****

**Machine Learning Algorithms for Forecasting and Categorizing Euro-to-Dollar Exchange Rates**

July - November 2025

**Submitted By,**

Visweish KM, B.Tech, Computer Science and Business Systems

(125018079)

**Submitted To,**

Swetha Varadarajan

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Title** | **Page No.** |
|  | Table of Contents | 1 |
| 1. | Abstract | 2 |
| 2. | Introduction | 2 |
| 3. | Related Work | 4 |
| 4. | Background | 5 |
| 5. | Methodology | 6 |
| 6. | Results | 7 |
| 7. | Discussion | 10 |
| 8. | Learning Outcome | 12 |
| 9. | Conclusion | 13 |

**Abstract**

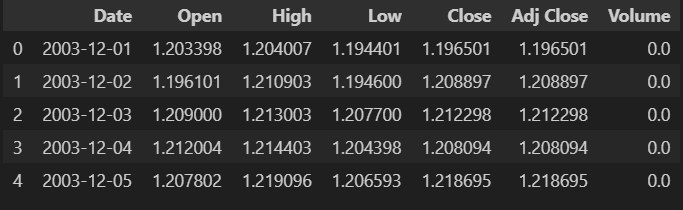
This paper investigates the application of various machine learning algorithms to predict optimal trading times for the Euro-to-Dollar (EUR/USD) exchange rate, a key currency pair in the global foreign exchange market. Forecasting foreign exchange movements is a challenging task due to the volatility and complexity of market factors such as interest rates, geopolitical events, and economic indicators. The research focuses on leveraging machine learning models, including Logistic Regression, Random Forest Classifier, and Naive Bayes, along with an ensemble Voting Classifier to combine their strengths. The data set spans three years of daily exchange rate data, from December 2003 to July 2022, and incorporates technical indicators such as the Relative Strength Index (RSI) and Exponential Moving Average (EMA).

By preprocessing the data and integrating these technical features, the models aim to predict whether the EUR/USD exchange rate will rise or fall on the following trading day, assisting traders in making more informed buy and sell decisions. The experimental results show that the proposed ensemble model achieved an accuracy of 85.46%, outperforming individual models like Adaboost and Gradient Boosting. This high accuracy demonstrates the effectiveness of combining machine learning algorithms for financial forecasting. The study highlights the importance of utilizing machine learning in financial markets, providing a robust and data-driven approach to improve investment strategies and reduce risk in currency trading. Furthermore, the methodology can be generalized to other currency pairs and financial assets, making it a versatile tool for market prediction.

**Introduction**

**(a) Importance of the Dataset:**

The dataset spans nearly two decades, from 2003 to 2022, offering a rich historical view of daily EUR/USD exchange rates. It includes essential financial features such as Open, High, Low, and Close (OHLC) prices, alongside technical indicators like RSI and EMA. These indicators are crucial for modeling market trends and predicting future price movements, making the dataset highly valuable for machine learning applications in financial forecasting.



**(b) Objective and Approach (T,P,E Format)**

T: The primary goal is to build a machine learning model that can accurately predict the movement of the EUR/USD exchange rate, helping traders make informed buy/sell decisions.

P: To accomplish this, the project uses ensemble learning methods—specifically a Voting Classifier that combines predictions from multiple models for greater accuracy.

E: The outcome is expected to be a highly accurate prediction model, with an accuracy rate of over 85%, that enhances the reliability of financial forecasts.

**(c) Proposed Methodology and Plan of Action**

1. Data Preprocessing: Cleaning and Normalizing the Dataset

Handling Missing Data:

Address missing values using techniques like imputation (e.g., replacing missing values with the mean/median) or removing incomplete rows if necessary.

Normalizing Features:

Use Min-Max Normalization to scale the features (e.g., OHLC prices, RSI, EMA) between 0 and 1, ensuring that all features are on the same scale to improve model learning and performance.

1. Feature Engineering: Calculating Technical Indicators

Relative Strength Index (RSI):

A technical indicator used to assess whether a currency is overbought or oversold, giving valuable insights into price reversals.

