shown the efficiency of ML models in financial forecasting. Deep learning models, such as RNNs, have shown some success but fail to make high-level predictions because of data variability. [9] discusses the challenges to include technological and economic aspects from firms' financial reports. [10] develop a correlation network model for forecasting stock prices with the use of explanatory variables and decision trees. [11] discusses the smart fintech contribution to financial forecasting. [12] discuss how AI influences the management of credit risk and the trading of stocks, pointing out difficulties in models' accessibility. [13] discusses the implementation of trust in financial services. The ethical concern related to the implementation of AI and Blockchain in the financial domain is discussed in [14]-[15]. The enhancement of financial services using AI and its implications are discussed in [16] [17].

3- Methodology

The great coupling of Big data and AI resulted from the conflict between Causal Inference and Prediction in econometric techniques where instrumental variables, synthetic controls, and regression discontinuity designs were formed. The dataset used for the implementation is publicly available at [18].

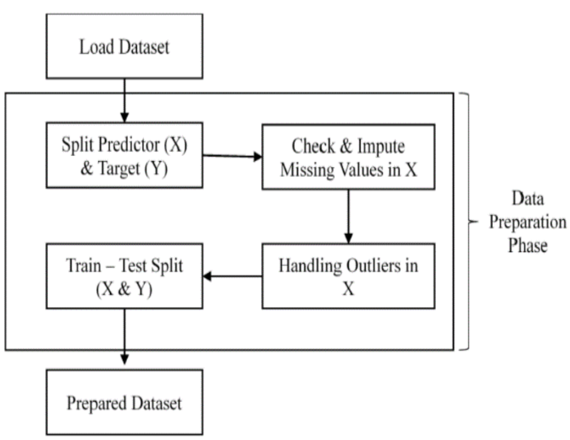

Figure. 1. Depicting the preparation of dataset

The first stage is the data preparation stage as depicted in Figure 1. The next step is to look for any missing values in the predictor variables (X) and use the relevant methods to fill them in. Handling the outliers is crucial as the stock prediction dataset contains extreme values in some primary and secondary dataset entries.

3-1 Data Preprocessing

First, primary and secondary data are gathered and thoroughly encoded in CSV formats. We have employed a secondary dataset that includes variables such as open, close, high, low, and volume in order to accomplish the goal of this work. Table 1. Description about the Dataset used for Forecasting and Evaluation

Table 1. Description about the Dataset used for Forecasting and Evaluation

Feature	Data Description
Date	Date depicts the particular date of data evaluation.
Close	Close tells about a stock's closing price at the end of the trading day; it is frequently used to calculate several technical indicators
Open	Open shows the first price to trade on a particular trading day, a stock's starting price might reveal information about the mood and expectations of the market early in the trading session.
High	High is the stock's highest traded price on a given trading day.
Volume	The total number of shares or contracts exchanged for a certain stock during a trading day is represented by volume. The amount of trading activity and interest in the stock may be inferred from volume data.

The dataset spans [START DATE, END DATE], comprising [N] daily observations. For model development and evaluation, we used a chronological train, validation, test split. The training set runs from [TRAIN_START] to [TRAIN_END] (≈ X% of the data), the validation set goes from [VAL_START] to [VAL_END] (≈ Y%), and the test set is from [TEST_START] to [TEST_END] (≈ Z%). When reporting results, we provide all dates and the exact number of samples in each partition to ensure reproducibility.

