UNIVERSITY OF ECONOMICS AND LAW

FACULTY OF FINANCE AND BANKING

-------☙📖❧-------

GRADUATION THESES

YEAR 2024

IDENTIFY THE FACTORS INFLUENCING THE MOVEMENT OF THE VN-INDEX THROUGH MACROECONOMIC NEWS AND TECHNICAL ANALYSIS BY MACHINE LEARNING

Instructor	**:** Dr. Pham Thi Thanh Xuan
Student	**:** Nguyen Thanh Phuong Uyen
Student Code	**:** K204141939
Class	**:** K20414C

Ho Chi Minh City, February 2024

UNIVERSITY OF ECONOMICS AND LAW

FACULTY OF FINANCE AND BANKING

GRADUATION THESES

YEAR 2024

IDENTIFY THE FACTORS INFLUENCING THE MOVEMENT OF THE VN-INDEX THROUGH MACROECONOMIC NEWS AND TECHNICAL ANALYSIS BY MACHINE LEARNING

Instructor	**:** Dr. Pham Thi Thanh Xuan
Student	**:** Nguyen Thanh Phuong Uyen
Student Code	**:** K204141939
Class	**:** K20414C

Ho Chi Minh City, February 2024

COMMENTS OF THE INSTRUCTOR

ACKNOWLEDGEMENT

The topic " Identify the factors influencing the movement of the VN-Index through macroeconomic news and technical analysis by machine learning " is the content I chose to research and write my graduation thesis after four years of studying the specialized master's Fintech program at school University of Economics and Law.

To complete the research process and complete this thesis, I received invaluable help, enthusiastic support, guidance, and encouragement, which enabled me to overcome challenges and successfully complete this meaningful yet demanding period.

I would like to express my deepest gratitude to all esteemed professors and lecturers of the Faculty of Finance and Banking at the University of Economics and Law, particularly to Dr. Pham Thi Thanh Xuan, my mentor. Her invaluable guidance and advice provided me with the necessary tools to navigate and successfully complete the internship. Her unwavering support and the conducive learning environment she created significantly contributed to my achievements during the graduate thesis complete period at school. I would like to express my heartfelt thanks to my family and friends for their unwavering support and encouragement, which served as a driving force for me to strive for improvement and successfully complete my internship report. In closing, I wish everyone good health, success in their careers, and happiness in life.

Acknowledging my limited theoretical knowledge and practical experience, I recognize that there may be shortcomings in my report. I eagerly await feedback from my teachers to enhance the clarity and quality of my work, as this will also contribute to my continuous learning and development for my future career endeavors.

TABLE OF CONTENTS

LIST OF TABLES, FIGURES

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS, SPECIAL CHARACTERS

ANN	Artificial neural network
HOSE	Ho Chi Minh City Stock Exchange
HNX	Hanoi Stock Exchange
RMSE	Root Mean Squared Error
KNN	K-Nearest Neighbor
SVM	Support Vector Machine
MSE	Mean Squared Error
MAPE	Mean Absolute Percentage Error
ML	Machine Learning
VNI	VN-Index
CPI	Consumer Price Index
SGX	Singapore Stock Market Index
U.S	United states
EU	European Union
WTI	West Texas Intermediate
ARDL	Autoregressive Distributed Lag
VAR	Vector auto regression
EDA	Exploratory Data Analysis
XGBoost	Extreme Gradient Boosting

ABSTRACT

Potential risks stemming from various causes have always presented challenges for both large and small investors, making the macroeconomic factors affecting stock market fluctuations a subject of particular interest in society. Rather than adhering to traditional analysis methods, this study proposes an alternative approach through machine learning techniques, which have become widely recognized tools for examining the effects of economic indicators on the Vietnamese stock market, a market that has experienced growth over the past 23 years in Vietnam. Utilizing data from the VN-Index from March 2016 to March 2024, the author has integrated various machine learning algorithms including Decision Tree Regression, Random Forest, XGBoost, KNN, ANN, SVM, and Linear Regression. The findings reveal that among the models evaluated, Random Forest stands out for its superior predictive performance. This model's ability to effectively manage nonlinear and complex data positions it as an excellent technical analysis choice for predicting the VN-Index's closing prices based on macroeconomic variables. While XGBoost and Decision Trees also demonstrated good performance with high R-Squared values, they did not match Random Forest. Linear Regression exhibited significantly lower R-Squared compared to Decision Trees and XGBoost, indicating its poorer data interpretation capability. Additionally, two other Machine Learning models, Support Vector Machine (SVM) and K-Nearest Neighbors (KNN), did not show good performance. Meanwhile, the Deep Learning model Artificial Neural Network (ANN) does not seem to be suitable for the data set and approach applied in this study. Through applying the SHAP method to the models during the study, the author identified four variables with the greatest impact on the Vietnamese stock market, in order: the S&P 500, money supply M2, exchange rates, and interest rates, with the S&P 500 and money supply M2 having the most significant effects. Moreover, global oil prices, the Consumer Price Index, and global gold prices were found to have a relatively minor impact and were not as influential as the aforementioned features.