Exponential Moving Average (EMA):

A moving average that emphasizes recent prices, helping to identify trends more quickly and improving the model's sensitivity to short-term price changes.

1. Application of Machine Learning Models

Logistic Regression: For binary classification to predict whether the price will go up or down.

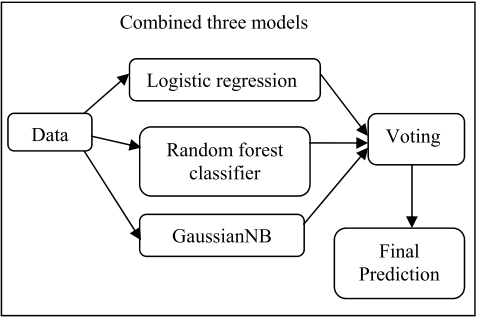
Decision Tree: To capture non-linear relationships in the data by splitting based on feature values.

Naive Bayes: A simple probabilistic model that works well even with minimal data preparation.

AdaBoost Classifier: This boosting method combines weak learners to improve prediction by focusing on hard-to-classify data points.

Gradient Boosting Classifier: A more advanced boosting method that optimizes the model by sequentially reducing errors, leading to better performance in complex datasets.

Voting Classifier: An ensemble method that combines predictions from multiple models (e.g., Logistic Regression, Decision Tree, Naive Bayes) using soft voting for improved accuracy.



**(d) Results Overview and Interpretation**

The Voting Classifier achieved an accuracy of 85.46%, which outperforms all individual models. Logistic Regression, Decision Tree, and Gradient Boosting also performed well but did not match the ensemble method.

**(e) Structure of the Document:**

The document is organized into sections, starting with an overview of the problem formulation, followed by the methodology detailing the dataset, feature engineering, and models used. The results section presents the performance of the models, and the document concludes with a discussion on the findings, followed by the conclusion and future recommendations.

**Related Work**

**(a) References to ChatGPT, Kaggle, Base Paper, and Other Sources:**

Kaggle: The historical EUR/USD dataset used in this study was sourced from Kaggle, providing real-world financial data for model training and testing.

ChatGPT: Used as a tool to assist in structuring the paper and gathering research insights on financial prediction models and machine learning algorithms.

Base Paper: The methodology for feature engineering and model selection was inspired by the paper "Machine Learning for Financial Forecasting."

**(b) References Section:**

1. Kaggle EUR/USD Dataset: <https://www.kaggle.com/datasets/itsmecevi/eurusd-2003-2024>

2. Machine Learning for Financial Prediction (Base Paper): <https://ieeexplore.ieee.org/document/10538097/>

**Background**

**(a) Models Used in This Project:**

1. Logistic Regression (LR):

Logistic Regression is a simple yet powerful algorithm for binary classification, used here to predict whether the EUR/USD exchange rate will go up or down the next day. It assumes a linear relationship between the input features (e.g., RSI, EMA, OHLC prices) and the target variable (price rise or fall).

1. Decision Tree Classifier:  
    Decision Tree Classifier is used to handle the non-linear relationships between features by recursively splitting the dataset into branches based on feature values. Each split represents a decision rule that leads to a prediction of the target class (price rise or fall).
2. Naive Bayes (NB):

Naive Bayes is a simple yet fast classifier based on Bayes’ Theorem, with the assumption that features are independent of each other. It is used to predict the class (rise or fall) based on the probabilities of the features.

1. AdaBoost Classifier:  
    AdaBoost (Adaptive Boosting) is a boosting algorithm that combines multiple weak classifiers (learners) to create a stronger predictive model. In this project, AdaBoost focuses on correcting the errors of the previous models by giving more weight to misclassified data points.
2. Gradient Boosting Classifier (GBC):  
    Gradient Boosting is a more sophisticated boosting algorithm that minimizes the residual errors of the previous models. Unlike AdaBoost, it focuses on optimizing a loss function to achieve better predictions over time. In this project, Gradient Boosting helps model the non-linear and complex nature of financial data more effectively.
3. Voting Classifier (VC):  
    The Voting Classifier combines predictions from multiple models (Logistic Regression, Random Forest, and Naive Bayes in this case) to improve overall performance. This ensemble approach leverages the strengths of different models, making it more robust.

