Asian Journal of Electrical Sciences
ISSN: 2249-6297 (P); 2583-7923 (E)
Vol.13 No.1, 2024, pp.26-34

www.crijournals.org
DOI: https://doi.org/10.70112/ajes-2024.13.1.4235

High-Accuracy Forex Trading Prediction Model Using Machine Learning Algorithms

Ernest E. Onuiri, Tochukwu F. Chikezie, Ifeoma E. Z. Obata and Ruth C. Amanze
Department of Computer Science, School of Computing, Babcock University, Nigeria
E-mail: onuirie@babcock.edu.ng, chikezie1001@student.babcock.edu.ng,
obata0949@student.babcock.edu.ng, amanzer@babcock.edu.ng
(Received 3 March 2024; Revised 22 March 2024; Accepted 11 April 2024; Available online 18 April 2024)

***Abstract* - The foreign exchange (Forex) market, the largest financial market globally, involves the simultaneous buying and selling of currency pairs such as the United States Dollar (USD) and the Japanese Yen (JPY). Due to the high leverage and volatility inherent in Forex trading, retail investors and traders face significant financial risks. To mitigate this, a predictive model was developed using supervised machine learning to improve trading outcomes. A dataset comprising EUR/USD currency pair data from MetaTrader5, covering a period of 7 years (2015–2022), was streamlined to 5 years (2017–2022) for relevance. The dataset contained 13 features and 1,029 records, which underwent data cleaning and processing. Feature selection was performed using the Boruta package in R, and various machine learning algorithms were employed for model training and testing. The model's accuracy was evaluated using the ROC curve and the Kappa statistic. The inclusion of the Simple Moving Average indicator improved forecast accuracy. Among the algorithms tested, the Generalized Linear Model (Logistic Model) performed best, achieving an accuracy of 86.28%, a specificity of 87.42%, a sensitivity of 86.5%, and an AUC (ROC) score of 86.97%. This predictive model has the potential to optimize profits, particularly in the context of binary options trading within the Forex market.**
***Keywords:* Forex Market, Machine Learning, Predictive Model, Feature Selection, Binary Options Trading**

I. INTRODUCTION

The Foreign Exchange (Forex) market is the world's largest financial market, with an average daily transaction volume exceeding 1 trillion United States Dollars, which is 30 times greater than the aggregate volume of all U.S. equity markets [1]. Forex trading involves the simultaneous purchase of one currency and the sale of another. Various currency pairs, such as the Euro/U.S. Dollar (EUR/USD) and the U.S. Dollar/Japanese Yen (USD/JPY), are traded in the Forex market. Companies and governments that buy or sell goods and services in other countries must convert foreign currency revenues into their domestic currency, accounting for about 5% of the daily turnover. The remaining 95% consists of profit-driven trading or speculation [2].

For speculators, individuals interested in making a profit through Forex trading, the best trading opportunities lie with the most traded currencies, commonly referred to as the Majors. The Majors, which include the Swiss Franc, Japanese Yen, Australian Dollar, British Pound, U.S. Dollar, Canadian Dollar, and Euro, now account for over 85% of all daily transactions. Forex trading is a true 24-hour market, beginning each day in New Zealand at 7 a.m. and moving across the globe as the business day starts in each financial center, eventually ending in New York [3].

More so than in any other financial market, currency fluctuations caused by economic, social, and political events can be responded to at any time of day or night. The Forex market is classified as an Over-The-Counter (OTC) or "interbank" market since transactions occur over the phone or through electronic networks between two counterparts. Trading is decentralized, unlike stock and futures markets, which operate on exchanges [4].

However, due to the volatility of the market, retail investors or traders face a high likelihood of incurring losses when speculating in Forex. Research conducted by Compare Forex Brokers [1], which evaluated 35 Forex brokers, found that an average of 71% of retail Forex traders experience losses while trading. Although 29% of traders make profits, 99% of retail Forex traders have failed to generate a profit for four consecutive quarters.

This raises the question: how can trading losses be minimized and profits maximized? The ability to predict market trends with a profitable degree of accuracy provides a significant advantage. This research aims to determine market trends to increase profit margins for retail traders. Accurately predicting these trends is challenging, especially for those new to the Forex market.

The aim of this research was to develop a model that accurately predicts the upward or downward trend of the foreign exchange market, specifically in day trading, to a profitable degree. The specific objectives of this research were to:

1. Conduct exploratory data analysis and apply feature selection techniques to identify features that correlate with the desired trend outcome.
2. Build models using supervised machine learning algorithms on EUR/USD currency pairs with historical data.
3. Evaluate, assess, tune, and compare the accuracy of different models. The best-performing model was selected for deployment.

II. REVIEW OF LITERATURE

Raimundo and Okamoto [5] conducted research to develop a trend indicator for predicting future currency prices. They introduced the SWVP (Similar Wavelet Patterns) indicator, which uses decomposed ripple patterns through Wavelets to make predictions. This indicator was developed by translating and diluting the Mother-Wavelet, formulated over financial time series, to highlight ripples with better correlations. The goal of their study was to create a new trend indicator based on harmonically related sine and cosine patterns from a series of Wavelet families to generate points of interest for buying or selling financial securities as a trading strategy.