Keywords: *VN-Index; Machine learning; Macroeconomic factors; Technical analysis; Vietnam stock market; Stock market fluctuations*

1. Introduction

1.1. Topic's Reasonable choice

The motivation behind selecting "Identifying the factors influencing the movement of the VN-Index through macroeconomic news and technical analysis by machine learning" as the research topic originates from the increasingly complex and dynamic nature of the financial market. It plays an indispensable role in attracting investment from both domestic and international investors for Vietnamese enterprises, while also serving as an effective capital channel for businesses looking to expand and develop. According to Mishkin (2004), the stock market in the financial system plays a crucial role in providing long-term capital for economic development. This market also acts as a mirror, reflecting the market's strength by evaluating the performance of large corporations. Hafer & Hein (2007) have pointed out that the development of the financial market, especially the stock market, is an essential component of a country's economic growth. Focusing on the VN-Index is aptly relevant as the stock market plays a pivotal role in shaping and adjusting national economic policies. Viewed as a key indicator of the economic health and growth potential of a nation's economy (GC, 2006), research by Antonios (2010) has unveiled a causal relationship from the stock market to economic development. Furthermore, the stock market's role in mitigating economic risk and its impact on the banking financial system is undeniable (Levine & Zervos, 1998). This significance renders the stock market an indispensable part of the economic development strategy in developing and emerging countries (Khan & Abdelhak, 2000). Dr. Tran (2018) also mentions that the stock market serves as an economic growth forecast indicator, where an upward market trend is often interpreted as a sign of upcoming economic expansion. While the Vietnamese stock market features indices like HNX Index, VN-30 Index, Upcom Index, etc., the VN-Index remains the most critical benchmark, widely utilized to reflect the overall fluctuations of the Vietnamese stock market. The market becomes more appealing due to investment promotion policies and rapid economic growth in Vietnam, especially with the VN-Index's 36% surge in 2021, indicating an overheated market condition.

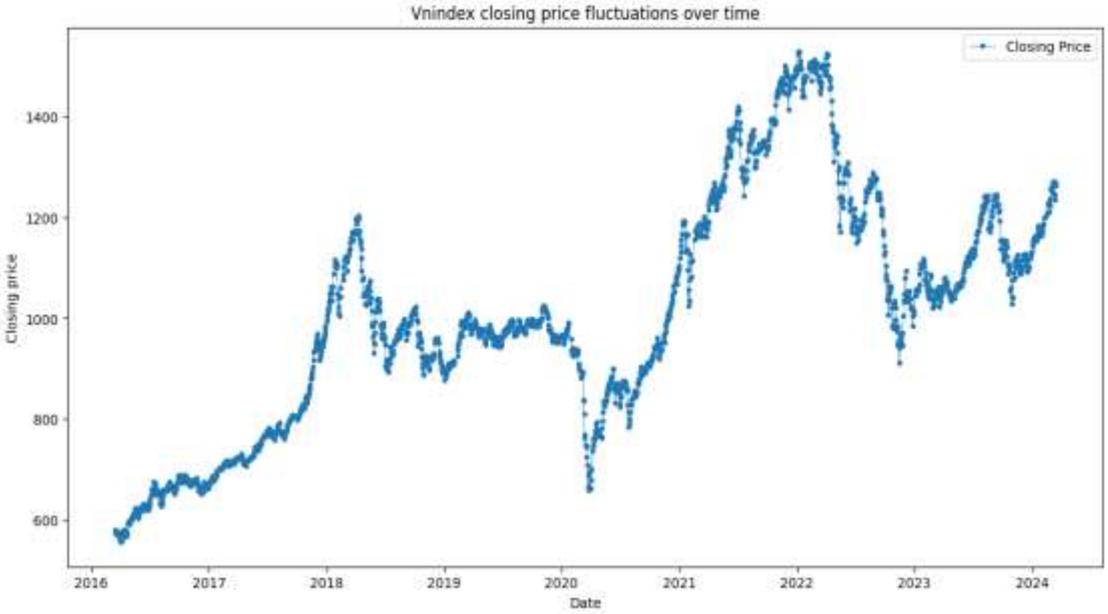

Figure 1. VN-Index from 2016 to 2024

Source: Author's summary

Despite its rapid development since its inception in 2000, the Vietnamese stock market is still regarded as an emerging market with a relatively small scale. Additionally, unpredictable risks from various sources continue to pose challenges for investors. Empirical studies have identified numerous factors affecting the stock market, stemming both from macroeconomic elements and company-specific factors. Significant research has delved into the impact of macroeconomic factors on stock prices and market efficiency, such as the works of Akbar and colleagues (2019); Singhal, Choudhary, and Biswal (2019); González and his team (2018), among others. Therefore, investigating and pinpointing the factors that influence stock market price fluctuations remains a primary concern for investors, corporations, and policymakers due to its contributions to economic development.

Advancements in computational power and technology have broadened human capabilities to tackle more complex problems. The application of machine learning algorithms is becoming increasingly widespread, with evidence showing they outperform traditional statistical methods in stock price forecasting (Bhattacharjee and Bhattacharja, 2019). While traditional financial analysis methods retain their value, they are increasingly facing limitations in capturing and predicting the multifaceted nature of market fluctuations. The integration of machine learning models represents a significant leap forward in financial analysis, offering the ability to process vast amounts of data, recognize patterns, and make predictions with an accuracy and efficiency unattainable through conventional means. Machine learning's prowess in handling large datasets, including structured data (like historical prices) and unstructured data (such as news articles), as well as its ability to uncover hidden relationships and patterns, makes it an ideal tool for this study. Utilizing a variety of models, including Decision Trees, Random Forests, XGBoost, and others, enables a comprehensive analysis that can adapt to the inherent complexities of the stock market.

Furthermore, this study aims to bridge the gap between theoretical knowledge and practical application. By identifying the key factors influencing the VN-Index and quantifying their impacts, this research not only contributes to academic literature but also provides tangible insights for investors, policymakers, and the broader financial community. The use of Python and its associated libraries underscores the practical aspect of this study, demonstrating how technology can be leveraged to enhance financial analysis and decision-making processes.

