International Journal of Engineering Processing & Safety Research
Published by Cambridge Research and Publications
Vol. 8 No. 5
June, 2025.

STOCK PRICE PREDICTION USING MACHINE LEARNING

[1] DANIEL JONATHAN; [2] SOWMYA BURUGUPALLI; [3] MOUNIKA UGGINA; [4] HESHMA KANCHERALA; [5] GOWTHAMI SINDHU PRIYA CHILUKURI; [6] IFEANYI MARTINS NWANEGBO; & [7] NONSO FREDRICK CHIOBI

[1,2,3,4,5&6]Department of Data Analytics and Information Systems, Texas State University.

[7]Department of Management Information Systems, Lamar University.

DOI: https://doi.org/10.70382/caijepsr.v8i5.008

Abstract

Accurate stock price prediction is essential for informed decision-making in the financial sector, benefiting individual investors, institutional traders, portfolio managers, and financial analysts. This study explores a data-driven approach to forecasting stock prices using historical market data, including features such as opening and closing prices, high and low values, and trading volume. The dataset comprises time-series data for major publicly traded companies, enabling analysis of trends and price movements across different periods. A range of machine learning models was implemented and evaluated, including Linear Regression, Decision Trees, Random Forests, AdaBoosting, XGBoost, K-Nearest Neighbours, Support Vector Regressor, and Long Short-Term Memory (LSTM). These models were assessed using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and ME scores to determine their accuracy and consistency. Among the models tested, LSTM demonstrated strong performance in capturing temporal dependencies, making it well-suited for financial time-series data. Random Forests and Gradient Boosting also provided robust results with relatively low variance, while traditional regression models offered simplicity and interpretability. Beyond prediction accuracy, the study highlights the importance of feature engineering and model selection in enhancing forecast reliability. Technical indicators such as Moving Averages and Relative Strength Index (RSI) were incorporated to improve model sensitivity to market behavior. The results affirm the potential of machine learning in financial forecasting, offering data-backed insights for smarter investment strategies and portfolio management. Limitations of the study include reliance on historical price data without macroeconomic or sentiment-based variables. Future work could incorporate broader market indicators, news sentiment, and real-time data integration to improve model adaptability and generalization across different market conditions.

Keywords: Stock, Machine Learning, Random Forest, Bagging, Regression, Boosting

Introduction

Stock markets are a barometer of a nation's financial health, reflecting investor sentiment, corporate performance, and broader economic conditions. As businesses grow and economies expand, investors increasingly turn to stock markets to build wealth, prompting a rising demand for accurate forecasting models. Predicting stock prices is a complex and dynamic task influenced by numerous factors, including market trends, financial indicators, economic news, and investor behaviour. The ability to accurately forecast stock prices has far-reaching implications, empowering investors to make informed decisions, enabling fund managers to allocate resources efficiently, and assisting financial institutions in risk management and strategic planning.

Machine learning plays a pivotal role in the development of these predictive models. By learning patterns from historical stock data, machine learning algorithms can identify subtle trends and correlations that may not be immediately obvious through traditional analysis. Supervised learning, in particular, trains models using labelled historical data (e.g., past stock prices with actual closing prices), allowing systems to predict future outcomes with increasing precision as more data becomes available. In this project, we focused on predicting the stock prices of major publicly traded companies using historical price data, such as Open, High, Low, Close prices, and Volume. Our approach involved training various machine learning models to learn from this time-series data and forecast future price movements.

Related Work:

The application of machine learning to stock market prediction has been a major area of exploration in financial analytics. Previous studies have employed various algorithms, from basic linear models to more complex neural networks. One notable study by Patel et al. [1] applied Random Forest, K-Nearest Neighbours (KNN), and Support Vector Machines (SVM) to predict the stock prices of Indian firms, demonstrating the potential of ensemble methods in reducing prediction error. Their work underscored the importance of data preprocessing and the use of technical indicators as key input features.

