J. Electrical Systems 20-11s (2024):1211-1219

[1]Jatin Navani

Dr. Nupur Giri

Can AI models Outperform the Traditional Buy-and-Hold Strategy?

Journal of Electrical Systems

Abstract: - The efficient market theory suggests that the 'buy and hold' (B&H) strategy is optimal due to its simplicity and lower costs. Artificial intelligence (AI) models, including machine learning (ML) is now a common research tool in financial applications. However, advancements in AI and machine learning offer the potential to outperform B&H. This paper investigates AI models applied to the Nifty 50 index, demonstrating that an AI-based strategy can surpass the performance of the traditional B&H approach. We compare various machine learning classifiers, such as LightGBM, KNN, XGB, SVC, and Random Forest, to evaluate their effectiveness. Additionally, we dive into the feature engineering process, converting a core set of financial indicators into comprehensive datasets for model training. A detailed Profit and Loss (PnL) comparison is conducted, revealing that our AI-based strategy not only outperforms the B&H strategy in general but also shows resilience during volatile market conditions. The findings highlights the significant potential of AI in developing superior trading strategies, offering insights into its practical applications in financial markets.

Keywords: stock trading, Machine Learning, Artificial Intelligence

INTRODUCTION

The efficient market theory in financial literature suggests that an investor cannot consistently outperform the markets using any investment tool, be it fundamental or technical analysis [1]. If this theory holds to be true, the best strategy which an investor should adopt is the 'buy and hold' strategy. Apart from being quite simple to execute or adopt, it involves lower transaction costs as well, compared to other more active strategies in investment. As the name does suggest, this B&H strategy refers to the purchasing of financial assets at the beginning of the period and holds them over some duration, usually a year, to realize returns at the end of that particular period.

Although the strategy of B&H has a quite strong theoretical background and is considerably seductive in practice, recent developments in sophisticated computational methods and easy access to data raised interest regarding whether AI models could create trading strategies that would outperform the B&H. AI and ML technologies have already disrupted a number of industries, including finance, by providing advanced tools for processing large and complex datasets to reveal hidden patterns. The usage of AI & ML for solving problems involving time-series data has gained immense popularity [14, 15, 16, 17].

This paper studies how AI models could perform better than B&H strategy with a focus on the Nifty 50 index. We use machine learning methods in building a reliable trading strategy targeted primarily at higher returns but also looking for consistency.

I. RELATED WORK

Forecasting and prediction techniques are extensively researched in the field of finance. These approaches frequently make use of machine learning and statistical techniques. A thorough review of the most recent techniques in this field is given by recent research works [18, 19, 20, 21], which also show how popular deep learning-based stock price prediction techniques are becoming.

[1]Department of Computer Engineering, Vivekanand Education Society's, Institute of Technology Chembur, Mumbai 400074
2022.jatin.navani@ves.ac.in
[2]Head of Department, Department of Computer Engineering, Vivekanand Education Society's Institute of Technology Chembur, Mumbai 400074
nupur.giri@ves.ac.in

A. *Classical Machine Learning Methods*

Classical machine learning algorithms are used in a large body of previous research literature to forecast future trends and prices in financial markets. Support vector regression is used by Yang et al. [26] to predict financial outcomes. They suggest extending the standard SVR model to take volatility changes in financial data into consideration. Usmani et al. [25] use a variety of machine learning techniques, including attributes like oil rates, gold and silver rates, and foreign exchange rates, to predict both positive and negative behaviour for the Karachi Stock Exchange (KSE). Ballings et al. [27] use ensemble techniques like Random Forest, AdaBoost, and Kernel factory in place of more conventional machine learning algorithms like neural networks, logistic regression, support vector machines, and K-Nearest Neighbour to forecast the direction of stock price movement.Patel et al. [28] compare the performance of four classifiers (Artificial Neural Networks, Support Vector Machines, Random Forest, and Naive Bayes) with two different approaches for input data, showing better performance with trend deterministic data over actual indicators.

B. *Deep Learning Methods*

Deep learning techniques have also been used in recent approaches for forecasting. With an average accuracy of 55.9% in 2017, Nelson et al. [14] investigate the use of recurrent neural networks, such as LSTM networks, to forecast future trends in stock prices using price history and technical indicators. [15] uses an LSTM-based model with an updated attention mechanism and empirical modal decomposition for improved accuracy to demonstrate how investor sentiment affects forecasts. [16] uses artificial and deep neural networks to predict the direction of the stock market's daily return, with the results confirmed by PCA. Levenberg-Marquardt, Bayesian Regularisation, and Scaled Conjugate Gradient techniques were effectively used by Selvamuthu et al. [29] to forecast stock market movements based on stock data.