1.2. Research objectives

The research into factors affecting stock market fluctuations always attracts special attention from investors and society. Therefore, studies on the stock market are predominant in developed countries, and analyses of the relationship or impact of certain factors on the stock market have been conducted using statistical methods, yet the results remain uncertain and contradictory across these studies. Hence, the objective of this study is to compare and find the most efficient machine learning model for predicting the price movements of the Vietnamese stock market based on the VN-Index. Moreover, the author aims to provide the clearest results to answer the question of which macroeconomic factors have a strong impact on the Vietnamese stock market.

1.3 Research scale and methodology

The index used to represent the Vietnamese stock market is the VNI. The data utilized in this study consists of daily records, spanning from March 2016 to March 2024, published on the website: https://finance.vietstock.vn/. The dependent variable is the VNI, and the independent variables include: CPI, exchange rate (VND/USD), oil prices, the S&P500 index, gold prices, money supply M2, and interest rates. To analyze and process the data, the Python programming tool is employed due to its powerful capability in handling and analyzing large datasets, as well as supporting the execution of complex machine learning methodologies. The machine learning techniques applied in this study include: Decision Tree Regression, Random Forest, XGBoost, KNN, ANN, SVM, and Linear Regression. These techniques were chosen based on their ability to process the type of data and the objectives of the study, from simple to complex, to ensure that the prediction results are accurate and reliable. The performance metrics used to indicate the models' efficiency in this study are RMSE, MSE, MAPE, K-Fold cross-validation, and Learning curves. These metrics help assess the quality and reliability of the prediction models and allow for a fair comparison of their effectiveness. Lastly, to estimate the impact level of each attribute on the VN-Index, the SHAP method is applied to each model, providing insights into the significance of each independent variable.

1.4. Research scientific and practical contribution

For the scientific contributions, in terms of theoretical development, this study expands the understanding of the stock market by integrating advanced machine learning techniques. The application of models such as Decision Tree, Random

Forest, XGBoost, and other machine learning models provides new insights into how macroeconomic factors and market elements influence the VN-Index. This helps to further elaborate the theory on the relationship between macroeconomics and the stock market. Simultaneously, the use of machine learning techniques to analyze both macroeconomic news represents a significant advancement in methodology. This study introduces a new approach, allowing for detailed and long-term analysis by utilizing daily data from March 2016 to March 2024. By employing Python and its supportive libraries, the research has provided a powerful and flexible method for analyzing and predicting stock market behavior.

Regarding the practical contributions, the outcomes of this study can help investors and fund managers gain a deeper understanding of the factors influencing the VN-Index, thereby making more informed investment decisions. High-performance predictive models provide valuable information for planning and market forecasting. By estimating the impact of each attribute (such as CPI, exchange rates, oil prices, etc.) on the VN-Index using the SHAP method, investors can adjust their investment strategies to accurately reflect market risks and opportunities. This research contributes to a broader understanding of the financial market in Vietnam, offering new insights into market dynamics to stakeholders - such as regulatory bodies, investment firms, and educational institutions. This can facilitate the development of stronger financial models, improve legal policies, and enhance the curriculum of academic programs related to finance.

1.5. Report structure

The author's research is divided into six main sections:

- Abstract
- Introduction
- Literatural background
- Data và Methodology
- Results and discussions

- Conclusions and Recommendations

In addition, the research article also has other subsections, specifically as follows:

- Title
- Comments of the instructor
- Acknowledgement
- Table of contents
- List of tables, figures
- List of abbreviations, special characters
- Appendix
- References

2. Literature background

2.1. Literature review on macroeconomic factors

Based on empirical evidence, most studies have demonstrated that oil prices, gold prices, interest rates, inflation, exchange rates, and money supply all impact the stock market (Sujit & Kumar, 2011; Nandha & Singh, 2011; Akbar et al., 2019; Alqattan & Alhaky, 2016; González et al., 2018; Geetha et al., 2011; Lee & Brahmasrene, 2018, 2020). The study by Kieu and Diep (2013) measured the relationship between macroeconomic factors and the volatility of the Vietnamese stock market (through the VN-Index stock prices). Four factors, including the CPI as a measure of inflation, the USD/VND exchange rate, and the M2 money supply were considered. The study found a positive relationship between the VN-Index stock prices and the domestic gold prices and M2 money supply in the long term. Conversely, it showed a negative relationship with inflation, while no relationship was found between the exchange rate and the stock price index. In the short term, the current stock price index was directly proportional to the stock price index of the previous month and inversely proportional to the exchange rate. The study by Nhu Quynh and Huong Linh (2019) measured the impact of six macroeconomic factors, including oil prices, the CPI (representing inflation), M2 money supply, interest rates, exchange rates, and gold prices on the Vietnamese stock market (through VN-Index stock prices) for the period 2000–2018. The results showed that, in the long term, inflation had a positive impact on the VN-Index, and interest rates had a negative impact on this index. Phong and Bach Van (2015) used monthly time series data from January 2001 to December 2013 to study the impact of macroeconomic factors on the Vietnamese stock market index. By using the ARDL model to study the short-term and long-term relationships between variables, the study found that, in both the short and long term, the money supply had a positive impact on the stock market index, while factors such as the exchange rate, loan interest rates, government bond interest rates, and inflation had a negative impact on the stock market index. Additionally, the study proposed several policies and directions for the sustainable development of the Vietnamese stock market.