In research by Gonzalez *et al.,* [6], a more in-depth examination of the algorithmic treatment of financial time series using the Forex market as an example was recommended. They replaced Recurrent Neural Networks with Temporal Convolutional Neural Networks in their comparison of deep learning models. By expanding the comparison to more established deep learning models and effective machine learning approaches, they supported their findings. The effectiveness of their algorithms was tested using a proposed financial framework.

Aggarwal and Sahani [7] compared Long Short-Term Memory (LSTM), simple Recurrent Neural Networks (SRNN), and Gated Recurrent Unit (GRU) models to estimate currency exchange rates for 22 currencies against the United States Dollar (USD) simultaneously. Additionally, the models forecasted foreign currency exchange rates for 30 consecutive days using data from the previous 365 days. The models used the same number of neural network layers, input, desired output, optimizer, and learning rate.

In their research, Deng and Sakurai [8] simulated foreign exchange rate trading using Ichimoku Kinko Hyo, a traditional Japanese technical indicator widely used for technical analysis in Japan. They developed and tested two trading strategies based on the support/resistance levels of the five Ichimoku components on short-term foreign currency rates. These strategies were applied to the following currency pairs: USD/JPY, EUR/USD, GBP/USD, USD/CHF, and AUD/USD. The results showed that one Ichimoku-based trading strategy generated a significantly higher average return than other baseline strategies.

Zhao [9] used a vector autoregressive model and the Granger causality test to examine the relationship between foreign direct investment (FDI), trade exports, and exchange rate volatility in China. His empirical study demonstrated that these three variables have a complementary causal connection. Zhao's research suggested that the depreciation of the RMB/USD exchange rate increased export trade, satisfying the Marshall-Lerner criterion. FDI improved the trade balance but had little effect on trade exports. Policy recommendations included enhancing the RMB exchange rate formation mechanism, bolstering independent intellectual property rights research and development, and improving the quality of foreign investment and value-added products.

III. METHODOLOGY

In building the machine learning model, the following steps were followed:

Step 1: Data gathering – The historical EUR/USD data was obtained from MetaTrader5.
Step 2: Data preparation/cleaning
Step 2.5: Data exploration
Step 3: Feature selection
Step 4: Model building and evaluation – The dataset was split into 70% for training and 30% for testing. Cross-validation was performed on the training data.
Step 5: Model selection and deployment

The process of cleaning and transforming raw data prior to processing and analysis is known as data preparation. It is a crucial step that often involves reformatting data, making modifications, and integrating datasets to enrich the data. Although data preparation can be time-consuming, it is essential for placing data in context, gaining insights, and reducing bias caused by poor data quality. Common steps in the data preparation process include standardizing data formats, improving source data, and reducing outliers.

A. Data Preparation Steps

The following steps were taken to ensure the data is reliable, usable, and stable:

1. Data Gathering: The data preparation process began with finding the appropriate data, either from an existing data catalog or added ad-hoc. For this research, the dataset was sourced primarily from MetaTrader 5, a multi-asset platform that supports trading in Forex, stocks, and futures. It also offers superior tools for comprehensive price analysis, algorithmic trading applications (trading robots, Expert Advisors), and copy trading.

2. Data Preparation/Cleaning: Data cleaning was the most time-consuming step of the data preparation process but was essential for removing erroneous data and filling gaps. Key tasks included:

a. Eliminating unnecessary data and outliers.
b. Completing missing values.
c. Constraining data to a predefined pattern.
d. Abstracting entries containing data not meant for public use.

After cleaning, the data was reviewed for errors introduced during the preparation process. Often, system errors manifest at this stage and must be corrected before moving forward. However, in this research, the fourth task was unnecessary. The MICE package, which uses predictive Mean Matching

(PMM) for data imputation, was employed to fill missing values. Where this method failed, complete case analysis was used.

3. Transform and Enrich Data: Data transformation involved updating formats and value entries to achieve specific results or make the data more accessible to a broader audience. Data enrichment incorporated and linked previously collected data with new sources to provide new perspectives.

4. Data Exploration: Exploring each dataset after collection was crucial. Exploratory data analysis (EDA) was used to identify patterns, detect anomalies, test hypotheses, and verify assumptions through summary statistics and graphical representations. The goal of EDA is to make sense of the data. For this research, R libraries such as moments, ggplot2, and psych were utilized. EDA concerns included forming hypotheses, ensuring statistical inferences were based on correct assumptions, adopting suitable statistical tools, and initiating surveys or experiments to obtain further data.

5. Store Data: Once prepared, the data could be stored or directed to a third-party application, such as a business intelligence tool, for processing and analysis. In this research, the R package Shiny was used for this purpose.

B. Data Sets

The dataset used consisted of multivariate variables to find correlations and potential causations that could influence the model. Multivariate datasets contain multiple variables, typically defined as those with three or more data types (variables). Correlation datasets represent a collection of values that show relationships between variables. In this case, the values were found to be interdependent. Correlation refers to a statistical relationship between two entities or variables.

C. Dataset Features

The dataset contains a total of 13 features. Original features, such as Open, High, Low, Close, Tick Volume, Volume, Spread, and Date, were sourced from MetaTrader 5. The Bull_Bear feature is an additional variable indicating whether the closing price was higher than the opening price. The remaining features, prefixed with SMA, result from applying the Simple Moving Average (SMA) indicator to the Opening, High, Low, and Close prices for the EUR/USD currency pair using the quantmod library.