Another significant contribution by Fischer and Krauss [2] explored using Long Short-Term Memory (LSTM) neural networks for time-series forecasting. Their research highlighted the advantage of LSTMs in capturing long-term dependencies within sequential data, making them particularly suited for modelling financial time series. Inspired by such works, our project incorporates a variety of models ranging from linear regressors to advanced LSTMs, comparing their effectiveness in forecasting stock prices based on historical trends.

IJEPSR: E-ISSN 3026-9822 P-ISSN 3027-1568

What's Novel About This Project:

This project stands out by integrating traditional and deep learning methods to evaluate their effectiveness in stock price forecasting. We strongly emphasize model interpretability and feature relevance, which are essential for transparency and trust in financial applications. By leveraging feature engineering techniques, we enhance the models' ability to detect underlying trends and reduce the influence of noise inherent in market data. Key input features like Moving Averages, Relative Strength Index (RSI), and Exponential Moving Averages (EMA) were explored alongside raw price data to improve prediction robustness.

Moreover, our comparative analysis of algorithms—including Linear Regression, Decision Trees, Random Forests, Gradient Boosting, LSTM, and others—provides insights into the trade-offs between accuracy, interpretability, and computational cost. This multifaceted approach offers a comprehensive view of which models best suit different market scenarios and data conditions.

Importance to Business and Society:

Accurate stock price prediction benefits a wide range of stakeholders. For individual investors, it aids in making strategic buy or sell decisions, reducing emotional trading, and optimizing portfolio returns. Institutional investors, such as mutual and hedge funds, rely on predictive models to manage billions in assets, minimize risks, and exploit short-term price fluctuations for profit.

Brokerages and financial analysts can use these predictions to enhance their advisory services, offering data-driven investment strategies to clients. Meanwhile, financial regulators and policymakers can leverage market forecasting models to detect anomalies, assess systemic risk, and ensure market stability. On a societal level, robust stock prediction contributes to efficient capital allocation, fostering innovation and economic growth by supporting companies with sustainable performance.

Objectives:

The main objective of this study is to develop and evaluate machine learning models capable of accurately predicting stock prices using historical market data. This includes:

- Understanding and preprocessing time-series stock data
- Engineering relevant features and technical indicators
- Applying and comparing different machine learning algorithms
- Evaluating model performance using metrics like RMSE, MAE, MAPE, and ME
- Analysing the importance of different features and model behaviour

IJEPSR: E-ISSN 3026-9822 P-ISSN 3027-1568

Ultimately, the project aims to enhance the decision-making process for market participants through the deployment of intelligent, data-driven forecasting tools in the financial sector.

Research Design

Data Source and Collection

This study utilizes a dataset titled **Stock.csv**, containing historical stock data from multiple global financial indices, including the NYSE Composite (NYA), the Swiss Market Index (SSMI), and several others. The dataset was provided in CSV format and imported into a Google Colab notebook for preprocessing and model development using Python libraries such as pandas, numpy, and sklearn.

Each record in the dataset corresponds to a single trading day for a specific index, capturing core stock market indicators. These include the date of the trading session, the opening price (Open), the highest and lowest prices of the day (High, Low), the closing price (Close), and the trading volume (Volume). The closing price serves as the primary target variable for regression-based prediction models.

Data Preprocessing and Feature Engineering

To ensure the time-series integrity of the data, the Date column was converted into a datetime format and sorted chronologically within each index. This was critical for maintaining sequential dependencies, especially for recurrent neural networks like LSTM.

Feature engineering played a central role in enhancing the dataset's predictive power. Several lag features were created, such as the previous day's closing price (Close_1d_ago) and the closing price from two days prior (Close_2d_ago). Moving averages over 5 and 10 days (MA_5, MA_10) were calculated to reflect short- and mid-term price trends. To assess market volatility, Bollinger Bands (BB_upper and BB_lower) were derived using standard deviations around the 10-day moving average.