Oil is considered an essential energy source, serving as an irreplaceable fuel for many industries, and also acts as a key commercial product globally. The fluctuation in oil prices is expected to significantly affect macroeconomic variables such as production costs, investment decisions, inflation, and national income, thereby influencing the stock market. The rise in oil prices can have either positive or negative effects on stock values. On one hand, since oil is a primary energy source for most companies, an increase in oil prices raises operational costs and decreases corporate profits. On the other hand, higher oil prices can reflect better business efficiency and growth in the market, indicating high business confidence which might positively impact the stock market. The study by Trinh and Linh Dan (2020) explored the asymmetric impact of oil price volatility on the Vietnamese stock market in both the short and long term following the 2008 financial crisis. Findings indicate that, in the long term, oil price volatility negatively affects the Vietnamese stock market, with rising oil prices worsening the market condition, while decreasing oil prices improve it. The impact of rising oil prices on the stock market was found to be stronger than that of falling prices, showing the asymmetry of oil price effects on the market in the long run. In the short term, the market tends to react oppositely compared to the long term. However, for countries with economies reliant on oil exports, oil prices may have a positive impact on the stock market index. Degiannakis and colleagues (2018) comprehensively synthesized theoretical and empirical research findings on the relationship between oil prices and stock prices, demonstrating that money supply impacts stock prices in two ways. On one hand, an increase in money supply boosts future inflation expectations and business costs, negatively affecting corporate profits and stock prices. Conversely, an increased money supply also enhances liquidity effects for the financial market and particularly for the stock market.

The quantity of money in circulation during a specified timeframe, known as the money supply, includes not only physical cash and bank deposits accessible on demand but also other assets with high liquidity, which can be swiftly converted into cash and widely used for transactions. The interplay between the money supply and the stock market exhibits a positive linkage, as detailed through the lens of monetary policy frameworks. Friedman and Schwartz (1965) initially elucidated this connection, positing that a surge in the money supply amplifies liquidity and credit availability for equity investors, culminating in elevated stock prices. This perspective was echoed by Nguyen and Nguyen (2013), who concurred with the existence of a direct bond between the money supply and stock market performance. A swell in the money supply notably boosts market liquidity, steered by the repercussions of monetary policy actions. During periods of expansive monetary policy, a drop in the economy's interest rates occurs, diminishing the discount rates applied to stocks, which in turn elevates anticipated stock prices and earnings. On the flip side, restrictive monetary policy phases bring about heightened interest rates, upping the discount rates used in valuation frameworks, thus rendering fixed-income securities more appealing, squeezing stock market liquidity, curtailing the propensity to borrow for investing in stocks, and thereby exerting a dampening effect on corporate earnings and stock price levels. Empirical investigations by Rahman, Sidek, and Tafri (2009); Rozeff (1974) have substantiated the impactful role of monetary policy in fostering stock market return growth. Further analysis by Maysami and Koh (2000) highlighted a conducive correlation between the money supply and the SGX index, underscoring the significant ripple effects of monetary expansion. Within the realm of global macroeconomic discourse, this phenomenon is termed the "beggar-thy-neighbor" effect, initially articulated within the Mundell-Fleming-Dornbusch framework. Obstfeld and Rogoff (1995) demonstrated that an upswing in domestic money supply could buoy not only the domestic but also the international economic landscape, transcending the domestic focus of the Mundell-Fleming-Dornbusch model. The impact of money supply on stock valuations manifests in dual aspects. An upsurge in money supply forecasts heightened inflation in the future and escalates operating expenses, adversely impacting corporate profitability and stock valuations. Conversely, an expansion in money supply bolsters the liquidity effect within the financial and stock markets specifically, fostering a favorable impact on stock valuations. Empirical backing from Mukherjee and Naka (1995) unveiled a positive influence of the Yen/USD exchange rate on the stock market index in Japan. Asprem's (1989) research into the sway of macro variables over stock market indexes across ten European nations disclosed a predominantly positive effect of money supply on stock indexes, accentuating the pivotal role of liquidity dynamics in the nexus between money supply and stock values.

According to Cecchetti, Schoenholtz, and Fackler (2006), the exchange rate is defined as the ratio for converting one country's currency into another's. This index significantly influences the trade balance and payments, thereby affecting the national output, employment creation, and overall economic stability. Exchange rates also directly impact the business operations of companies. In Vietnam, research by Dinh and Nguyen (2008) indicated that exchange rate volatility has adverse effects on the stock market. An increase in the exchange rate (VND/USD) signifies the appreciation of USD against VND, allowing more VND to be exchanged per USD, thereby generating profits from currency conversion from USD to VND, enabling investors to earn higher returns than investing in the stock market. However, other studies have shown that the impact of exchange rates on stock markets can be either positive or negative depending on the country. Ho and Huang (2015) pointed out that exchange rates affect the stock markets in Brazil, India, and Russia but not in China. Abdalla and Murinde (1997) discovered a causal relationship between exchange rates and the stock markets in India, South Korea, and Pakistan but not in the Philippines. Using the VAR model, Rjoub (2012) demonstrated that exchange rates negatively affect the US stock market and positively affect the Turkish stock market during the period 2001-2009.

Interest rates represent the cost that borrowers incur to gain access to capital (Devereux & Yetman, 2002), and are determined by the supply and demand for capital usage. As the demand for capital increases, interest rates rise accordingly and vice versa. From a business perspective, borrowing for working capital or expenditures increases debt costs, which in turn reduces corporate profits and shareholder dividends, potentially leading to a decrease in stock prices. Furthermore, higher interest rates diminish the value of future dividend incomes, making stocks less attractive to investors. Several studies have demonstrated an inverse relationship between interest rates and stock prices. Alam and Uddin (2009) concluded that stock prices in most countries are negatively affected by interest rates, a finding that was also supported by Adam and Tweneboah (2008).