C. *Statistical Methods*

For the purpose of projecting future data, time-series data studies commonly employ statistical techniques like Autoregressive Integrated Moving Average (ARIMA). For instance, [23] uses the ARIMA approach to forecast short-term stock prices on the NSE and NYSE. [22] suggests a hybrid model that combines kernel methods and Independent Component Analysis (ICA) to predict US indices. The time series data are typically produced by a linear process, according to these statistical methods [24]. By simulating the time series generation process, future values of the series are produced or forecast. Financial time series, on the other hand, are fundamentally complicated and non-linear. In order to tackle this, [25] employs a combination of Artificial Neural Networks and ARIMA on the KSE100 index.

Our review of the literature indicates that most previous studies did not take into account international factors when predicting stock prices. While not used for forecasting, some studies qualitatively examine how global indices affect the NIFTY50 index. Because stock markets are so complex and inefficient, technical indicators—which have been the mainstay of previous research—are insufficient on their own to make precise predictions. Global events that have a significant impact on Indian markets are not taken into account by technical indicators. In order to overcome these drawbacks, we create a hybrid model that more precisely predicts the NIFTY50 index by combining technical indicators and global market factors.

Furthermore, most research predicts the difference between the closing price and the previous closing price (CLOSE - Previous CLOSE). In contrast, our approach predicts the difference between the closing price and the opening price (CLOSE - OPEN). This allows us to make more immediate and actionable trading decisions based on the opening price of the day.

Additionally, it is important to address issues such as lookahead bias and data leakage, which can compromise the validity of predictive models. For example, [30] used indicators for the same day as the prediction, which resulted in unrealistically high accuracies greater than 90% due to lookahead bias. We have taken great efforts to make sure our model is free from lookahead bias and data leakage in order to prevent these problems. This includes turning off shuffling during model training, as shuffling can introduce lookahead bias in time-series data.

II. Feature Engineering

A. Initial Dataset

We began with a core set of 10 main datasets, each selected for its relevance and potential impact on market predictions, particularly in relation to the Nifty 50 Index:

- Nifty 50 Index: Benchmark Index of India representing the performance of the top 50 companies listed on the National Stock Exchange of India.
- India VIX: Volatility Index representing expected annual volatility in Nifty50 over the next 30 days [13].
- Gold: Historical price data of gold, specifically focusing on near-month futures, which are the futures contracts closest to their expiry date in a given month, as this provides the most relevant market signals [10].
- Crude Oil: Price data of crude oil, also considering near-month futures for the same reason as Gold, to capture the most immediate market conditions [11].
- IND-USD Currency Reference Rate: Exchange rate data between Indian Rupee (INR) and US Dollar (USD), influencing export-import dynamics [9].
- DJIA (Dow Jones Industrial Average): Price-weighted index of 30 large, publicly owned companies in the United States [12].
- Nasdaq: Stock market index comprising over 3,000 stocks listed on the Nasdaq Exchange [12].
- CAC (France Index): Stock market index representing 40 largest publicly traded companies in France [12].
- Nikkei 225: Stock market index for the Tokyo Stock Exchange, representing 225 large companies in Japan, providing insights into Asian market trends [33].
- Nifty 50 Futures Prices: Near-month futures prices for the Nifty 50 Index, used for back testing.

B. Features

Variety of features were used on these 10 datasets to increase their predictive power.

- OPEN_PREVIOUS_HIGH_GAP: The difference between the opening price and the previous day's high.
- LOG_RETURN: The natural logarithm of the ratio of the previous day's closing price to the closing price two days prior.
- OPEN_GAP: The difference between the opening price and the previous day's closing price.
- EMA_Open_Diff : Exponential Moving Average (EMA) difference between opening prices over different periods (5 days, 54 days).
- EMA_LOG_RETURN_5, EMA_LOG_RETURN_54: EMA of logarithmic returns over 5-day and 54-day periods.
- EMA_LOG_RETURN_Diff: Difference between EMA of logarithmic returns over different periods.
- EMA_OPEN_GAP_5, EMA_OPEN_GAP_54: EMA of opening gaps over 5-day and 54-day periods.
- EMA_OPEN_GAP_Diff: Difference between EMA of opening gaps over different periods.
- EMA_Close_DIff: EMA difference between closing prices over different periods.
- ATR (Average True Range): Average of the true ranges over a specified period, reflecting market volatility based on the high, low, and close of the previous day.
- RSI (Relative Strength Index): Momentum oscillator measuring the speed and change of price movements based on the close of the previous day.