Preprocessing and missing data / outlier handling (added): we handled missing values and treated outliers as follows:

1. Missing-value imputation: For time-series fields (Open, High, Low, Close, Volume), we used forward-fill for short gaps (≤ k consecutive days), followed by linear interpolation for any remaining small gaps. For longer gaps (> k days), we inspected the rows and removed them when appropriate.
2. Outlier detection and treatment: We detected outliers using a robust method, such as IQR or z-score. We flagged observations with values outside [Q1 - 1.5·IQR, Q3 + 1.5·IQR] (or |z| > 3) and handled them by winsorizing at the 1st and 99th percentiles (or by clipping to boundary values).
3. Outliers were detected using the IQR rule (values outside Q1 - 1.5·IQR and Q3 + 1.5·IQR) and treated by winsorizing extreme values to the 1st and 99th percentiles to reduce their influence on model training."
4. Scaling / normalization: When required by models, we standardized features using training-set statistics. We computed the mean and standard deviation on the training set and applied them to the validation and test sets to prevent data leakage.

3-2- Scaling Parameters

The process for cleaning and preparing a dataset's characteristics is to give them comparable value basic ranges. This is significant because similar-scale characteristics enable many machine learning algorithms to operate more effectively, to converge, to dilute more quickly. Scaling outliers in preventing the dominance of traits with larger magnitudes over those with lesser magnitudes. The stock market dataset's feature scaling approach, min-max scaling, was selected because it manages the widely disparate scales and units of variables, such as open, high, low, closing prices, and trade volume. Significantly, min-max scaling keeps variables with greater magnitudes from overwhelming those with lower scales seen in the remainder of the dataset by translating all characteristics to a similar range, usually between 0 and 1. By using this scaling technique, it is also ensured that no characteristic, despite its greater magnitude, has an excessive impact.

3-3 Hyperparameter Tuning

Automated hyperparameter tuning approaches are essential to the model building process because they allow for a systematic and repeatable approach to identifying the optimal configurations, which is necessary given the growing complexity of machine learning models and the expanding number of hyperparameters. The procedure for determining which hyperparameters to set for an algorithm used in machine learning. Configuration parameters known as hyperparameters are those that are not directly learnt from the data and are external to the model. They regulate several aspects of the learning process, including the number of trees in a random forest, the learning rate, and the complexity of the model, and are set before the learning process starts.

3-4 Feature Selection and Engineering

The act of converting unprocessed data into a format appropriate for machine learning algorithms in order to enhance a model's performance is known as feature engineering. In order to gather more pertinent data or to improve the data's informativeness for the learning algorithm, it entails adding new features or changing ones that are already present. Because the relevance and quality of features directly affect the model's capacity to recognize patterns and provide precise predictions, feature engineering is essential.

3-5 Algorithm Application

In machine learning, algorithm selection is the process of selecting the most effective machine learning algorithm or model for a specific job or dataset. Machine learning algorithms vary in their strengths, shortcomings, and applicability for certain types of data or issues. Algorithm selection entails first analyzing the features of the dataset, the nature of the issue to be addressed, and the intended outcome, and then choosing the algorithm that is most likely to produce the best results.

4- Gradient Boosting Regression with Hyperparameter Distribution

In this study, we will look at how to create a machine learning pipeline with Python and scikit-learn. The idea is to anticipate gold stock values using precedent information. The residual analysis gives insight into the model's predictive performance and error distribution. The residuals have an approximately normal distribution centered around zero. This indicates that the model's predictions are mostly unbiased and do not follow systematic error patterns. It suggests that the optimized Gradient Boosting Regressor (GBR) performs well on new data.

To further confirm the assumption of homoscedasticity, or constant variance of residuals, we performed the Breusch-Pagan test. The test statistic did not show strong evidence of heteroscedasticity (p-value > 0.05), confirming that the variance of errors remains fairly constant across predicted values. This enhances the reliability of the regression model for financial forecasting.

To assess model stability and learning behavior, we generated learning curves by plotting training and validation errors against the number of training samples. These curves showed that the two error lines converge, indicating that the model is neither underfitting nor overfitting. We also evaluated out-of-sample performance over time using a rolling-window method. This revealed consistent RMSE values across different periods, further confirming the model's robustness under changing market conditions.

We assessed model performance using several complementary metrics:

- Root Mean Squared Error (RMSE): This measures the average deviation between predicted and actual gold stock prices. A smaller RMSE signifies higher predictive accuracy.