Gold serves as a medium for storing value. Unique among assets, it offers the potential for high liquidity. Gold price fluctuations impact almost all economies, including the stock market. The underlying explanation for this correlation is investors diverting capital to the gold market from the stock market when gold prices increase, attracted by the higher yield potential of gold. This shift reduces the demand for stocks, leading to lower stock prices. Truong (2014) articulated that the volatility in gold prices inversely correlates with stock returns. Numerically, a 1% rise or fall in gold prices results in a 0.72% fall or rise in stock returns, respectively. Utilizing the VAR model, Akbar et al. (2019) also proved the bidirectional and inverse impact between gold prices and stock prices. Beyond economic indicators, research has shown that the Vietnamese stock market is influenced by global economic dynamics and commodity prices. Macroeconomic news updates from the US have significantly impacted the Vietnamese stock market, as demonstrated by Mai NC (2016). Hussainey K, Khanh Ngoc L (2009) identified a robust positive correlation between Vietnamese stock prices and the US industrial sector and money market, as well as a positive relationship between the VN Index and the S&P 500 according to Tien NH (2021). In terms of the relationship between commodity prices and the Vietnamese stock market, Tien NH (2021), and Nguyen TN, Nguyen DT, Nguyen VN (2020) observed that Vietnamese stock prices have a direct relationship with oil prices, while the VN Index exhibits an inverse relationship with gold prices.

Other global factors related to the world financial markets, particularly those emanating from the world's largest economy, the United States, exert a profound impact on stock markets around the globe, especially affecting emerging and developing economies. The S&P 500 index is widely viewed as the premier metric for large-cap U.S. stocks, capturing the operational success of the leading 500 firms and representing about 80% of the existing market capitalization. Firstly, the contagion effect of the global stock market must be acknowledged. Mollah and colleagues (2016) explored the daily indices of MSCI (Morgan Stanley Capital International) across 55 stock markets using the DCC-GARCH model (Dynamic Conditional Correlation GARCH) and identified that the U.S. stock market exerts a contagion effect on several global markets during global financial crises and the Eurozone crisis. Naser (2016) uncovered evidence for short-term integration between emerging and developed countries' stock markets through the Dynamic Model Averaging (DMA) method. Sugimoto and associates (2014) employing a diffusion index method, found that African stock markets were most severely impacted by contagion from global stock markets. Richards (2005) suggested that external investors and conditions have a greater influence on emerging Asian markets.

Utilizing asymmetric BEKK and GARCH models (Generalized AutoRegressive Conditional Heteroskedasticity) (Kim et al., 2015; Li & Giles, 2015), significant one-way shock transmission and volatility from the U.S. market to emerging Asian markets were demonstrated. Moreover, the future outlook of the world's largest economy, like the United States, significantly impacts the stock markets in developing nations. Specifically, as the U.S. economy shows robust recovery and U.S. interest rates increase, investment flows may revert to the U.S., adversely affecting the stock markets in developing countries.

Inflation, defined as the ongoing increase in the general price level within an economy over a certain period (Le & Dang, 2017), has been linked to stock market indices in a manner suggesting direct correlation by Adam and Tweneboah (2008). They posited that the market efficiently reallocates resources by mirroring general price increases over the long term. Contrary to this finding, a majority of studies indicate an inverse relationship between inflation and stock market performance, as evidenced by research from Subhani, Gul, and Osman (2010); Geetha, Mohidin, Chandran, and Chong (2011); Mousa, Safi, Hasoneh, and Mohammad (2012). High inflation devalues currency, prompting individuals to invest in tangible assets like gold or real estate over liquid assets such as cash or financial securities, causing significant societal capital to lie dormant as inactive assets. Additionally, inflation escalation increases operational costs for businesses, driving up product prices and reducing consumer demand, which in turn diminishes corporate earnings. This scenario renders businesses less appealing to stock market investors, potentially leading to a "herd mentality" and extensive capital pullout from the stock market. It is clear that excessively high inflation negatively impacts both the broader economy and the stock market specifically.

2.2. Literature Review on Machine Learning to identify macroeconomic factors

In the field of macroeconomics, machine learning is increasingly deployed for classification challenges, such as predicting the movements of the stock market and company failures. Employing machine learning to detect macroeconomic factors impacting the stock market is an innovative strategy that is still not widely adopted in scholarly research. Several significant studies have been undertaken to forecast financial risks. Alessi and Detken (2018) devised an alert system using the random forest algorithm to pinpoint systemic risks from a collection of data on EU banking crises, with macroeconomic indicators as the forecasting variables. Their results validated the superior forecasting ability of the random forest model, indicating its potential for macroeconomic prediction. Beutel and colleagues (2019) evaluated the out-of-sample forecasting efficacy of different early warning models for systemic banking crises across developed economies, discovering that machine learning approaches not only demonstrate a high level of in-sample accuracy but also surpass traditional logit models in out-of-sample predictions. Chatzis and his team (2018) employed various machine learning algorithms to forecast economic risks in 39 nations, proving that deep neural networks markedly enhance classification precision and offer a potent approach for developing a more efficient and risk-sensitive global systemic early warning mechanism compared to conventional methods.