Features of the dataset include High, Low, Open, Close, Tick Volume, Volume, SMA_Close, SMA_Open, SMA_High, SMA_Low, Spread, Date, and Bull_Bear. The following are explanations of the features:

1. DATE: The designated period for each price action. The research uses a daily period.
2. OPEN: The price at which the currency pair begins its first trade at the start of the trading day.
3. CLOSE: The price at which the currency pair concludes its last trade at the end of the trading day.
4. HIGH: The highest price at which the currency pair trades within a period.
5. LOW: The lowest price at which the currency pair trades within a period.
6. TICK_VOLUME: A measure of the smallest upward or downward fluctuation in the price of the currency pair.
7. VOLUME: The amount of EUR/USD traded over a period.
8. SPREAD: The difference between a forex broker's sell rate and buy rate when exchanging or trading currencies.
9. SMA_OPEN: The Simple Moving Average indicator applied to the Open price.
10. SMA_HIGH: The Simple Moving Average indicator applied to the High price.
11. SMA_LOW: The Simple Moving Average indicator applied to the Low price.
12. SMA_CLOSE: The Simple Moving Average indicator applied to the Close price.
13. BULL_BEAR: A binary value indicating whether the Close price is higher than the Open price. A value of 1 is assigned if it is, and 0 if otherwise.

Simple Moving Average (SMA): An arithmetic moving average calculated by adding recent prices and dividing that sum by the number of periods used in the calculation.

D. Dataset Sample Size

The sample size is determined by the number of years used to train the data and the features. The original dataset comprised historical EUR/USD data from 2018 to 2022. Prior to feature selection, the dataset included a total of 13 features and 1,038 records, with each row representing daily trend rates over 5 years. After feature selection, the dataset was reduced to 6 features and 1,029 records.

E. Data Cleaning

This is a crucial step in any machine learning research. Before selecting or applying any models, performing data cleaning operations is essential for obtaining unbiased results. In this case, the dataset was already clean; therefore, this process was omitted.

F. Machine Learning Stage and Methodology

This stage involves applying algorithms for testing and training the data. It is where the model is built, trained, and evaluated. As previously mentioned, the dataset was split into training and testing subsets. Cross-validation was then performed on the training set. This stage was algorithm-driven.

Training and testing are methods used to measure the accuracy of the model. The process is termed Train/Test because the dataset is divided into two subsets: training and testing sets - 70% for training and 30% for testing.

The training set is used to train the model. The model is initially created and then tested with the testing set. When an acceptable model is achieved, at least in terms of training data, it is subsequently tested with the testing data to verify if it produces consistent results. Testing the model serves as an accuracy check. Once the model is validated as accurate, it can be used for active predictions.

IV. IMPLEMENTATION

The implementation of the machine learning model to predict the EUR/USD currency trend is documented. This includes an analysis of the features and a detailed explanation of how the model is built. Additionally, screenshots of the code and the dashboard of the deployed model are provided.

A. Pre-Processing

This includes setting the working directory, loading the data, inserting Simple Moving Average indicator values into the dataset, and performing data cleaning.

A visual representation of the features in the dataset, along with the percentage of missing values, shows that missing values occurred only in the Simple Moving Average feature and to a negligible degree.