In essence, log returns represent the market's overall returns and volatility. To determine the direction of the market, we computed the EMA_LOG_RETURN_5 and EMA_LOG_RETURN_54, which are the slow and fast moving averages of log returns. As a benchmark for the model, the difference between these moving averages offers a reliable trend indicator.To measure market volatility, we included technical indicators like RSI and ATR.

The events that happen overnight due to natural disasters or wars may have a huge effect on markets. In this respect, features such as EMA_Open_Diff and OPEN_PREVIOUS_HIGH_GAP were added to capture the overnight volatility caused by these outside variables. Moving averages of the OPEN_GAP at five and twenty-two days were also calculated to find the difference and identify the overnight trend.

These features are commonly utilized in financial time series data [32]. Applying this methodology to all 10 time series datasets, we derived a total of 136 features. For the USD/INR dataset, where only a single price is available per day, most of these features are not applicable.

III. METHADOLOGY

A. Model Selection

Several machine learning classifiers were used to identify the most effective model for predicting market movements. The classifiers evaluated in our study include:

- LightGBM
- K-Nearest Neighbors (KNN)
- XGBoost (XGB)
- Support Vector Classifier (SVC)
- Random Forest

To ensure the reliability of the models, the dataset underwent thorough data cleaning and normalization before being divided into training and testing sets. Numerical values were scaled using Power Transformer. The training set comprised 85% of the cleaned and normalized data, while the remaining 15% was allocated to the testing set for model evaluation.

B. Model Used

The classifier's performance was evaluated based on its accuracy. Initially, accuracies were computed with shuffle ON during training:

It was observed that with shuffle ON during training, the accuracies were generally higher. However, using shuffle ON can introduce leakage and lookahead bias, leading to unrealistic performance metrics.

To overcome these issues, particularly lookahead bias, we opted to proceed with shuffle OFF during model training. This decision was made to ensure the reliabilty of our predictive models. The accuracies achieved with shuffle OFF are as follows:

By implementing shuffle OFF, we were able to create models that are resistant to both look ahead bias and data leakage, which improved the reliability of our predictions.

Among the evaluated models, LightGBM achieved the highest accuracy. LightGBM trains many decision trees successively. In this case, every new tree is developed to correct the mistakes made by former trees. This approach is repeated either until it achieves a predefined number of trees or the model performance improves. What makes LightGBM special is that it uses histogram-based algorithms that discretize continuous values of features into discrete bins. The gain in reduced memory usage and faster training is thereby drastic. One of the main features of LightGBM is that it supports categorical feature input directly; it does not require too much preprocessing. A technique named Gradient-Based One-Side Sampling(GOSS) and Exclusive Feature Bundling has been developed to enhance its efficiency and accuracy.

Model	Hyperparameters	Accuracy (shuffle ON)	Accuracy (shuffle OFF)	Precision	Recall	F1-Score
SVC	C=10, kernel='linear'	52%	51.2%	51%	50%	51%
KNN	n_neighbors=5	50.2%	48.1%	48%	49%	48%
Random Forest	n_estimators=150, random_state=42	52.8%	51.8%	52%	51%	52%
LightGBM	boosting_type='dart', colsample_bytree=0. 7205150416582854, learning_rate=0.07968893628963661, max_depth=4,n_estimators=300, num_leaves=46,reg_alpha=0.005634429560259627, reg_lambda=0. 6132579571381214, subsample=0.12996776915582092	53.9%	54.5%	56%	54%	55%
XGBoost	Default parameters [31]	54.1%	53.4%	53%	53%	53%

Table 1:Evaluation of Model

GOSS will pay more attention to the most important training instances, while EFB reduces the number of features by grouping mutually exclusive features together. The ensemble of trees in LightGBM is capable of capturing complex nonlinear relationships within large datasets. Now, this feature presents LightGBM as a very effective tool in financial predictions markets, to understand very subtle patterns to get an edge in developing profitable strategies for trading.