Innovative findings have emerged from studies investigating the use of machine learning for forecasting stock price directions. Gu and colleagues (2020) explored the adoption of machine learning techniques for empirical asset pricing. Their research demonstrated significant benefits for investors from machine learning forecasts. In certain cases, these forecasts could outperform established regression-based strategies documented in existing literature. The authors pinpointed decision trees and neural networks as the most efficient approaches, thanks to their exceptional predictive power by capturing complex non-linear interactions among predictors, which are often overlooked by other methods. Moreover, the research identified a consensus across all methodologies regarding a relatively small set of dominant predictive indicators. The most impactful predictive factors were associated with price-related elements, such as profit reversals and momentum, whereas stock liquidity measures, stock volatility, and valuation ratios were highlighted as the next most powerful predictors in asset pricing context. Cakici et al. (2023) delved into the subject of equity anomalies predicting market risk compensation. The study analyzed up-to-date data from the U.S. and global markets using diverse machine learning techniques and covered 42 countries from January 1990 to December 2021. The researchers concluded that typically, anomalies cannot predict overall market earnings. However, this finding was only applicable to the U.S. and lacked external validity in two dimensions: it could not be generalized to an international setting, and it was not accurate across different anomaly sets, regardless of the chosen and designed factor strategies. The study indicated that any predictive capacity was confined to a select few specific anomalies and heavily influenced by minor methodological choices. Dong and associates (2022) sought to examine the link between long-term abnormal portfolio returns and the predictability of superior market returns over time. By analyzing 100 representative anomalies from literature and employing various reduction techniques, such as machine learning, forecast combination, and dimensionality reduction, they provided the first systematic proof of this correlation. The study concluded significant out-of-sample predictive capacity for long-term abnormal portfolio returns regarding superior market returns, both statistically and economically. The predictive potential, the researchers suggest, arises from the asymmetric constraints on price arbitrage and overpricing adjustments, highlighting the critical role of employing groups of long-term abnormal portfolio returns from cross-sectional literature for predicting market outperformance on an out-of-sample basis, assuming forecasting strategies are protected against overfitting.

Instead of relying on traditional statistical methods commonly used previously, this study employs machine learning techniques to investigate how economic indicators affect the Vietnamese stock market. These techniques not only demonstrate the impact of similar indicators as traditional methods but also quantify the relative importance of these indicators using feature importance techniques that will be described in section 3. Given the real-world scenario often involves multiple interconnected factors, ranking the importance of these factors could significantly contribute to understanding the situation.

3. Data và Methodology

3.1. Research process

After aggregating data from various financial and global websites in phase 1, phase 2 begins with the author evaluating the data's completeness and quality. This task involves checking for missing values, examining for outliers, and assessing data uniformity. The subsequent step in phase 2 provides an overview of the data through performing descriptive statistics and calculating the Pearson correlation coefficient among quantitative variables to evaluate the linear relationship between them. Concluding phase 2, the author visualizes the data to identify which macroeconomic factors have been the strongest drivers of stock prices from 2016 to the present.