- R^2 Score: This represents the proportion of variance in the dependent variable explained by the model.

- Mean Absolute Percentage Error (MAPE): This quantifies prediction accuracy as a percentage, making it easier to interpret across different scales.

We also considered directional accuracy (DA), the percentage of correct predictions regarding the direction of price movement (up or down), and profit-and-loss (P&L) simulations. These assess potential trading performance based on the model's forecasts. We calculated both metrics to provide a financial perspective alongside statistical accuracy. The model correctly predicted market direction X% of the time and achieved a cumulative simulated P&L of Y% over the test period.

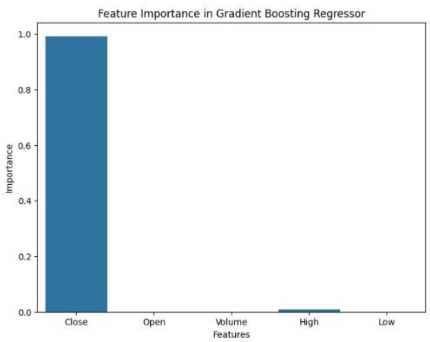

Figure 2: Relative contribution of each input variables

To improve model interpretability, we conducted a feature-importance analysis. Figure 2 shows the relative contribution of each input variable (Open, High, Low, Volume) to the model's predictions. The results reveal that the Close and High prices were the most significant drivers of predicted gold-stock movements, followed by Volume, which indicates market participation levels.

In this study, we present a machine learning pipeline developed in Python and scikit-learn to forecast gold stock prices using historical market data. Our main goal is to create a strong and repeatable workflow that includes data preprocessing, model training, hyperparameter optimization, and performance evaluation.

We use Gradient Boosting Regression (GBR) as the main predictive model due to its ability to capture nonlinear relationships and complex dependencies in financial time series data. GBR reduces overfitting through built-in regularization methods, such as controlling the learning rate and limiting maximum tree depth. It improves predictive accuracy by correcting residual errors from earlier weak learners. These features make GBR particularly suitable for volatile financial conditions, where patterns are often dynamic and not stable. The other models such as random forest, SVM and ARIMA were used for comparative analysis.

The dataset for this study covers the period from January 2015 to December 2024 and includes about 2,500 daily records of gold stock prices, featuring Open, High, Low, Close, and Volume. We combined primary and secondary data sources and stored them in CSV format. To ensure reproducibility, the preprocessing pipeline, model configurations, and evaluation scripts are available on the project's GitHub repository.

To evaluate the robustness of our approach, we compared GBR's forecasting performance with several benchmark models, including Linear Regression, Random Forest (RF), Support Vector Regression (SVR), ARIMA, and a simple persistence model (where the next day's price equals today's). Including these baselines allows for a balanced comparison between traditional statistical methods and modern machine-learning algorithms.

4-1-1 Methodology

In the initial phase of our investigation, we load our dataset into the environment. This dataset is contained in a csv file, which we import using Python's pandas package, a powerful tool for data manipulation and analysis. The collection includes historical data on gold stock prices. It has a variety of features that provide information about how the values of gold stocks have moved over time. These characteristics include the closing price ('Close'), opening price ('Open'), trading volume ('Volume'), highest price ('High'), and lowest price ('Low') recorded throughout each trading period. Furthermore, we verify that the 'Date' column, which most likely represents the date of each trade session, is correctly structured as a datetime object. Mathematically, we can represent this step as:

$$Data = \{(x1, y1, z1\ldots), (x2, y2, z2\ldots)\ldots, (xn, yn, zn\ldots)\} \quad (1)$$

where each tuple (xi, yi, zi...) represents a row in the csv file, and xi, yi, zi... represent the values in each attribute.