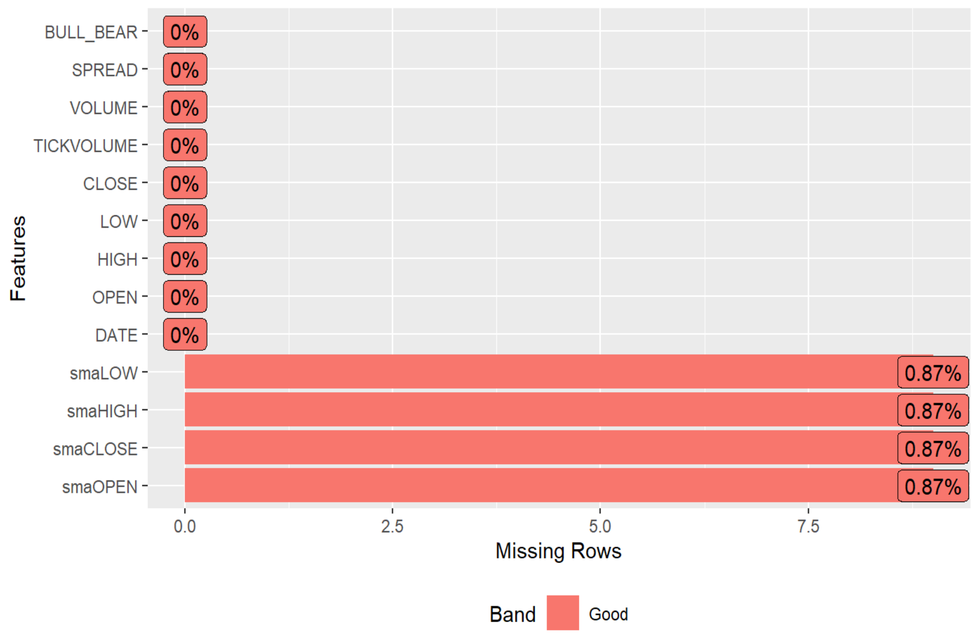

Fig. 1 Missing values before Complete Case Analysis

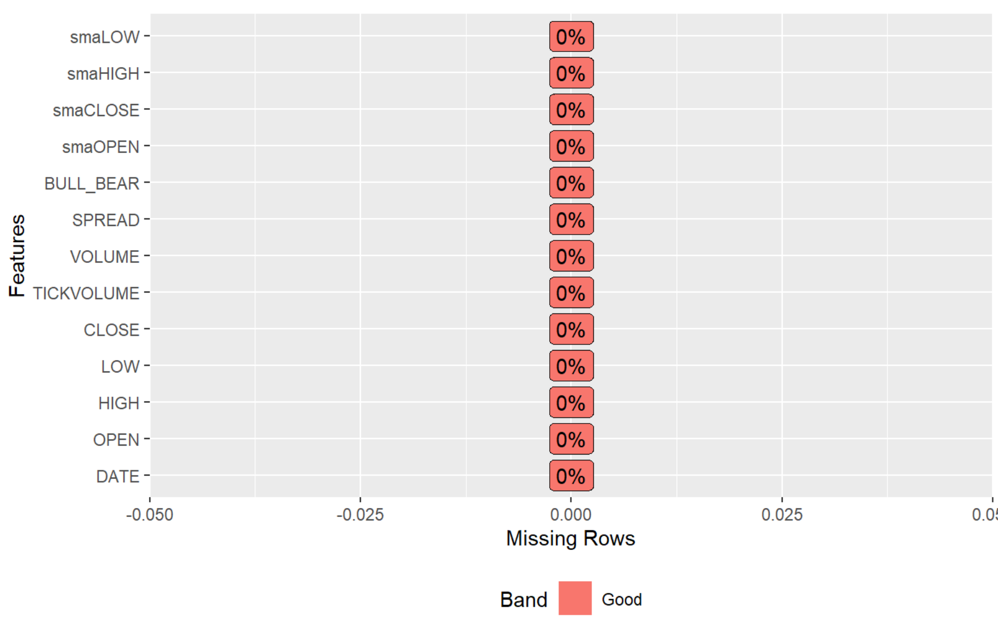

Fig. 2 Dataset after complete case analysis

A visual representation of the features in the dataset and the percentage of missing values was created after performing complete case analysis on the dataset. The dataset now has no missing values. After cleaning the data, the next phase involved conducting a univariate analysis of the dataset features using the R library "moments" to determine the kurtosis and standard deviation of the features, as shown in Figs. 3 through 5.

This analysis is crucial for evaluating price movement. The tail of a distribution is characterized by the kurtosis measure, which addresses the question: "How close are the distribution's outlying values to those of a standard normal distribution?" The standard normal distribution has a kurtosis of 0.

A negative kurtosis score indicates a thin-tailed distribution, meaning the sample values are closer to the median than in a standard normal distribution. Conversely, a positive kurtosis value signifies a fat-tailed distribution, where extreme outcomes are more frequent than would be expected from a standard normal distribution.

Fat-tailed distributions are particularly significant as they suggest the market can produce many extreme values that deviate from the standard normal distribution. The analysis reveals that price movements are largely fat-tailed, with the kurtosis test indicating a strong presence of outliers.

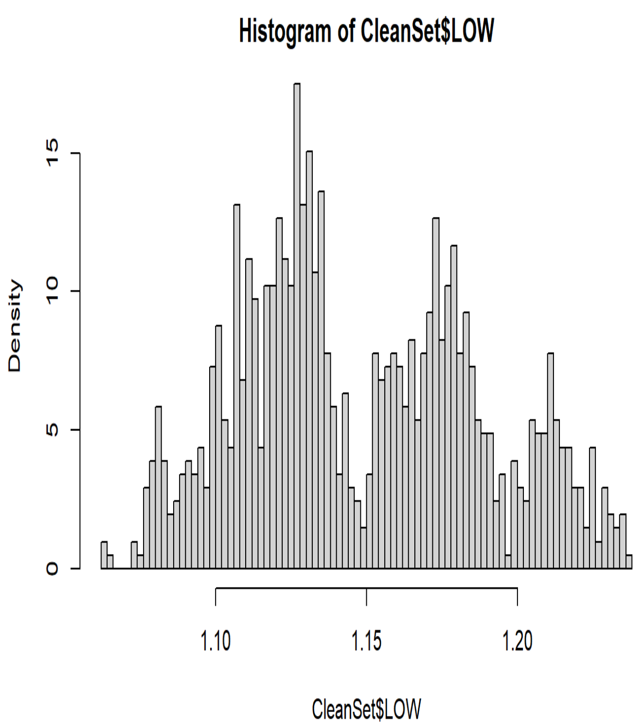

Fig. 3 Univariate analysis of the feature LOW shows the histogram of the feature low from the final dataset called CleanSet

This represents a fat-tailed distribution, indicating that the feature generates a significant number of outliers.