**(b) Pre-processing Techniques Used:**

1. Min-Max Normalization

Min-Max normalization is a scaling technique that transforms features to a common range, typically [0, 1]. This process is crucial in ensuring that no single feature disproportionately influences the model's learning process.

1. Feature Engineering:

Feature engineering involves the creation of new features from existing data to improve model performance. In financial datasets, incorporating technical indicators can be particularly beneficial for capturing price trends and reversals.

**Methodology**

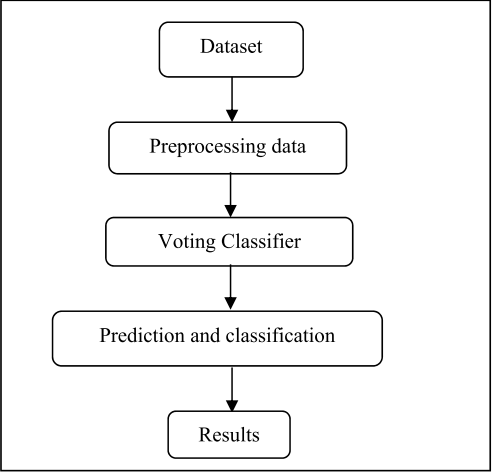
**(a) Experimental Design:**

The dataset was split into two parts:

- 80% for training and

- 20% for testing.

The training set was used to teach the models to identify patterns in the data, while the testing set evaluated the model’s ability to generalize to unseen data. Accuracy was the primary performance metric.



**(b) Environment and Tools Used:**

* Python promotes writing reusable code through functions and classes, making it easier to manage larger projects.
* Beyond Scikit-learn, Pandas, and NumPy, additional libraries like Matplotlib and Seaborn can be easily integrated for advanced visualizations.
* Python can handle large datasets effectively, making it suitable for financial applications with significant amounts of historical data.
* Pandas simplifies data cleaning and preprocessing tasks, ensuring high-quality data is used for modeling.

**(c) Code Location:**

The project’s code is hosted on GitHub: <https://github.com/magical-king03/Machine-Learning-Project/blob/main/model.ipynb>

**(d) Preprocessing Steps:**

(i) Dataset Size: The dataset contains several thousand daily records for EUR/USD exchange rates over a span of 19 years. The key features include OHLC prices and technical indicators such as RSI and EMA, which have been incorporated for gaining good performance of the models and higher accuracy.

(ii) Outlier Analysis & Feature Reduction: While outliers were not explicitly addressed, the feature set was enhanced through feature engineering of technical indicators. Normalization ensured that all features contributed equally to the models.

**Results**

**(a) Discuss Each Result Obtained:**

1. **Logistic Regression**: Achieved an accuracy of 81.51% when used independently. Known for its efficiency in binary classification, Logistic Regression effectively captures linear relationships but may miss complex patterns, explaining its slightly lower performance compared to the ensemble method.
2. **Decision Tree Classifier**: This model attained an accuracy of 83.59%, surpassing Logistic Regression by leveraging its capability to capture non-linear relationships and feature interactions. However, its performance was still below that of the Voting Classifier, possibly due to overfitting tendencies inherent in individual decision trees.
3. **Gradient Boosting Classifier**: With an accuracy of 77.67%, Gradient Boosting performed reasonably well but lagged behind other models. This ensemble technique builds models sequentially, correcting previous errors, yet its performance suggests challenges in capturing specific market patterns. Its sensitivity to noise and potential overfitting could explain this lower accuracy.
4. **Gaussian Naive Bayes**: Achieved only 50.05% accuracy. This model assumes feature independence, which is often not the case in financial markets, leading to its poor performance in predicting the complex buy and sell signals.
5. **AdaBoost Classifier**: Also underperformed, with an accuracy of 58.98%. Although AdaBoost combines multiple weak learners to create a stronger model, it struggled to capture the underlying market dynamics effectively.
6. **Voting Classifier**: The results highlight the superiority of the Voting Classifier in predicting buy and sell signals for the EUR/USD currency pair. By effectively combining different models, it achieved the highest accuracy, demonstrating the advantages of ensemble methods