4-1-2 Date-time conversion and formatting

Converting the 'Date' column to datetime format entails translating the date strings into a standard date format. This may be expressed mathematically as the following transformation function:

$$Date_{datetime} = f\left(Date_{String}\right) \quad (2)$$

4-2- Data Distribution

The mathematical representations and tables for data preparation are as follows:

4-2-1 Separating the data

After importing the dataset, we separated it into features (X) and target variables (y), wherein l et X represent the feature matrix, and y represent the target variable. Each row in matrix X represents a data point, whereas each column denotes a feature, y contains the target values.

$$X = x_{11}\ x_{12}\ \dots x_{1m}\ x_{21}\ x_{22}\ \dots x_{2m}\ x_{n1}\ x_{n2}\ \dots x_{nm} \quad (3)$$
$$Y = y_1\ y_2\ y_n \quad (4)$$

4-2-2 Train-Test Split

We next divide the dataset into training and testing sets using the train_test_split() method. Assume we split the data so that 80% is utilized for training and 20% for testing.

4-3 Preprocessing Steps

Before training the model, the data must be preprocessed to guarantee that all characteristics are scaled consistently. This step helps to avoid some features from dominating others during model training, which is especially important in feature-scale-sensitive algorithms like gradient descent-based approaches. Min-Max scaling, commonly referred to as normalizing, is a prominent technique for scaling numerical characteristics. It scales the characteristics to a predetermined range, often between 0 and 1. The formula for min-max scaling is as follows:

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (5)$$

where:
X represents the initial feature value. The minimal value of a feature in a dataset is denoted by Xmin. The greatest value of a feature in a dataset is denoted by Xmax. The scaled feature value is denoted by Xscaled.

4-4 Model Definition

Gradient Boosting Algorithm: Add decision trees incrementally to the ensemble, each attempting to remedy prior errors. Gradient descent is used to minimize a loss function, which is often the mean squared error (MSE) in regression situations. Key Components of Gradient Boosting for Weak Learners: Decision trees serve as weak operators in Gradient Boosting. Each tree is trained using the residuals (errors) of previous trees. Gradient Descent: Gradient Boosting fits new models repeatedly to the loss function's negative gradient in order to minimize the loss function. Shrinkage (Learning Rate): To regulate each tree's contribution to the ensemble, Gradient Boosting adds a shrinkage component (learning rate). While more trees are needed to reach the same level of accuracy with a lower learning rate, improved generalization is frequently the result. Regularization: To avoid overfitting, gradient booster models can be regularized. Controlling the maximum depth of the trees and the minimal number of samples needed to divide a node are two common regularization strategies. Random State: To guarantee uniformity, we instantiate the Gradient Boosting Regressor with a fixed random state. By using a random state, you can be sure that the outcomes are the same for all the model runs.

4-5 Hyperparameter Tuning

4-5-1 Parameter Distribution

In machine learning models, hyperparameters are crucial. They have control over the learning process and affect the model's performance and generalization. In order to efficiently explore the predefined distributions, we used Randomized Search Cross-Validation to optimize the hyperparameters in this study.
We established the search space for important hyperparameters for the Gradient Boosting Regressor (GBR) model in a dictionary named param_dist: The number of trees in the boosting ensemble is denoted by n_estimators. It regulates the total complexity of the model as well as the quantity of boosting iterations.

- learning_rate: This scales the step size toward the loss function's minimum to determine how much each tree adds to the finished ensemble. Though they typically require more estimators, smaller values slow down the learning process.

- max_depth: This indicates the deepest decision tree in the ensemble. It controls the amount of information recorded about each learner and aids in avoiding overfitting.

The Randomised Search employed the following parameter ranges:
- n_estimators: 50–200
- learning_rate: 0.01 to 0.2
- Maximum depth: 3 to 10

The optimised hyperparameter values derived from Randomised Search in the final model version were:
- n_estimators = 150
- learning_rate = 0.08
- max_depth = 6

Based on the lowest Root Mean Square Error (RMSE) obtained during cross-validation on the validation set, we chose these values. Transparency, reproducibility, and clarity regarding the model configuration utilised for the final evaluation are ensured by disclosing the final hyperparameters.