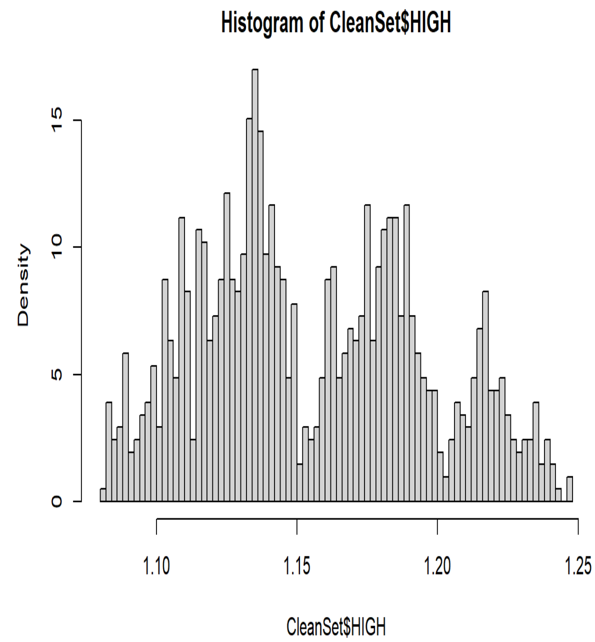

Fig. 4 Univariate analysis of the feature HIGH shows the histogram of the feature high from the final dataset called CleanSet

This is a fat-tailed distribution, which indicates that the feature generates a significant number of outliers.

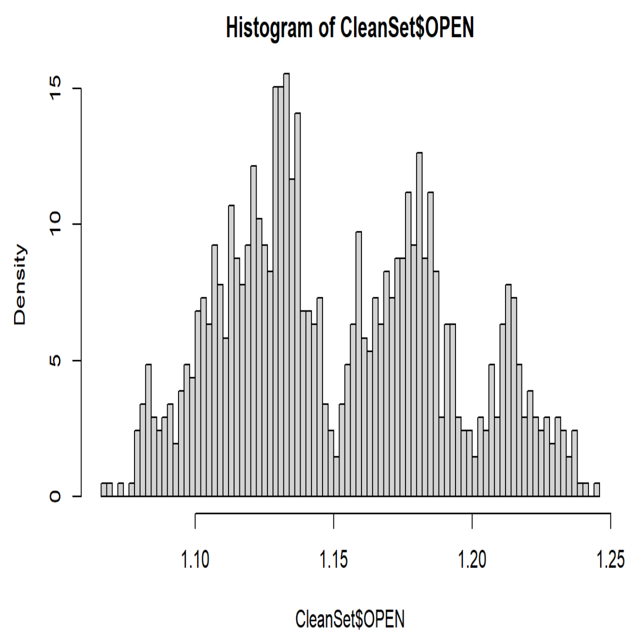

Fig. 5 Univariate analysis of the feature OPEN shows the histogram of the feature open from the final dataset called CleanSet

This is a fat-tailed distribution, which indicates that the feature generates a significant number of outliers.

After performing univariate analysis of the dataset features to determine the kurtosis and standard deviation, bivariate analysis was conducted using the R library "ggplot2," as shown in Figs. 6 through 8.

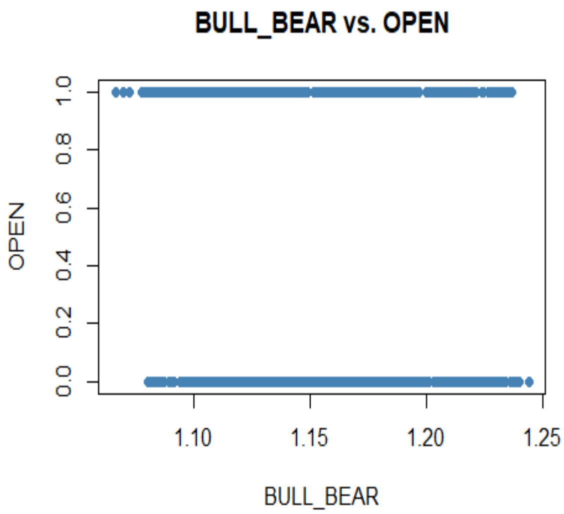

Fig. 6 Scatterplot of dependent variable and OPEN shows the feature open plotted against the dependent variable Bull_Bear

The feature "Open" shows a positive correlation with the dependent variable.

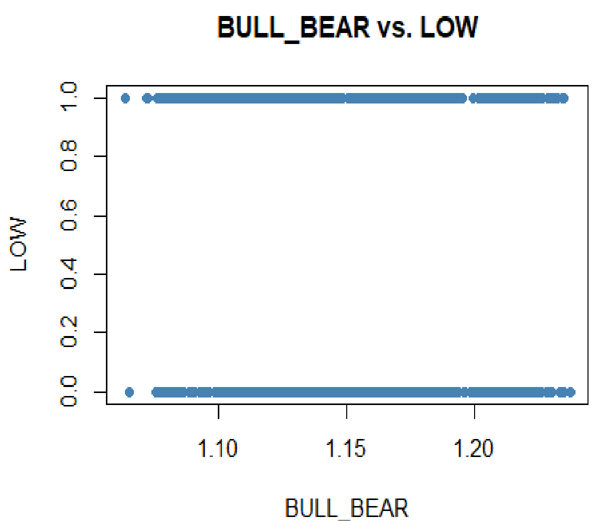

Fig. 7 The scatterplot of the dependent variable and "Low" shows the feature "Low" plotted against the dependent variable "Bull_Bear"

The feature "Low" shows a positive correlation with the dependent variable.