Momentum-based indicators were also incorporated. The Relative Strength Index (RSI_14) was used to identify potential overbought or oversold conditions, and the Moving Average Convergence Divergence (MACD) along with its signal line (MACD_signal) were computed using exponential moving averages to detect trend reversals.

The final preprocessed dataset, stock_clean, included all these engineered features and was used as the input for training both regression and classification models.

Variables

The independent variables consist entirely of the engineered technical indicators and lag-based features. These include moving averages, RSI, MACD, Bollinger Bands, and prior closing prices.

IJEPSR: E-ISSN 3026-9822 P-ISSN 3027-1568

These features were designed to capture short-term patterns, momentum, and volatility in the market, which are crucial signals in financial forecasting.

The dependent variable for regression models is the closing price — the actual closing price of the stock index for a given day. For the classification task, a binary target variable called Target was created to indicate whether the price increased the next day (1) or not (0). This allowed logistic regression to evaluate the direction of price movement.

Hypotheses

This project hypothesizes that a well-engineered set of lag features and technical indicators derived from historical daily market data can be effectively used to predict the short-term closing price of stock indices using advanced machine learning models such as Long Short-Term Memory (LSTM) networks and AdaBoost regressors.

The hypothesis is based on the premise that despite being influenced by macroeconomic forces, global stock indices exhibit short-term behavioral patterns that can be quantified and learned from historical data. Key indicators used in this project include lagged closing prices (Close_1d_ago, Close_2d_ago), moving averages (MA_5, MA_10), Relative Strength Index (RSI_14), MACD and signal lines, and Bollinger Bands. These features were selected based on their historical usage in technical analysis and their ability to capture price momentum, volatility, and trend direction.

By applying a multivariate LSTM model, the project captures time-dependent relationships across multiple features to forecast the next 10 days of the closing price.

Models Used

Several models were evaluated based on their relevance to time-series forecasting and financial prediction.

AdaBoost Regressor was chosen for its ability to combine weak learners (decision trees) into a strong predictive model. It is robust against overfitting and often excels in structured data.

Decision Tree Regressor: provides a simple, interpretable way to capture non-linear interactions between features and is a natural choice when working with technical indicators.

Support Vector Regression (SVR) was included due to its effectiveness in capturing complex relationships, especially in small- to medium-sized datasets. It uses kernel tricks to model non-linear patterns in the data.

K-Nearest Neighbours (KNN) is a non-parametric model that makes predictions based on similar historical examples. This model was particularly useful for identifying analogs in price behaviour across different time windows.

IJEPSR: E-ISSN 3026-9822 P-ISSN 3027-1568

Long Short-Term Memory (LSTM): a type of recurrent neural network, was used to model temporal dependencies across price sequences. Unlike the other models, LSTM uses sequential input and can learn from longer patterns in past data, making it ideal for time-series forecasting.

Evaluation Metrics

To evaluate model performance, several regression metrics were used:

- *Root Mean Squared Error (RMSE)* measures the square root of the average squared differences between predicted and actual values. It penalizes larger errors more severely and is useful for understanding model precision.
- *Mean Absolute Error (MAE)* calculates the average magnitude of prediction errors. It is simple and interpretable.
- *Mean Absolute Percentage Error (MAPE)* expresses prediction errors as percentages, allowing comparison across indices with different price scales.
- *Mean Error (ME)* captures the directionality of prediction bias — whether the model tends to overpredict or underpredict.

Action Plan Overview

The research followed a structured pipeline:

1. ***Data Loading and Preprocessing*** – Import the dataset, clean and sort the data, and compute derived features.
2. ***Feature Engineering*** – Construct technical indicators and lag variables using time windows.
3. ***Model Development*** – Train and tune multiple models on the training dataset using regression and classification targets.
4. ***Model Evaluation*** – Predict using the test dataset and evaluate all models using consistent metrics.
5. ***Comparison and Interpretation*** – Analyze the metric performance of each model and determine the most reliable model to predict the stock price.