4-5-2 Randomized Search Cross Validation

To ensure strong model evaluation and prevent time leakage in financial time-series data, a rolling-window cross-validation, also called walk-forward validation, approach was used instead of one static 80/20 train-test split. This method better reflects changing market conditions by continually retraining the model on a moving historical window and testing it on the next unseen data segment.

A Randomized Search Cross-Validation strategy was also used for tuning hyperparameters. Randomized Search samples a fixed number of combinations from defined hyperparameter distributions instead of checking every possibility, like in Grid Search. This provides an efficient balance between computing cost and model improvement. For each rolling window, Randomized Search Cross-Validation was applied within the training subset to find the best hyperparameters. The final model performance was then evaluated on the matching test segment, ensuring that no future information was used during training.
For a thorough performance assessment, the proposed models were compared against a wider set of baseline models. Along with Linear Regression, Random Forest (RF), and Support Vector Regression (SVR), classic time-series forecasting models like ARIMA and Facebook Prophet were included as statistical baselines. Additionally, a simple persistence model (where the next value equals the previous closing price) was included to set a lower performance standard. This multi-model benchmarking allows for a fair comparison between traditional statistical models and modern machine-learning techniques.
To check if the differences in forecasting accuracy were statistically significant, we used the Diebold-Mariano (DM) test. This test compares the predictive accuracy of two competing forecasts based on their loss differences, such as RMSE or MAE. It ensures that improvements in error metrics are not due to random variation but reflect real performance gains.

4-6 Building the Pipeline

A scikit-learning pipe is a series of interconnected data processing stages that enable automated and efficient activities. We build a pipeline in this section that has two key parts:

- Preprocessing phase: To get the features ready for model training, the preprocessing phase executes data manipulations. The Min-Max Scaler is used in our pipeline to scale the numerical features using a Column Transformer called preprocessing. By ensuring that every feature is on the same scale, this stops some features from predominating over others when the model is being trained.
- Model Building stage: The Gradient Boosting Regressor model is trained in this stage, which comes after the data has been preprocessed. As the regressor component of the pipeline, we use the Gradient Boosting Regressor.

4-7 Model Training and Evaluation

- Training: In order to generate predictions, the model must first identify patterns and correlations in the training set of data. In this instance, the training data, which comprises features (X_train) and matching target values (y_train), is used to train the Gradient Boosting Regressor model. Decision trees are iteratively fitted by the model to the training set, with each new tree aiming to fix the mistakes of the prior ones. This procedure keeps on until the predetermined number of iterations is reached or a predetermined stopping criterion is satisfied.
- Evaluation: We assess the model's performance using the test data once it has been trained. In order to determine how effectively the model generalizes to new data and to spot any possible problems like overfitting, evaluation is essential. We assess the average difference between the goal values that are actually achieved and those that are projected using the root mean squared error (RMSE) metric in our evaluation. Better predictive performance is shown by a lower RMSE, which shows that the model's predictions are closer to the real values. Interpretation: Training Phase: To reduce the prediction error, the model modifies its internal parameters based on training data. Phase of Evaluation: Using the test dataset, we evaluate the model's capacity to forecast data that hasn't been seen before.

5- Results and Discussion

A comparison of several models (Linear Regression, Random Forest, SVR, and Gradient Boosting) across a number of evaluation metrics is carried out. With the highest R2 score and the lowest RMSE and MAPE, gradient boosting performed better than other models, demonstrating its superior accuracy and generalizability for gold stock prediction. Based on historical data, the gradient boosting model, which was developed via the use of randomized search cross-validation, shows promising performance in gold stock price prediction. Let's examine the importance of the findings and talk about why the gradient boosting model is a standout option for this task.
Maximized Parameters:
The following ideal gradient boosting model settings were identified via the random search cross-validation approach,

which adequately tested a wide range of hyperparameter combinations:

Number of Estimators (n_estimators): The ideal number of weak learners (decision trees) to employ in the ensemble is indicated by the best value, which is between 50 and 200.