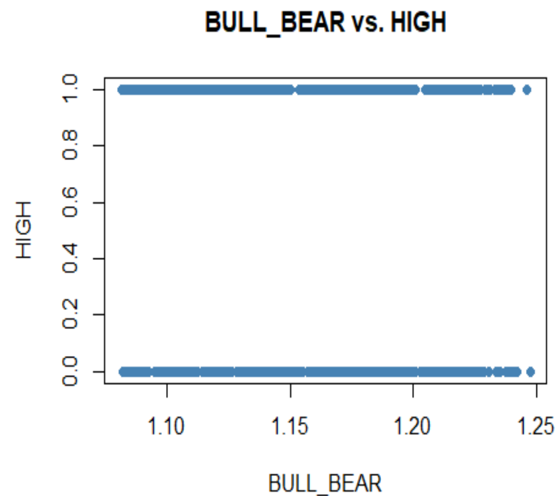

Fig. 8 The scatterplot of the dependent variable and "High" shows the feature "High" plotted against the dependent variable "Bull_Bear"

The feature "High" shows a positive correlation with the dependent variable. The inference drawn from the bivariate analysis is that the dependent variable shows a positive correlation with the independent variables.

Multivariate analysis was performed using the "psych" package. The correlations between the features, except for "Volume," which had an NA correlation test value, are highlighted in Table I (A) below.

TABLE I (A) CORRELATION MATRIX OF THE FEATURES

	OPEN	**HIGH**	**LOW**	**TICK VOLUME**
OPEN	1	1	1	0.11
HIGH	1	1	1	0.14
LOW	1	1	1	0.08
TICK VOLUME	0.11	0.14	0.08	1
SPREAD	-0.22	-0.22	-0.22	0.02
BULL_BEAR	-0.05	-0.01	0	-0.03
smaOPEN	0.98	0.98	0.98	0.14
smaHIGH	0.98	0.98	0.98	0.16
smaLOW	0.98	0.98	0.98	0.12

TABLE I (B) CORRELATION MATRIX OF THE FEATURES

	SPREAD	**BULL_BEAR**	**smaOPEN**	**smaHIGH**	**smaLOW**
OPEN	-0.22	-0.05	0.98	0.98	0.98
HIGH	-0.22	-0.01	0.98	0.98	0.98
LOW	-0.22	0	0.98	0.98	0.98
TICKVOLUME	0.02	-0.03	0.14	0.16	0.12
SPREAD	1	-0.01	-0.22	-0.22	-0.21
BULL_BEAR	-0.01	1	-0.04	-0.04	-0.04
smaOPEN	-0.22	-0.04	1	1	1
smaHIGH	-0.22	-0.04	1	1	1
smaLOW	-0.21	-0.04	1	1	1

Table I(A) presents a correlation matrix of the features in the dataset. All features showed low correlation with the dependent variable “Bull_Bear,” except for “Low.”

B. Implementing Feature Selection

Using the Boruta package, feature selection was carried out. The feature selection process in the Boruta package works as follows: A randomization technique generates jumbled duplicates of the features in the dataset (referred to as shadow features). The main dataset is then used to train a random forest classifier, and a feature importance measure (Mean Decrease Accuracy) is used to assess the value of each feature. Generally, features with higher mean values are considered more significant. Features determined to be of little or no value, compared to the highest possible Z-scores for their shadow features, are eliminated during each iteration [10].

The process concludes when all features are either confirmed or rejected, or when the Boruta algorithm exceeds the set limit of random forest iterations. Five features were selected by Boruta: smaLOW, smaOPEN, HIGH, LOW, and OPEN.

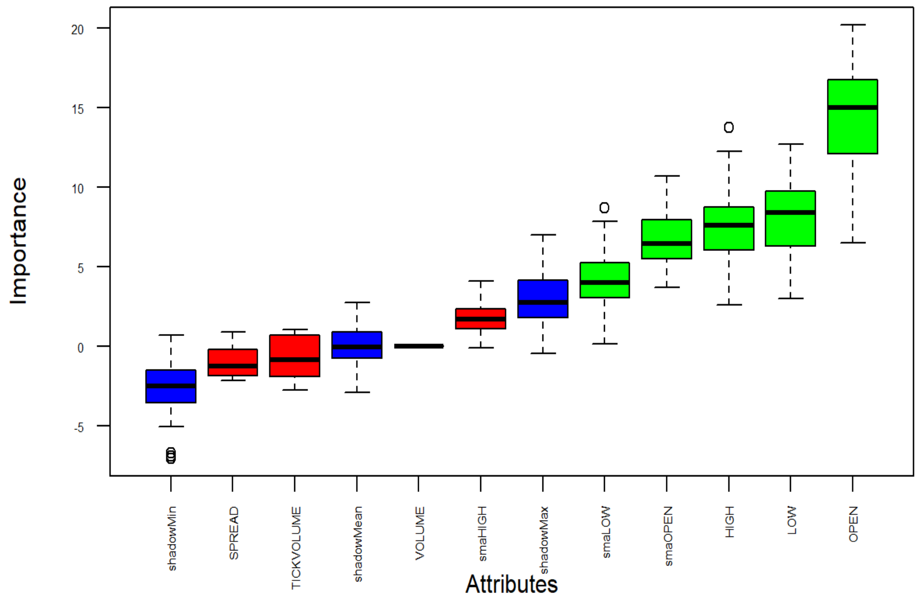

Fig. 9 Boruta selected features shows the selected features by boruta

Feature selection was carried out using the Boruta package, and the selected features are indicated by the color green.