To optimize the parameters of LightGBM, we employed Bayesian optimization using Gaussian Processes through the gp_minimize function from the sklearn library [34].

C. Predictive Target

Our study aims to predict the daily difference between the closing and opening prices of the Nifty 50 Index. Specifically, we predict whether the closing price (CLOSE) will be higher or lower than the opening price (OPEN) for each trading day.

$$\text{Prediction} = \text{CLOSE} - \text{OPEN} \qquad (1)$$

- If the predicted value is 1, it indicates that the closing price will be greater than the opening price, suggesting a buy signal.
- If the predicted value is 0, it indicates that the closing price will be less than or equal to the opening price, suggesting a short position or no action.

D. Trading Strategy

Based on the predictions, we formulate our trading strategy as follows:

- Buy Signal (Prediction = 1): If the model predicts that the closing price will be higher than the opening price, we take a long position at the start of the day.
- Short Signal (Prediction = 0): If the model predicts that the closing price will be lower than or equal to the opening price, we take a short position at the start of the day.

IV. RESULTS AND PERFORMANCE COMPARISON

We evaluate results using a test data set comprising the last 15% of the total dataset. The 15% test data covers results from May 2022 onwards, providing insights into model performance in recent market conditions.

A. Profit and Loss (PnL) Comparison

Fig. 1 PnL Comparison (2022-04 to 2024-07)

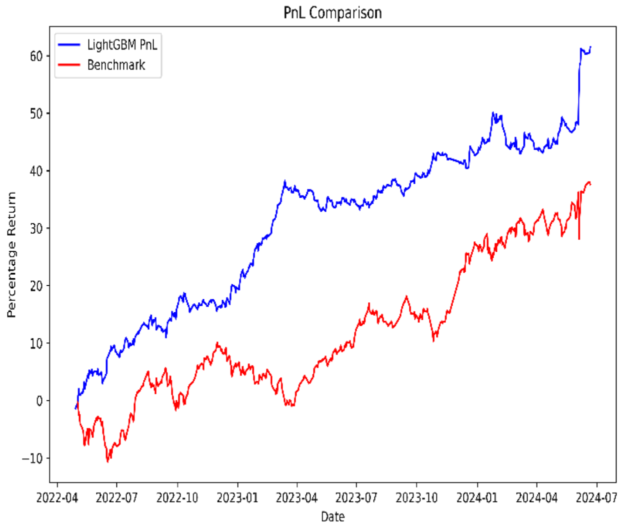


Figure 1 illustrates the Profit and Loss (PnL) comparison between our strategy and the Buy & Hold (B&H) strategy for the period from April 2022 to July 2024.

Our strategy significantly outperformed the B&H strategy during this period. While the B&H strategy yielded returns of ₹6,398 i.e. (37% returns) , our strategy achieved a return of ₹10,451 i.e (61% returns), demonstrating its superior performance.

B. Futures PnL Comparison (2022-04 to 2024-07)

Fig. 2 Cumulative PnL Comparison (2022-04 to 2024-07) After Transaction Charges.

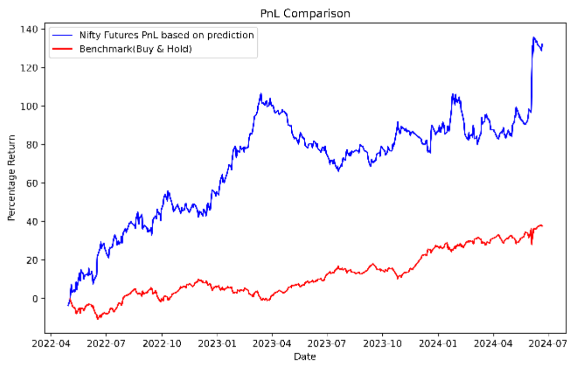


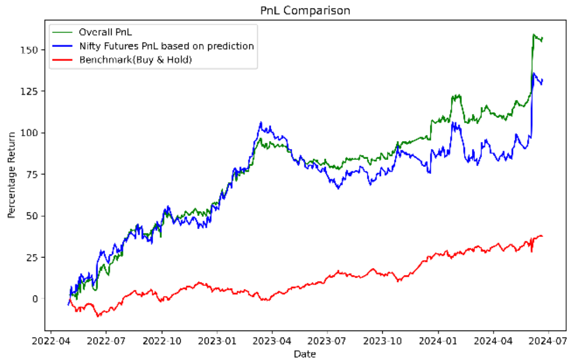
Figure 2 illustrates the cumulative Profit and Loss (PnL) comparison of our strategy with the Buy & Hold (B&H) strategy for Nifty 50 futures from April 2022 to July 2024, after transaction charges.