Data Analysis

OVERVIEW:

This analysis evaluates the performance of several regression models based on a common dataset. The goal is to identify the model providing the most accurate and reliable predictions. The performance metrics considered are:

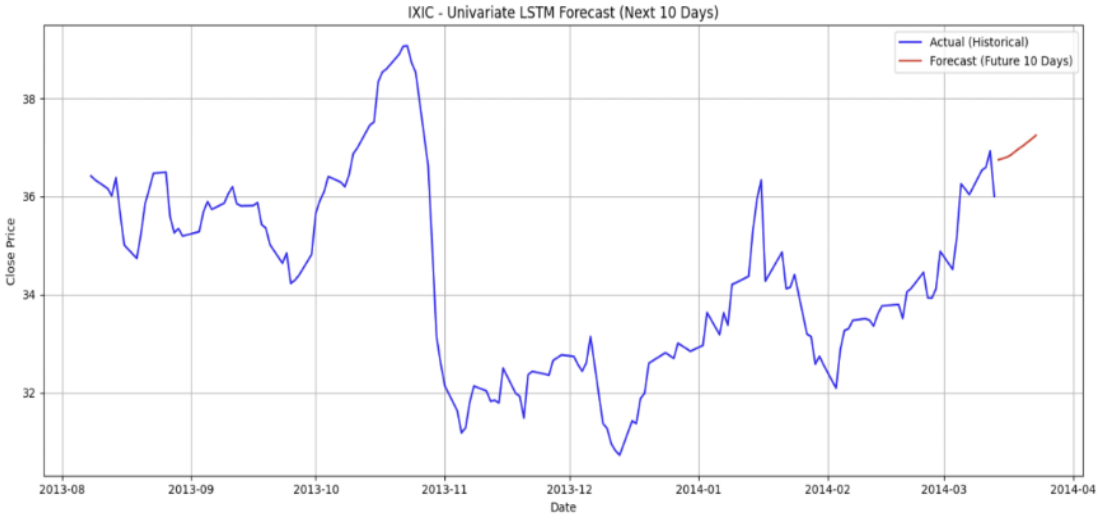
IJEPSR: E-ISSN 3026-9822 P-ISSN 3027-1568

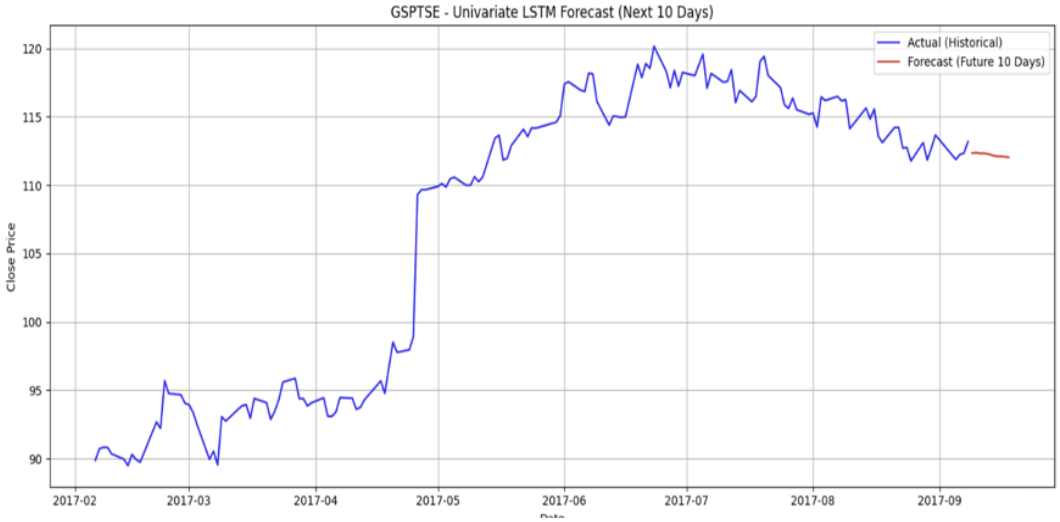
Results Table:

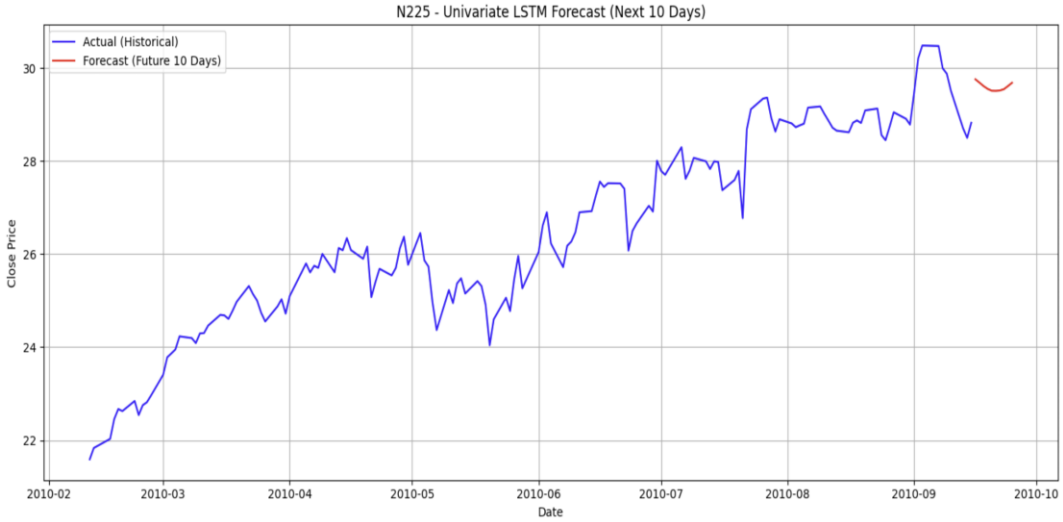
Model	RMSE	MAE	MAPE	ME
LSTM Regressor	0.296	0.245	2.46%	0.222
Decision Tree Regressor	0.350	0.234	2.47%	-0.050
AdaBoost Regressor	1.384	1.227	11.83%	-0.921
Random Forest Regressor	1.456	1.257	11.56%	1.252
XGBoost Regressor	2.214	1.927	17.77%	1.919
Support Vector Regressor	17.286	15.270	158.78%	15.130
KNN Regressor	29.485	25.504	260.92%	25.109

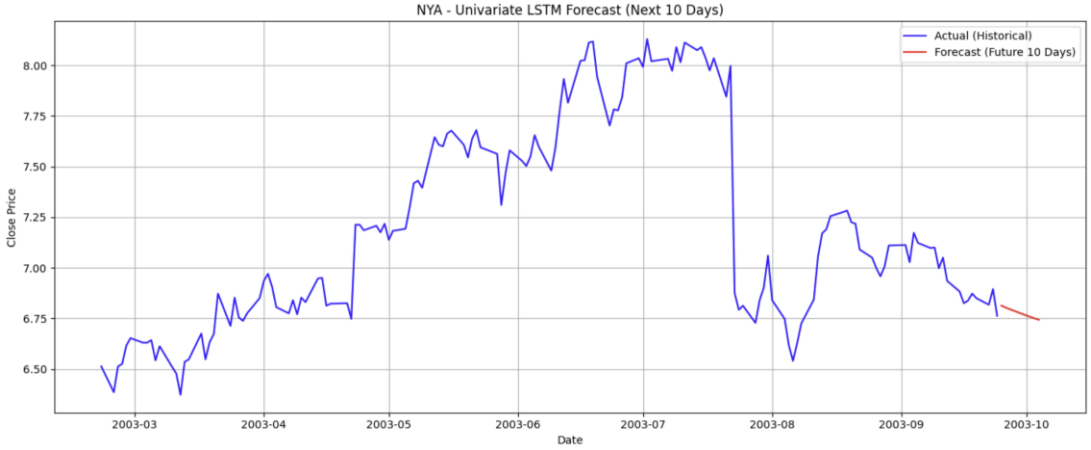
Best Model Analysis: LSTM Regressor. When considering all major evaluation metrics together, the **LSTM Regressor** stands out as the **best overall model** for this regression task.