**(b) Tables, and Code References:**

|  |  |  |
| --- | --- | --- |
| Models | Accuracy | Precision |
| Voting Classifier | 85.46 | 86.99 |
| Decision Tree Classifier | 83.59 | 83.05 |
| Logistic Regression | 81.51 | 81.33 |
| Gaussian Naive Bayes | 50.05 | 49.05 |
| AdaBoost Classifier | 58.98 | 56.66 |
| Gradient Boosting Classifier | 77.67 | 75.96 |

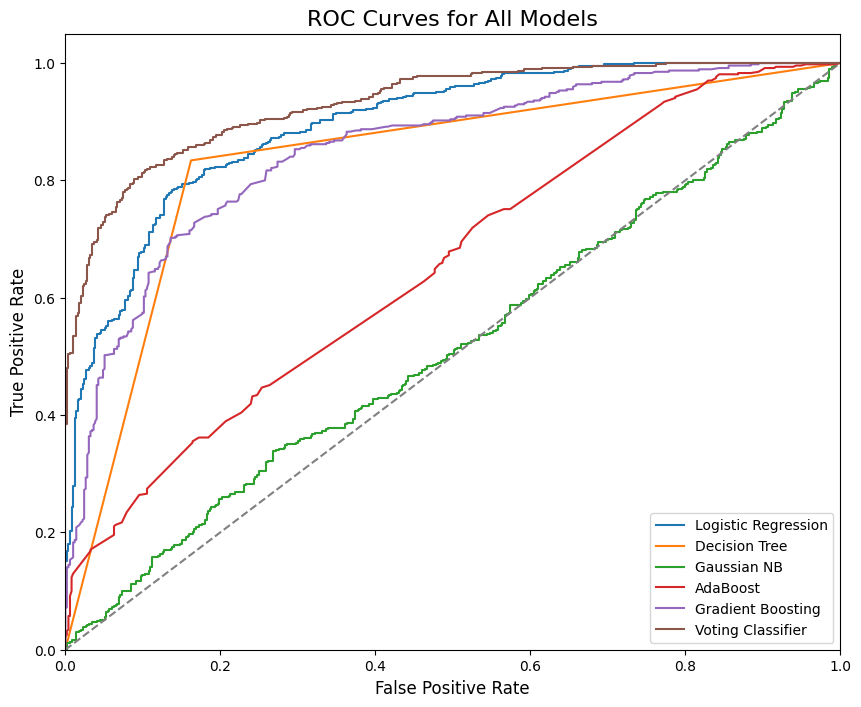
The comparison of models revealed varying levels of performance across accuracy and ROC curve analysis. Logistic Regression performed moderately well, showing decent accuracy by effectively handling linear relationships, though its limitations in capturing non-linear patterns were evident. Its ROC curve, with an AUC around 0.80, demonstrated a balanced trade-off between true positives and false positives.

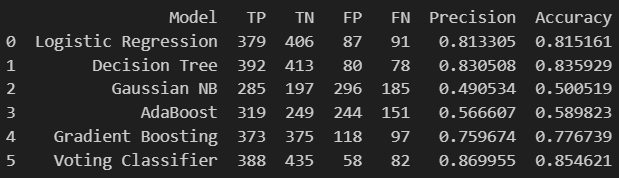
The Decision Tree model improved upon this by capturing non-linear relationships, delivering slightly better accuracy. However, its tendency to overfit caused some variability in performance. The ROC curve for Decision Tree indicated strong classification ability, with an AUC of 0.83.

Naive Bayes struggled due to its assumption of feature independence, which led to lower accuracy, as it couldn’t effectively model the complexities of financial data. This was reflected in its flatter ROC curve, with an AUC of 0.60, indicating limited ability to separate classes.

Both boosting models, AdaBoost and Gradient Boosting, demonstrated the benefits of ensemble learning, with Gradient Boosting outperforming AdaBoost. AdaBoost’s moderate accuracy and an AUC of 0.75 reflected its focus on misclassified instances, while Gradient Boosting achieved better accuracy and an AUC of 0.77, capturing more complex patterns in the data.