C. Model Building

The dataset was split into training and testing sets, and the “caret” library was used for machine learning. The dataset was fed into eight machine learning algorithms, selected in the following order:

1. *Linear Algorithms:* Generalized Linear Regression (GLM) and Linear Discriminant Analysis (LDA)
2. *Non-Linear Algorithms:* K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Neural Network (pcaNNet), and Multi-Layer Perceptron (MLP)
3. *Ensemble Methods:* Random Forest (RF) and Naive Bayes

Analysis was performed using 10-fold cross-validation with three repeats.

D. Model Evaluation

The model was evaluated using accuracy and Kappa statistics.

TABLE I (C) ACCURACY TABLE FOR MODEL PERFORMANCE

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
LG	0.760563	0.824772	0.87412	0.862835	0.890411	0.945206
LDA	0.780822	0.824772	0.862062	0.8618833	0.89003	0.945206
KNN	0.527778	0.57734	0.622457	0.6235531	0.652778	0.767123
RF	0.569444	0.684932	0.710331	0.7127464	0.744791	0.849315
SVM	0.777778	0.809028	0.846146	0.8416074	0.87456	0.90411
NB	0.416667	0.479452	0.520548	0.5157415	0.547389	0.630137
MLP	0.486111	0.493151	0.506849	0.5037039	0.507042	0.555556
NN	0.458333	0.520548	0.548621	0.5456702	0.569444	0.638889

Table I(C) shows the performance of the machine learning algorithms using the accuracy metric. The Generalized Linear Model (GLM) performed best with a mean value of 0.8628, while the Multiple Layer Perceptron (MLP) performed least with a mean value of 0.5037.

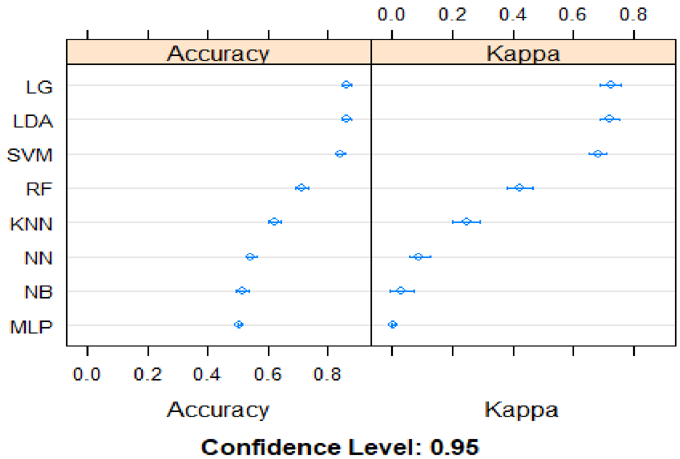

Fig. 10 A rendering of the machine learning performance

This displays the performance of the machine learning algorithms using two performance metrics: Accuracy and Kappa statistics. From the figure, it is observed that the Generalized Linear Model (GLM), Linear Discriminant Analysis (LDA), and Support Vector Machine (SVM) are the leading algorithms, with mean accuracies of 0.8628, 0.8619, and 0.8416, respectively. Hyperparameter tuning of the top two performing models, Generalized Linear Model (GLM) and Linear Discriminant Analysis (LDA), did not improve the model. Consequently, the Generalized Linear Model (GLM) was selected as the top-performing model.

E. Statistics for Best Performing Model

Exporting the statistics report from R, the performance of the Generalized Linear Model is detailed below:

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 135  19
         1  21 132

               Accuracy : 0.8697
                 95% CI : (0.8268, 0.9053)
    No Information Rate : 0.5081
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.7394

 Mcnemar's Test P-Value : 0.8744

            Sensitivity : 0.8654
            Specificity : 0.8742
         Pos Pred Value : 0.8766
         Neg Pred Value : 0.8627
             Prevalence : 0.5081
         Detection Rate : 0.4397
   Detection Prevalence : 0.5016
      Balanced Accuracy : 0.8698