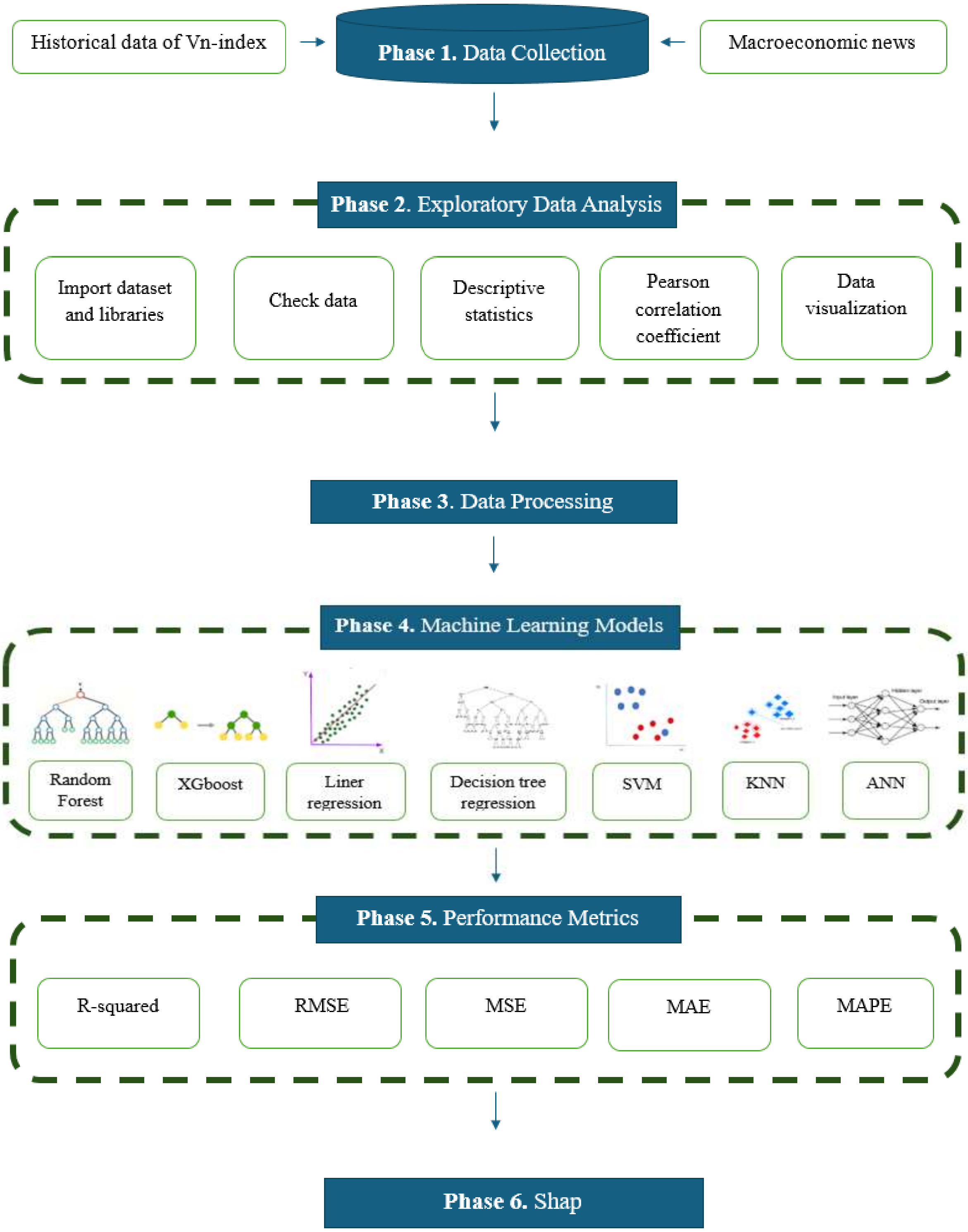

Figure 2. General architecture of approach the purpose of the topic

Source: Author's summary

In phase 3, the researcher segments the data into a training dataset (80%) and a test dataset (20%). Following this division, the researcher constructs 7 machine learning models, employing the training dataset in conjunction with a 10-fold cross-validation method for fine-tuning hyperparameters and systematically evaluating each model's performance across multiple validation sets in phase 4. Moreover, at this stage, the presence of bias or variance in the models is examined, along with their operational effectiveness, through Learning Curves. Next, in phase 5, model performance is assessed by utilizing hyperparameters related to the highest average accuracy (or lowest RMSE) observed during cross-validation, then applying these optimal settings to an independent test set and calculating the RMSE for the best-performing model, serving as a quantitative assessment of the model's prediction accuracy when applied to the test dataset. Finally, in the study's phase 6, SHAP values are applied to interpret the impact of each feature on the model's predictions, offering deeper insight into the relationship between features and the target variable

3.2. Data source and data sampling

This study explores the impact of seven factors on the VNI index, including: global gold prices, global oil prices, M2 money supply volume, interest rates, exchange rates, consumer price index, and the S&P 500 index. By collecting data over a period of nearly ten years, from March 2016 to March 2024, this study provides insights into how these economic and financial factors affect the stock market..

- **VNI stock price index variable**

The VNI index was first launched at HOSE in July 2000, considered the primary index reflecting the movement of the stock market, surpassing other indices like the HNX Index, VNI 30 in terms of market representation. The VN-Index is calculated using the Passcher formula:

$$\mathrm{VN-Index} = \frac{\sum_{i=1}^{n} P_{1i}\ x\ Q_{1i}}{\sum_{i=1}^{n} P_{0i}\ x\ Q_{0i}} x100$$

In which:

- Pt represents the current price of a stock at time t.
- Qt denotes the number of outstanding shares of a stock at time t (also referred to as the floating stock).
- ∑(Pt×Qt) is the summation of the market capitalization of all the listed stocks at the current time t. Market capitalization is calculated by multiplying the stock price with the number of outstanding shares.
- P0 is the base period price of the stock (this is often a historical price set on a specific base date).
- P0×Qt represents the base market capitalization, which is the total market capitalization of all the listed stocks during the base period.

The significance of the VNI lies not just in its market capitalization and longstanding presence but also in its ability to attract a vast number of investors as well as a considerable number of listed companies with significant capital, robust finances, efficient business operations, and adherence to disclosure regulations. The VNI Index is recorded in points, based on the last trading session's closing prices of each month, with data sourced from the Ho Chi Minh City Stock Exchange website (https://www.hsx.vn/). Notably, the Ho Chi Minh City Stock Exchange also conducts periodic delisting for companies that no longer meet the standards, enhancing the transparency and effectiveness of the VNI as a representative index of the Vietnamese stock market.

- **Oil prices variable**

In the oil industry, crude oil is differentiated based on geographical origin (such as WTI from West Texas or Brent from the North Sea) along with physical characteristics like weight and viscosity, categorized into light, medium, and heavy types. Moreover, crude oil is classified as "sweet" if it contains less than 0.5% sulfur, or "sour" when it has about 1% sulfur or more, requiring more refining processes to meet desired standards because sulfur, when burned, produces sulfur oxides, a major pollution source. Particularly, heavy oil tends to have a higher sulfur content. A barrel of oil, standardized as 42 U.S. gallons (equivalent to 158.987 liters), is the conventional unit in crude oil transactions. For commercial purposes, the global oil price is monitored through the monthly average WTI crude oil price, listed on the New York Stock Exchange (NYSE). Data on oil prices are gathered from reputable online resources (https://www.investing.com/).

- **Money supply M2 variable**

Money supply in the economy refers to the total amount of circulating money, including cash and various types of deposits, aimed at meeting the liquidity and reserve requirements of market participants. A balance between money supply and demand is crucial to maintaining economic stability. However, discrepancies between supply and demand can lead to undesired economic consequences. Factors such as financial market development and differences between countries can impact the composition and measurement of money supply, causing it to vary over time. In this context, tools like traveler's checks and new types of deposits have been introduced by the U.S. Federal Reserve (Fed) to classify under the categories M1 and M2. Specifically, M2 includes longer-term deposits and certain financial products not included in M1, such as money market accounts and mutual fund shares (excluding institutional). In Vietnam, M2 also encompasses bills issued by commercial banks. Identifying an appropriate measure of money supply is important for accurately forecasting economic indicators such as inflation and business cycles. M1 and M2 are commonly used indicators for measuring money supply, and Vietnam has chosen to use M2 as a standard measure. Data on money supply is collected from the official website of Vietnam Securities (https://finance.vietstock.vn/).

- **Interest rate variable**

Interest rates represent the fee that borrowers must pay on the capital they borrow and play a central role in monetary policy adjustments. They influence investment decisions, consumption levels, and unemployment rates. Central and reserve banks typically lower interest rates to stimulate investment and consumption within the economy. However, maintaining low interest rates as part of macroeconomic policy can lead to risks, including the potential for economic bubbles, particularly in real estate and stock markets. The interest rate data for this study is based on the refinancing rate, adjusted and announced by the State Bank of Vietnam through their official website (https://www.sbv.gov.vn/).