- Our Strategy (After Charges):
  - Initial Capital: ₹5,000 (1 quantity)
  - Total PnL: ₹6604.15 (Transaction Charge: ₹4 per day) i.e. (132% returns)
- B&H Strategy:
  - Initial Capital: ₹17,329 (as of 2022-04-29, when the Nifty was at 17,329.25)
  - Total PnL: ₹6398.55 i.e. (37% returns)

C. Combined ETF and Futures Investment Scenario

Hypothetically, we invested in a Nifty 50 ETF on 2022-04-29, when the Nifty was at 17,329.25. Considering a haircut, ₹15,000 was allocated for futures trading based on our predictions.

Fig. 3 Combined Investment Comparison (2022-2024)



- B&H Nifty 50 ETF Investment(1 quantity): Total return of ₹ 6398 i.e. (37% returns)
- Nifty 50 Futures Trading (3 quantity after considering haircut): Total return of ₹19,812 (132% returns)(After Transaction Charges : ₹12 per day)
- Overall PnL: ₹ 26,745 i.e. (157% returns)

D. Compounding Potential in Futures Trading

Unlike B&H and ETFs, where compounding is inherently built into the investment structure, futures trading traditionally does not benefit from automatic compounding due to the need to maintain margin requirements. However, by systematically reinvesting profits into additional positions, our strategy could effectively introduce a compounding effect, amplifying returns over time. This reinvestment approach would leverage the profits generated by the strategy to increase position sizes, potentially accelerating growth in cumulative returns. Thus, compounding through reinvestment in futures positions could serve as a powerful multiplier, further enhancing the performance gap between our AI-driven strategy and the B&H benchmark.

V. CONCLUSION

In this study, we developed a hybrid model to predict the Nifty 50 Index by integrating technical indicators and global market factors. Our approach addresses limitations of previous works by considering a wider range of influencing factors and mitigating issues such as lookahead bias and data leakage. Through comprehensive evaluation, we demonstrated that our strategy consistently outperforms the traditional Buy & Hold strategy.

Our findings show that when carefully trained and validated machine learning models can offer significant predictive power and resilience. By incorporating global indices and addressing potential biases, we provide a reliable framework for market prediction and trading.

REFERENCES

[1] Cohen, G., & Cabiri, E. (2015). Can technical oscillators outperform the buy and hold strategy? Applied Economics, 47(30), 3189–3197. https://doi.org/10.1080/00036846.2015.1013609

[2] DJIA. (2024, July 11). Retrieved from Yahoo Finance: https://sg.finance.yahoo.com/quote/%5EDJI/history/

[3] Nifty 50. (2021, April 9). Retrieved from NSE India: https://www.nseindia.com/reports-indices-historical-index-data

[4] USD-INR. (2021, April 9). Retrieved from MSE India: https://www.msei.in/markets/currency/historical-data/rbireferenceratearchives

[5] CAC. (2024, July 11). Retrieved from Yahoo Finance: https://finance.yahoo.com/quote/%5EFCHI/history/.

[6] Nasdaq. (2024, July 11). Retrieved from Yahoo Finance: https://finance.yahoo.com/quote/%5EIXIC/history/

[7] INDIA VIX. (2024, July 11). Retrieved from NSE India: https://www.nseindia.com/reports-indices-historical-vix

[8] Gold and Crude Oil. (2024, July 11). Retrieved from MCX India: https://www.mcxindia.com/market-data/bhavcopy

[9] Nandan, Tanuj, Pravin Kumar Agrawal, and Ashutosh Pratap Singh. "Empirical Relationship between Nifty 50 and Exchange Rate with Reference to India." Social Science and Development Policy: 67.

[10] Singh, R. P., & Kishor, N. (2014). Co-Integration of Gold Price Movement with Nifty Indices: A Study in Indian Context. Transnational Corporations Review, 6(1), 42–57. https://doi.org/10.1080/19186444.2014.11658380

[11] SriRam, P. "A study on the impact of crude oil prices on stock market, commodity market and select macroeconomic variables." (2017).