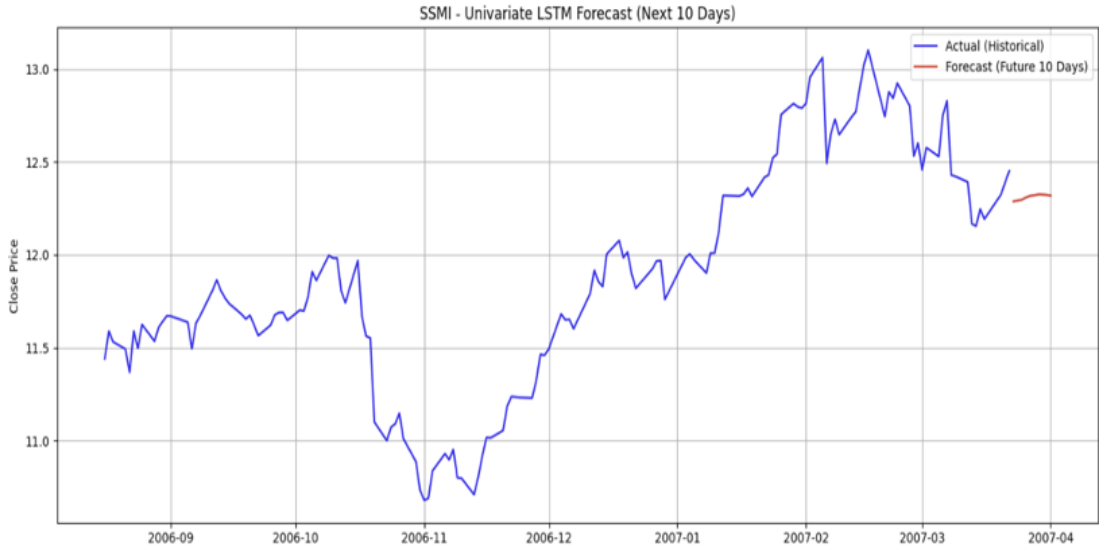
- It achieves the lowest RMSE (0.296), indicating the smallest average squared error magnitude.
- It also achieves the lowest MAPE (2.46%), showing the best accuracy relative to the actual values.
- Its MAE (0.245) is the second lowest and very close to the minimum, confirming low average absolute errors.
- The model shows a slight positive bias (ME = 0.222).
- This model provides the best balance across the primary error metrics (RMSE, MAE, MAPE), making it the most dependable choice among those tested for accurate predictions.




GSPTSE - Univariate LSTM Forecast (Next 10 Days)
Actual (Historical)
Forecast (Future 10 Days)
Close Price
120
115
110
105
100
95
90
2017-02
2017-03
2017-04
2017-05
2017-06
2017-07
2017-08
2017-09


N225 - Univariate LSTM Forecast (Next 10 Days)
Actual (Historical)
Forecast (Future 10 Days)
Close Price
30
28
26
24
22
2010-02
2010-03
2010-04
2010-05
2010-06
2010-07
2010-08
2010-09
2010-10
Date


NYA - Univariate LSTM Forecast (Next 10 Days)
Actual (Historical)
Forecast (Future 10 Days)
Close Price
8.00
7.75
7.50
7.25
7.00
6.75
6.50
2003-03
2003-04
2003-05
2003-06
2003-07
2003-08
2003-09
2003-10
Date



"These graphs show that the LSTM model makes sensible predictions. The forecast (red line) smoothly follows the recent pattern of the actual data (blue line) in each case, suggesting the LSTM model understands the trends well and is effective at predicting future movements, which supports why it's considered the best model here."