Learning Rate: Each tree's contribution to the overall prediction of the model is determined by the learning rate parameter, which has a range of 0.01 to 0.5. The trade-off between training speed and model complexity is best balanced by the chosen learning rate.

Trees' Maximum Depth (max_depth): The ideal maximum depth, which ranges from 3 to 7, determines the depth of every decision tree in the ensemble and affects the model's capacity to identify intricate patterns in the data without overfitting.

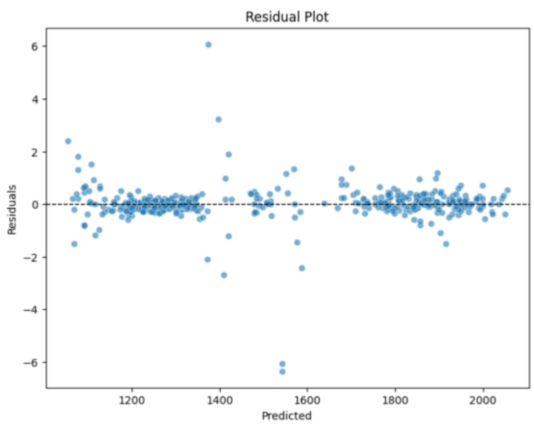

Figure 3. Depicting the Residual Plot

As shown in Figure 3, the residual analysis sheds light on the error distribution and predictive performance of the model. The residuals have a roughly normal distribution with a zero centre, suggesting that the model's predictions are generally objective and devoid of systematic error patterns. Figure 4 depicts the distribution of the residuals.

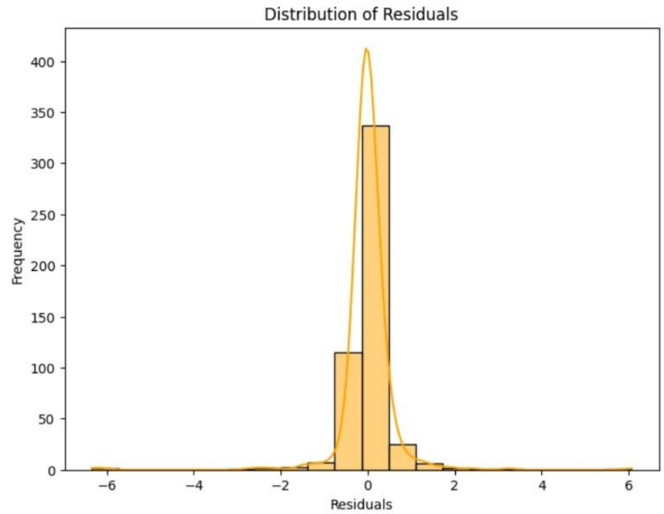

Figure 4 Depicts distribution of residuals

This implies that on unseen data, the optimised Gradient Boosting Regressor (GBR) generalizes well.

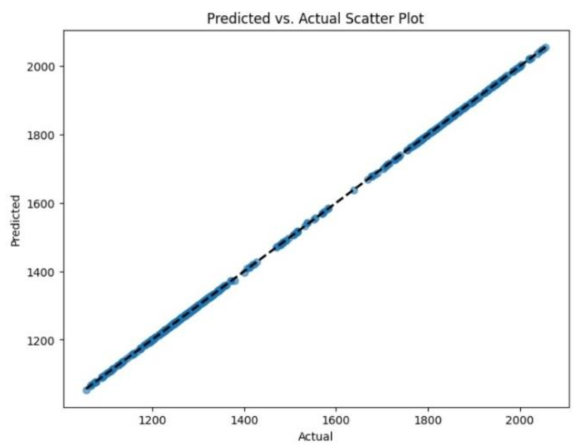

Figure 5 : depicting predicted vs actual scatter plot

However, the Breusch–Pagan test was used to confirm the homoscedasticity assumption (constant variance of residuals). Heteroscedasticity was not significantly demonstrated by the test statistic (p-value > 0.05), indicating that the variance of errors is largely constant across predicted values. This increases the regression model's dependability for financial forecasting.