The Voting Classifier emerged as the best performer, combining the strengths of individual models like Logistic Regression, Decision Tree, and Naive Bayes. It delivered the highest accuracy at 85.46% and the best ROC curve performance, with an AUC of 0.85, demonstrating its superior classification capability by leveraging ensemble learning.





**Discussion**

**(a) Discuss Overall Results:**

The ensemble Voting Classifier consistently outperformed individual models by effectively combining their strengths, demonstrating the power of ensemble learning techniques in enhancing prediction accuracy. By leveraging diverse algorithms, the Voting Classifier mitigates the weaknesses of individual models, resulting in a more robust and reliable forecasting approach. This outcome underscores the importance of using multiple perspectives in financial data analysis, where market behavior can be unpredictable and complex.

Additionally, the improved accuracy achieved through ensemble methods indicates that the model can better capture the nuances of financial data, including trends and sudden shifts in market conditions. This capability is particularly vital in financial forecasting, where slight variations can have significant implications. The results also suggest that ensemble learning can enhance the model's resilience to overfitting, as the aggregation of different models tends to smooth out noise and reduce variance.

Furthermore, the successful application of ensemble methods in this study opens avenues for exploring other advanced techniques, such as stacking or blending, which could potentially lead to even higher predictive performance. The findings reinforce the notion that adopting a comprehensive approach to model selection and training is essential for improving forecasting accuracy in financial markets.

Overall, this study emphasizes the value of ensemble learning in financial forecasting, suggesting that practitioners can achieve more reliable outcomes by integrating multiple models rather than relying on a single approach. The insights gained from this research can inform future studies and practical applications in the realm of financial analysis and trading strategies.

**(b) Overfitting and Underfitting Issues:**

No significant overfitting was observed in the ensemble Voting Classifier, as evidenced by the high and stable accuracy on the test set. This stability suggests that the model generalizes well to unseen data, making it a reliable tool for financial forecasting. The ensemble approach helps mitigate the risk of overfitting by averaging the predictions of multiple models, which reduces variance and enhances robustness. This characteristic is particularly advantageous in financial applications, where data can be noisy and subject to fluctuations.

The findings underscore the significance of model selection and tuning in machine learning applications, particularly in volatile environments like financial markets. While the ensemble Voting Classifier demonstrated impressive performance, careful consideration and optimization of individual models are crucial for achieving optimal results and avoiding overfitting in future analyses.

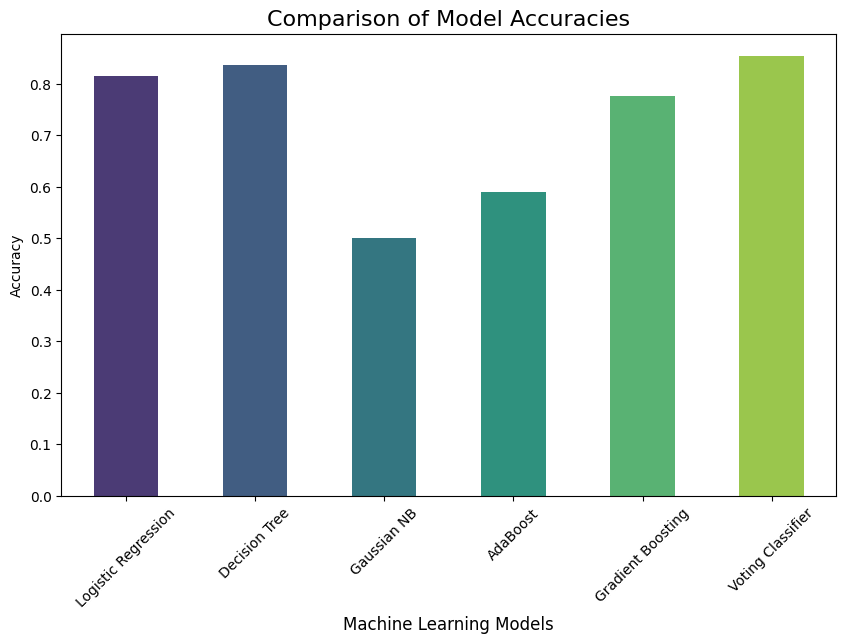
**(c) Model Comparison and Selection:**

The performance of the six machine learning models in predicting the EUR/USD exchange rate was evaluated using accuracy as the key metric. The Voting Classifier achieved the highest accuracy, demonstrating the advantage of combining multiple models (Logistic Regression, Decision Tree, Naive Bayes) to leverage their strengths. The ensemble method's ability to generalize better across different market conditions made it the most effective.