[12] Sowmya, Budampati VS, and Enugu Akhila Reddy. "A Study On Relationship Between Select Global Indices And NIFTY." Aweshkar Research Journal 29.1 (2022).

[13] Shaikh, I., Padhi, P. Inter-temporal relationship between India VIX and Nifty equity index. Decision 41, 439–448 (2014). https://doi.org/10.1007/s40622-014-0046-0.

[14] Nelson, D. M., Pereira, A. C., & de Oliveira, R. A. (2017, May). Stock market's price movement prediction with LSTM neural networks. In 2017 International joint conference on neural networks (IJCNN) (pp. 1419-1426). IEEE.

[15] Jin, Z., Yang, Y., & Liu, Y. (2019). Stock closing price prediction based on sentiment analysis and LSTM. Neural Computing and Applications, 1-17.

[16] Zhong, X., & Enke, D. (2019). Predicting the daily return direction of the stock market using hybrid machine learning algorithms. Financial Innovation, 5(1), 1-20.

[17] Selvamuthu, D., Kumar, V., & Mishra, A. (2019). Indian stock market prediction using artificial neural networks on tick data. Financial Innovation, 5(1), 1-12.

[18] Cavalcante, R. C., Brasileiro, R. C., Souza, V. L., Nobrega, J. P., & Oliveira, A. L. (2016). Computational intelligence and financial markets: A survey and future directions. Expert Systems with Applications, 55, 194-211.

[19] Sharma, A., Bhuriya, D., & Singh, U. (2017, April). Survey of stock market prediction using machine learning approach. In 2017 International conference of electronics, communication and aerospace technology (ICECA) (Vol. 2, pp. 506-509). IEEE.

[20] Hu, Z., Zhao, Y., & Khushi, M. (2021). A survey of forex and stock price prediction using deep learning. Applied System Innovation, 4(1), 9.

[21] Thakkar, A., & Chaudhari, K. (2021). Fusion in stock market prediction: A decade survey on the necessity, recent developments, and potential future directions. Information Fusion, 65, 95-107.

[22] Ince, H., & Trafalis, T. B. (2017). A hybrid forecasting model for stock market prediction. Economic Computation & Economic Cybernetics Studies & Research, 51(3).

[23] Ariyo, A. A., Adewumi, A. O., & Ayo, C. K. (2014, March). Stock price prediction using the ARIMA model. In 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation (pp. 106-112). IEEE.

[24] Kumar, D. A., & Murugan, S. (2013, February). Performance analysis of Indian stock market index using neural network time series model. In 2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering (pp. 72-78). IEEE.

[25] Usmani, M., Adil, S. H., Raza, K., & Ali, S. S. A. (2016, August). Stock market prediction using machine learning techniques. In 2016 3rd international conference on computer and information sciences (ICCOINS) (pp. 322-327). IEEE.

[26] Yang, H., Chan, L., & King, I. (2002, August). Support vector machine regression for volatile stock market prediction. In International Conference on Intelligent Data Engineering and Automated Learning (pp. 391-396). Springer, Berlin, Heidelberg.

[27] Ballings, M., Van den Poel, D., Hespeels, N., & Gryp, R. (2015). Evaluating multiple classifiers for stock price direction prediction. Expert systems with Applications, 42(20), 7046-7056.

[28] 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.

[29] Selvamuthu, D., Kumar, V., & Mishra, A. (2019). Indian stock market prediction using artificial neural networks on tick data. Financial Innovation, 5(1), 1-12.

[30] 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

[31] "XGBoost Parameters", XGBoost Documentation, Available: https://xgboost.readthedocs.io/en/stable/parameter.html

[32] J. D. Cryer and K.-S. Chan, Time Series Analysis With Applications in R, Springer Texts in Statistics, Springer, 2008. Available: https://mybiostats.wordpress.com/wp-content/uploads/2015/03/time-series-analysis-with-applications-in-r-cryer-and-chan.pdf

[33] Nikkei 225. (2024, July 11). Retrieved from Yahoo Finance: https://finance.yahoo.com/quote/%5EN225/history/

[34] Scikit-Optimize, “skopt.gp_minimize,” [Online]. Available: https://scikit-optimize.github.io/stable/modules/generated/skopt.gp_minimize.html.