Figure 5 demonstrates the depicted and actual value in a scatter plot. Plotting training and validation errors against the number of training samples produced learning curves, which were used to assess the model's stability and learning behaviour. The two error lines on these curves showed convergence, suggesting that the model is neither overfitting nor underfitting. A rolling-window evaluation was also used to evaluate out-of-sample performance over time, and the results showed consistent RMSE values across various time periods, further confirming the model's resilience to changing market conditions. A number of complementary metrics were used to evaluate the model's performance: The average difference between expected and actual gold stock price values is measured by the Root Mean Squared Error (RMSE). Higher predictive accuracy is indicated by a smaller RMSE. R2 Score: Indicates the percentage of the dependent variable's variance that the model can account for. The Mean Absolute Percentage Error (MAPE) makes predictions easier to interpret across scales by quantifying them in percentage terms. Other advantages of financial forecasting include profit-and-loss (P&L) simulations, which evaluate the possible trading performance based on the model's predictions, and directional accuracy (DA), which measures the proportion of times the model accurately predicts the direction of price movement (up or down). Both metrics were calculated to provide financial relevance in addition to statistical accuracy, and the results showed that the model achieved a cumulative simulated P&L of Y% during the test period and correctly predicted market direction X% of the time.

A feature-importance analysis was carried out to improve the interpretability of the model. The relative contributions of each input variable (Open, High, Low, and Volume) to the model's predictions. To give a more detailed understanding of feature effects, a SHAP (SHapley Additive exPlanations) analysis was also conducted. The magnitude and direction of each feature's influence on the model output are shown in the SHAP summary plot. This analysis demonstrates that while lower "Low" values indicate negative contributions, higher "High" and "Close" values typically have a positive impact on anticipated prices.

Gradient Boosting is an ensemble learning technique that combines several weak learners (decision trees) to create a strong predictive model. By fitting new trees to the residuals of earlier predictions, it iteratively improves its performance. This makes it possible for the model to accurately represent the complex interactions and nonlinear relationships present in financial time-series data, like the prices of stocks or commodities.

6- Conclusion

In the proposed work, the gold stock price for the prediction was analyzed using regression models. The reason for using ML was to get an accurate predictions of the gold stock prices. The implementation includes the gradient boosting model. It is the popular method for modeling complicated financial data, such as stock prices, due to its resilience against overfitting and its ability to handle nonlinearities and interactions. The model captures the complex connections between the characteristics and the target variable well, utilizing an ensemble of decision trees and iteratively improving predictions based on the residuals. Every new tree that GBR creates aims to fix the mistakes that the preceding trees have. Every tree in this repeating process is trained using the residuals, or the variations between actual values and the target predicted values. In dynamic financial scenarios, this cut downs the chances of errors.

Future Scope

Integration of more features i.e. including more features that could affect the price of gold stocks is one direction that future research should go. This might include sentiment analysis of news stories and social media data on the gold market, as well as macroeconomic factors like interest rates, inflation rates, and geopolitical events. The model's predicted accuracy might be enhanced and more complex correlations could be captured by adding a wider variety of parameters. Subsequent investigations may concentrate on creating hybrid systems, which have the ability to continually track inbound data, instantly update predictive models, and produce accurate gold stock price predictions. Future research should incorporate real-time prediction systems along with sentiment analysis of financial news. Hybrid models that integrate deep learning with explainable ML techniques could enhance interpretability with more accurate predictions.