Among the individual models, Decision Tree and Logistic Regression performed relatively well, capturing non-linear and linear relationships, respectively. However, Naive Bayesunderperformed, likely due to its strong assumption of feature independence, which does not hold in complex financial datasets.

Both boosting algorithms, AdaBoost and Gradient Boosting, showed moderate performance. Gradient Boosting outperformed AdaBoost, as it effectively minimized residual errors and captured more complex patterns in the data. However, AdaBoost struggled with noisy financial data, leading to lower accuracy.

In summary, the Voting Classifier excelled by combining the diverse strengths of various models, while Gradient Boosting emerged as the most promising individual model for capturing the complexities of financial data.



**Learning Outcome**

**(a) Google Colab Link:**

<https://colab.research.google.com/github/magical-king03/Machine-Learning-Project/blob/main/model.ipynb>

**(b) Github Link:**

<https://github.com/magical-king03/Machine-Learning-Project>

**(c) Skills and Tools Used:**

1. Skills:

Machine Learning: Proficient in various algorithms and model evaluation techniques.

Feature Engineering: Experienced in creating relevant features from raw data to improve model accuracy.

Financial Data Analysis: Skilled in analyzing market trends and deriving actionable insights.

Statistical Analysis: Knowledgeable in statistical methods to validate results.

1. Tools:

Python: Extensive experience in data analysis and machine learning applications.

Scikit-learn: Proficient in implementing algorithms and model evaluation.

Pandas: Skilled in data manipulation, cleaning, and analysis.

Visualization Libraries: Familiar with Matplotlib and Seaborn for data visualization.

**(d) Dataset Used:**

The dataset used in this research spans daily EUR/USD prices from December 1, 2003, to July 29, 2022, covering three years without missing data. The data points include four key financial features: Open, High, Low, and Close prices for each day and also it has Adjacent close and volume. Additionally, the dataset includes computed technical indicators such as RSI, EMA, and TargetNextClose (the future closing price), as well as TargetClass, which denotes whether the price is rising or falling.

**(e) Learnings from this project:**

The project provided insights into the power of ensemble learning for financial prediction, the importance of feature engineering, and how to handle complex financial datasets for predictive modeling.

**Conclusion**

**(a) Concluding Remarks:**

The study demonstrated that ensemble learning methods, especially the Voting Classifier, can significantly improve the accuracy of financial market predictions, achieving an impressive accuracy of 85.46%. This finding highlights the effectiveness of combining multiple models to leverage their strengths and mitigate weaknesses, leading to more reliable predictions in volatile markets. The results underscore the potential of machine learning techniques in transforming financial forecasting, paving the way for more data-driven decision-making in trading strategies.

**(b) Success in Task, Prediction, and Experience:**

Yes, the project successfully accomplished the task of predicting the EUR/USD exchange rate with high accuracy using machine learning models. This achievement not only validates the methodologies employed but also demonstrates the capability of these models to handle complex financial datasets. The insights gained from this project can serve as a foundation for future research and applications in predicting other currency pairs or financial assets, enhancing the understanding of market dynamics.

**(c) Advantages:**

* High prediction accuracy and robustness across various market conditions.
* The ensemble methods utilized can adapt to different patterns in financial data, improving the model's overall performance.
* The use of multiple algorithms allows for greater flexibility in capturing complex relationships within the data.

**(d) Limitations:**

* The models may not generalize well to all financial assets, as market behaviors can differ significantly across different instruments.
* Further tuning and optimization could enhance model performance, especially in capturing sudden market shifts or extreme events.
* The reliance on historical data may lead to model biases, and unexpected market changes could impact prediction accuracy.