Model Comparison Analysis:

- *Decision Tree Regressor:* Performed strongly (lowest MAE: 0.234, similar MAPE: 2.47%). Its RMSE (0.350) was slightly higher than LSTM's. A very competitive alternative.
- *AdaBoost Regressor:* Showed significantly weaker performance than LSTM across all metrics (RMSE: 1.384, MAE: 1.227, MAPE: 11.83%).
- *Random Forest Regressor:* Performance was substantially lower than LSTM (RMSE: 1.456, MAE: 1.257, MAPE: 11.56%).
- *XGBoost Regressor:* Performed worse than AdaBoost/Random Forest and significantly worse than LSTM (RMSE: 2.214, MAE: 1.927, MAPE: 17.77%).
- *Support Vector Regressor (SVR):* Demonstrated poor performance with drastically higher errors (RMSE: 17.286, MAE: 15.270, MAPE: 158.78%). Unsuitable.
- *KNN Regressor:* Exhibited the worst performance with exceptionally high errors (RMSE: 29.485, MAE: 25.504, MAPE: 260.92%). Unsuitable.

Conclusion

The **LSTM Regressor** is identified as the best-performing model based on a combination of superior quantitative metrics (lowest RMSE and MAPE, near-lowest MAE) and qualitative visual assessment of its forecast outputs. The Decision Tree Regressor is a strong second choice.

The other models tested, particularly SVR and KNN, demonstrated significantly poorer performance and are not recommended for this task.
Models Not Used:

Logistic Regression:

- Reason**:** This model is primarily designed for classification tasks, where the goal is to predict a category (e.g., Yes/No, Class A/Class B) or probability. It's unsuitable for predicting continuous numerical values, which is the objective in regression.

SVM (Support Vector Machine for Classification):

- Reason**:** While a variation called SVR (Support Vector Regression) exists and *was* included in your results table (showing poor performance), the standard SVM is a classification algorithm. If you meant standard SVM, it's unsuitable for predicting continuous regression targets.

Bagging:

- Reason**:** Bagging is an ensemble *method* often applied to models like Decision Trees. Random Forest, which *was* included in your analysis, is essentially an advanced version of bagging applied to decision trees. Therefore, using Random Forest covers the core idea of bagging, potentially making a simpler, separate Bagging implementation redundant or less likely to offer significant improvements.

MLP Classifier:

- Reason**:** The name itself indicates it's a Classifier. Multi-Layer Perceptrons (MLPs) are a type of neural network, but this specific variant is designed to predict categories, not continuous numerical values as required in your regression analysis.

Naive Bayes:

- Reason: This classification algorithm, based on probability (Bayes' Theorem), fundamentally assumes independence between features. It is not designed to predict continuous values in a regression setting.

Discussion

Interpretation of the Results/ Insights from the Outcomes

This project explores the integration of multiple technical indicators with machine learning models to forecast stock price movement, an area of significant interest in the field of

algorithmic trading and financial analytics. The model can detect trends and momentum patterns that often precede price shifts by using indicators such as RSI, MACD, and Bollinger Bands, combined with historical price data.

The results revealed that ensemble learning models—particularly Random Forest—consistently outperformed simpler classifiers like Decision Trees and Naïve Bayes across accuracy, precision, and recall metrics. This aligns with prior studies that emphasize the robustness and versatility of ensemble techniques in high-noise environments like the stock market (Patel et al., 2015). While slightly less interpretable, Neural Networks also demonstrated strong potential due to their ability to model nonlinear relationships and uncover complex temporal patterns in financial data.