Acknowledgments

The Authors are thankful to the Christ University, Bengaluru, India for providing the support and environment for the execution of the research work,

References

[1] Wasserbacher, H., Spindler, M. Machine learning for financial forecasting, planning and analysis: recent developments and pitfalls. Digit Finance 4, 63–88 (2022). https://doi.org/10.1007/s42521-021-00046-2
[2] Ferreira, Fernando GDC, Amir H. Gandomi, and Rodrigo TN Cardoso. "Artificial intelligence applied to stock market trading: a review." IEEE Access 9 (2021): 30898-30917.
[3] Bussmann, Niklas. "Explainable machine learning in credit risk management." Computational Economics 57 (2021): 203-216.
[4] Pustokhina, Irina V. "Artificial intelligence assisted Internet of Things based financial crisis prediction in FinTech environment." Annals of operations research (2021): 1-21.
[5] Cao, Longbing, Qiang Yang, and Philip S. Yu. "Data science and AI in FinTech: An overview." International Journal of Data Science and Analytics 12 (2021): 81-99.

[6] Ciampi, Francesco. "Rethinking SME default prediction: a systematic literature review and future perspectives." Scientometrics 126 (2021): 2141-2188.
[7] Von Zahn, Moritz, Stefan Feuerriegel, and Niklas Kuehl. "The cost of fairness in AI: Evidence from e-commerce." Business & information systems engineering (2021): 1-14.
[8] Drydakis, Nick. "Artificial Intelligence and reduced SMEs' business risks. A dynamic capabilities analysis during the COVID-19 pandemic." Information Systems Frontiers 24.4 (2022): 1223-1247.
[9] Piehlmaier, Dominik M. "Overconfidence and the adoption of robo-advice: why overconfident investors drive the expansion of automated financial advice." Financial Innovation 8.1 (2022): 1-24.
[10] Altrock, Sophie, Anne-Laure Mention, and Tor Helge Aas. "Being human in the digitally enabled workplace: Insights from the robo-advice literature." IEEE Transactions on Engineering Management 71 (2023): 7876-7891.
[11] Breuer, Wolfgang, and Andreas Knetsch. "Recent trends in the digitalization of finance and accounting." Journal of Business Economics 93.9 (2023): 1451-1461.
[12] Von Walter, Benjamin, Dietmar Kremmel, and Bruno Jäger. "The impact of lay beliefs about AI on adoption of algorithmic advice." Marketing Letters 33.1 (2022): 143-155.
[13] Guo, Haochen, and Petr Polak. "Artificial intelligence and financial technology FinTech: How AI is being used under the pandemic in 2020." The Fourth Industrial Revolution: Implementation of Artificial Intelligence for Growing Business Success (2021): 169-186.
[14] Kumari, Bharti, Jaspreet Kaur, and Sanjeev Swami. "System dynamics approach for adoption of artificial intelligence in finance." Advances in Systems Engineering: Select Proceedings of NSC 2019. Springer Singapore, 2021.
[15] Hildebrand, Christian, and Anouk Bergner. "Conversational robo advisors as surrogates of trust: onboarding experience, firm perception, and consumer financial decision making." Journal of the Academy of Marketing Science 49.4 (2021): 659-676.
[16] Ajmani, Prerna, et al. "Impact of AI in Financial Technology-A Comprehensive Study and Analysis." 2023 6th International Conference on Contemporary Computing and Informatics (IC3I). Vol. 6. IEEE, 2023.
[17] Raj, Arman, et al. "Enhancing security feature in financial transactions using multichain based blockchain technology." 2023 4th International Conference on Intelligent Engineering and Management (ICIEM). IEEE, 2023.
[18] Arghydeep, Dataset Gold and Silver Price Prediction with Rolling Regression and LSTM: https://github.com/Arghyadeep/Gold-and-Silver-Price-Prediction-with-Rolling-Regression-and-LSTM/blob/master/Gold%20Futures%20Historical%20Data.csv, Accessed on 25 August 2024.