       'Positive' Class : 0
```

Fig. 11 An Export of the Confusion Matrix and Statistics for Generalized Linear Model

This shows the confusion matrix of the Generalized Linear Model, the best-performing model, as well as other evaluation metrics.

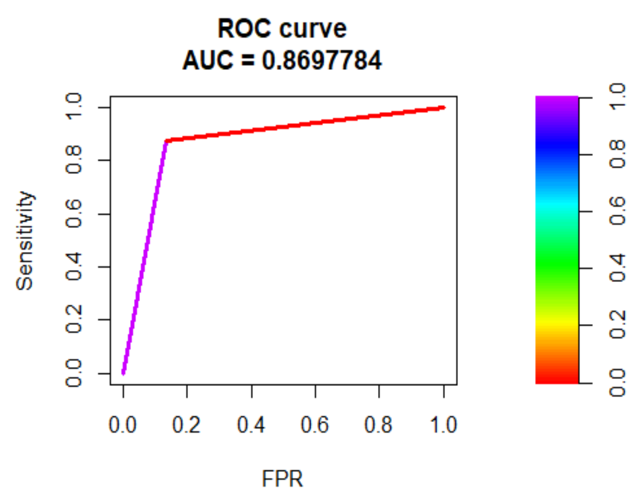

Fig. 12 Performance Result of the Model using the ROC curve

This shows the accuracy curve of the Generalized Linear Model (LG) algorithm, which exhibits near-perfect agreement.

V. CONCLUSION

Preventing loss of funds with 100% accuracy in the forex market is not feasible; however, mitigating the loss is possible. With this rationale, the idea to predict the trend for the day was conceived. Predicting the trend at the start of the day provides a valuable edge to novice traders. The Linear Regression model can help forecast forex trends, such as bull (up) or bear (down) trends. The problem was addressed as a classification issue, and eight machine learning algorithms were employed to handle this classification problem. Among them, the Generalized Linear Model (logistic regression) performed best, achieving an accuracy of 86.97%. Major milestones in the project included acquiring clean data, discovering the quantmod package for inputting the Simple Moving Average (SMA) indicator, which significantly improved accuracy, performing univariate, bivariate, and multivariate analysis on the dataset, building the model, and conducting predictions. The developed model was hosted on Shiny, an R programming language package, allowing users to predict trends by inputting features such as ‘Open’, ‘High’, ‘Low’, and ‘smaOpen’, and ‘smaLow’. It is crucial to use a sizeable range when dealing with time-series data for prediction, as it enables the model to handle diverse probabilities and outcomes. Additionally, it is important not to restrict the dataset to only one algorithm. In developing the predictive model, algorithms from three categories-linear, nonlinear, and ensemble-were selected to ensure the model achieved an obtainable degree of accuracy. The results of various supervised machine learning algorithms are presented in Table I (C), where the Generalized Linear Model performed best. Incorporating the Simple Moving Average (SMA) indicator significantly improved model accuracy. The model can predict the trend of the EUR/USD currency pair at the start of the day with an accuracy of 86.97%. The project can be integrated into an Expert Advisor on MetaTrader 5 or any broker that supports algorithmic trading for profitable daily trades. Building the model with real-time data would enhance accuracy, although such a model would require automation. Additionally, the model was limited to the EUR/USD currency pair; other currency pairs could be incorporated in future work.

References

[1] Compare Forex Brokers, “2021 Forex Trading Statistics + Industry Guide [Fact Checked],” June 9, 2021. [Online]. Available: https://www.compareforexbrokers.com/forex-trading/statistics. [Accessed: Sep. 13, 2024].

[2] J. Chen, “Currency Pair Definition,” Investopedia, March 10, 2022. [Online]. Available: https://www.investopedia.com/terms/c/currency pair.asp. [Accessed: Sep. 13, 2024].

[3] “Lesson 14: Forex Trading Sessions,” Forexcec. [Online]. Available: https://www.forexcec.com/gx/en/education/lesson-14-forex-trading-sessions. [Accessed: Sep. 13, 2024].

[4] A. Ganti, “Foreign Exchange Market Definition,” Investopedia, November 8, 2020. [Online]. Available: https://www.investopedia.com/terms/forex/f/foreign-exchange-markets.asp. [Accessed: Sep. 13, 2024].

[5] M. S. Raimundo and J. Okamoto, “SVR-wavelet adaptive model for forecasting financial time series,” in *Proc. 2018 Int. Conf. on Information and Computer Technologies (ICICT)*, 2018. [Online]. Available: https://doi.org/10.1109/infoct.2018.8356851. [Accessed: Sep. 13, 2024].

[6] S. H. González, P. I. López, G. P. Bringas, H. Quintián, and E. Corchado, *Hybrid Artificial Intelligent Systems: 16th International Conference, HAIS 2021, Bilbao, Spain, September 22–24, 2021, Proceedings* (Lecture Notes in Computer Science, vol. 12886, 1st ed. 2021). Springer, 2021.

[7] P. Aggarwal and A. K. Sahani, “Comparison of Neural Networks for Foreign Exchange Rate Prediction,” *IEEE*, Nov. 26, 2020. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9342733. [Accessed: Sep. 13, 2024].

[8] S. Deng and A. Sakurai, “Short-term foreign exchange rate trading based on the support/resistance level of Ichimoku Kinko Hyo,” *IEEE*, April 26, 2014. [Online]. Available: https://ieeexplore.ieee.org/document/6948127. [Accessed: Sep. 13, 2024].

[9] C. Zhao and J. Du, “Causality Between FDI and Economic Growth in China,” *The Chinese Economy*, vol. 40, no. 6, pp. 68–82, 2007. [Online]. Available: https://doi.org/10.2753/ces1097-1475400604. [Accessed: Sep. 13, 2024].

[10] E. E. Onuiri, “Clinical Associations and Genetic Alterations to Predict Radiotherapy Treatment Response in Patients with Triple Negative Breast Cancer (TNBC),” Thesis, Rutgers University, August 2020. [Online]. Available: https://rucore.libraries.rutgers.edu/rutgers-lib/64650/PDF/1/play/. [Accessed: Sep. 13, 2024].