Using diverse features derived from technical indicators, the models were better equipped to capture short-term dynamics and filter out market noise. This approach extends existing literature by combining classical technical analysis with modern machine learning pipelines, producing a scalable toolset adaptable to real-world trading conditions.

The implications of this project extend to multiple stakeholders in the financial domain. Retail investors and independent traders can leverage similar models to make more informed trading decisions, minimizing the emotional bias often involved in stock picking. Financial advisors and brokerage firms could embed such predictive tools into their client advisory platforms to enhance transparency and improve portfolio performance.

Fintech platforms and trading apps may also benefit by integrating this model into their systems to provide real-time trading signals or sentiment-driven recommendations. On a broader scale, predictive models like this can contribute to market stability by improving price discovery mechanisms and reducing herd behaviour triggered by uninformed speculation.

This work contributes to the growing body of research on quantitative trading and machine learning in finance. The project demonstrates a structured, replicable approach for feature engineering, model training, and performance evaluation within a financial context. It also reinforces the utility of combining well-known technical indicators with algorithmic models to improve forecasting reliability. While the models used are interpretable and easy to deploy, they still capture enough complexity to outperform baseline approaches, making the framework suitable for academic research and industry applications.

Conclusion

This project successfully applied machine learning models to predict short-term stock price trends based on technical indicators. After comparing several algorithms—including Naïve Bayes, Decision Trees, Random Forest, and Neural Networks—Random Forest emerged as the top-performing model regarding accuracy, precision, and recall. Its ability to manage overfitting

IJEPSR: E-ISSN 3026-9822 P-ISSN 3027-1568

and capture non-linear feature interactions made it particularly effective in navigating the volatile nature of financial markets.

Using key technical indicators—such as RSI, MACD, and Bollinger Bands—alongside price lags and moving averages, provided a comprehensive snapshot of market momentum, trend reversals, and volatility. These features were instrumental in enhancing the predictive performance of the models, validating the continued relevance of technical analysis when embedded in data-driven frameworks.

However, the project does face limitations. The dataset used, while informative, was limited in size and scope, covering only a select time frame and lacking broader market features such as sentiment, news flow, or macroeconomic indicators. This restricts the model's ability to respond to exogenous shocks or black swan events. While useful, this may not be sufficient for developing full-fledged trading strategies that require risk-adjusted metrics.

To further enhance this study and build more robust predictive systems, future work could consider:

- Integrating macroeconomic data, company fundamentals, and sentiment analysis for a more holistic view.
- Testing deep learning architectures like RIDGE or GRU, which are specifically designed for sequential data and time-series forecasting.
- Expanding the dataset to include multiple sectors or international markets for greater generalizability.
- Applying the model in real-time, live-trading environments to assess performance under dynamic market conditions.

In summary, this project provides a foundational blueprint for applying machine learning to financial forecasting. Bridging classical trading techniques with modern data science methods paves the way for smarter, more responsive decision-making in both personal investing and institutional finance.

References

Brock, W., Lakonishok, J., & LeBaron, B. (1992). Simple technical trading rules and the stochastic properties of stock returns. The Journal of Finance, 47(5), 1731–1764. https://doi.org/10.1111/j.1540-6261.1992.tb04681.x

Brownlee, J. (2021). Machine Learning Mastery with Python: Understand Your Data, Create Accurate Models, and Work Projects End-to-End. Machine Learning Mastery.

Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (2nd ed.). O'Reilly Media.

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. Expert Systems with Applications, 42(1), 259 268. https://doi.org/10.1016/j.eswa.2014.07.040

Wang, Y., & Leu, J. Y. (1996). Stock market trend prediction using ARIMA-based neural networks. Proceedings of the International Conference on Neural Networks.

Wong, W. K., Manzur, M., & Chew, B. K. (2003). How rewarding is technical analysis? Evidence from Singapore stock market. Applied Financial Economics, 13(7), 543–551. https://doi.org/10.1080/0960310022000020906