- **Exchange rate variable**

The exchange rate values one currency unit against another country's currency unit. It represents the relative value of one currency to another. In international transactions, the US dollar (USD) is often preferred as the main currency for payments due to its widespread acceptance and reliability in global business. Data on exchange rates are sourced from the official website of the State Bank of Vietnam, focusing on the exchange rate between USD and VND, reflecting the activities and fluctuations in the foreign exchange market (https://www.sbv.gov.vn/)

- **Consumer price index Variable**

This index measures price increases across the entire economy, indicating a decrease in currency value over time and reduced purchasing power. Inflation, defined as the general rise in prices of goods and services, diminishes the buying power of money. This can be contrasted with other economies, where inflation reflects the degree of currency depreciation against other currencies on the international market. Within a country, inflation is understood as the decrease in purchasing power of that nation's currency, while globally, it measures depreciation against other currencies. The distinction between these two perspectives remains a topic of discussion among economists. The opposite of inflation is deflation. An inflation rate at 0 or a small positive number signifies price stability. Data on the Consumer Price Index are derived from the monthly percentage change in CPI compared to the previous month and are provided from the official website of Vietnam Securities (https://finance.vietstock.vn/).

- **Gold price Variable**

Gold is one of the most valued precious metals, measured in troy weight units and often expressed in grams. When gold is alloyed with other metals, the term "karat" is used to indicate the proportion of gold in the alloy, with 24 karats representing 100% gold. The purity of gold can also be expressed through a decimal system ranging from 0 to 1, where a higher value denotes greater purity, for example, 0.995 indicates a high level of purity. Gold prices are established through trading on gold and derivatives markets. Historically, the London gold fixing process starting in September 1919 laid the foundation for an international gold price standard, complemented by another pricing method in 1968 to serve the US market. As global gold prices rise, the price of gold in Vietnam typically increases as well, since the Vietnamese gold market is not entirely isolated from the global gold market. International gold price movements also have a significant impact on XAU/VND. However, the extent and speed at which gold prices rise in Vietnam may vary due to factors such as exchange rates, taxes, domestic supply and demand. Data on global gold prices are sourced from a financial website (https://finance.yahoo.com/).

- **S&P 500 Variable**

The S&P 500 index is widely regarded as a primary gauge for the performance of the U.S. stock market, consisting of around 500 of the largest companies based in the United States. The index is maintained by S&P Dow Jones Indices and its components are selected by a committee. It's associated with multiple ticker symbols such as ^GSPC, INX, and $SPX, depending on the market or website. The index also serves as a component in calculating the Conference Board Leading Economic Index, which is used to forecast the direction of the U.S. economy. For companies to be eligible for inclusion in the S&P 500, they must meet certain criteria regarding market capitalization, liquidity, trading volume, and must be listed on major U.S. exchanges, among other requirements. If the S&P 500 experiences a significant change, either an upturn or downturn, it may affect global market sentiment. A positive performance might boost investor confidence worldwide, potentially leading to increased investment in markets like Vietnam. Conversely, a downturn can lead to a decrease in global risk appetite, with possible negative impacts on the VN-Index. S&P 500 index data source is taken from the S&P Dow Jones Indices website (https://tradingeconomics.com/).

3.3. Model specification.

- **Target Variable**

The target variable of the machine learning model in this study is **'Closing_price'**, which represents the closing price of the security. The main aim is to predict the exact closing price based on the defined input variables.

- **Features**

Table 1. Data Description

Features	Description
Oil price	International crude oil price.
CPI	Consumer price index (which measures inflation) of Vietnam.

Exchange rate	Exchange rate of Vietnamese Dongs to US Dollars.
S&P 500	S&P 500 index, representing the performance of the US stock market..
XAU/USD	World gold price
Money supply M2	The money supply in the Vietnamese economy is categorized into the M2 classification
Interest rate	Base interest rate of the State Bank of Vietnam

Source: Author's summary

3.4. Data preprocessing

In the data preprocessing phase for machine learning models, the first step undertaken by the author was the removal of unnecessary columns such as 'Date' and 'Closing_price' from the dataset. The 'Date' column is typically excluded in most machine learning prediction models as it does not directly contribute to price predictions, whereas 'Closing_price' is the target variable to be predicted, therefore, it is removed from the input data set. Subsequently, the dataset is divided into training and testing sets using the train_test_split function, with test_size=0.2 indicating that 20% of the data is allocated for the testing set and the remaining 80% for the training set. The random_state=42 parameter ensures consistency and reproducibility in data splitting across different runs, making the results replicable.

After preparing and splitting the data, machine learning models such as RandomForestRegressor,LinearRegression,DecisionTreeRegressor,KneighborsReg ressor, MLPRegressor, SVR, and XGBRegressor can be trained on the training dataset and then evaluated on the testing dataset using metrics like mean_squared_error, mean_absolute_error, and r2_score. Each model employs its approach to learn from the input data to predict the closing price, and comparing their performance on the testing set helps determine the most suitable model for this specific problem.

3.5. Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) constitutes a crucial phase in Chapter 3. This process involves a thorough examination and summarization of datasets with the aim of uncovering fundamental understandings, detecting patterns, relationships, and outliers, as well as identifying anomalies within the data. EDA serves as the foundational step in data analysis, providing deeper insights into the structure and distribution of data. It also helps in identifying outliers and verifying the dataset for any missing or inconsistent entries.

During this process, the author also employs correlation analysis to discover relationships among variables. This method involves identifying correlations between two variables using a correlation matrix. The correlation matrix is a table that displays the correlation coefficients between two variables in the dataset. Each cell in the table represents the correlation between two variables. The higher the coefficient, the stronger the relationship between the variables. In this study, the author constructs the correlation matrix using the corr() function from the pandas library.

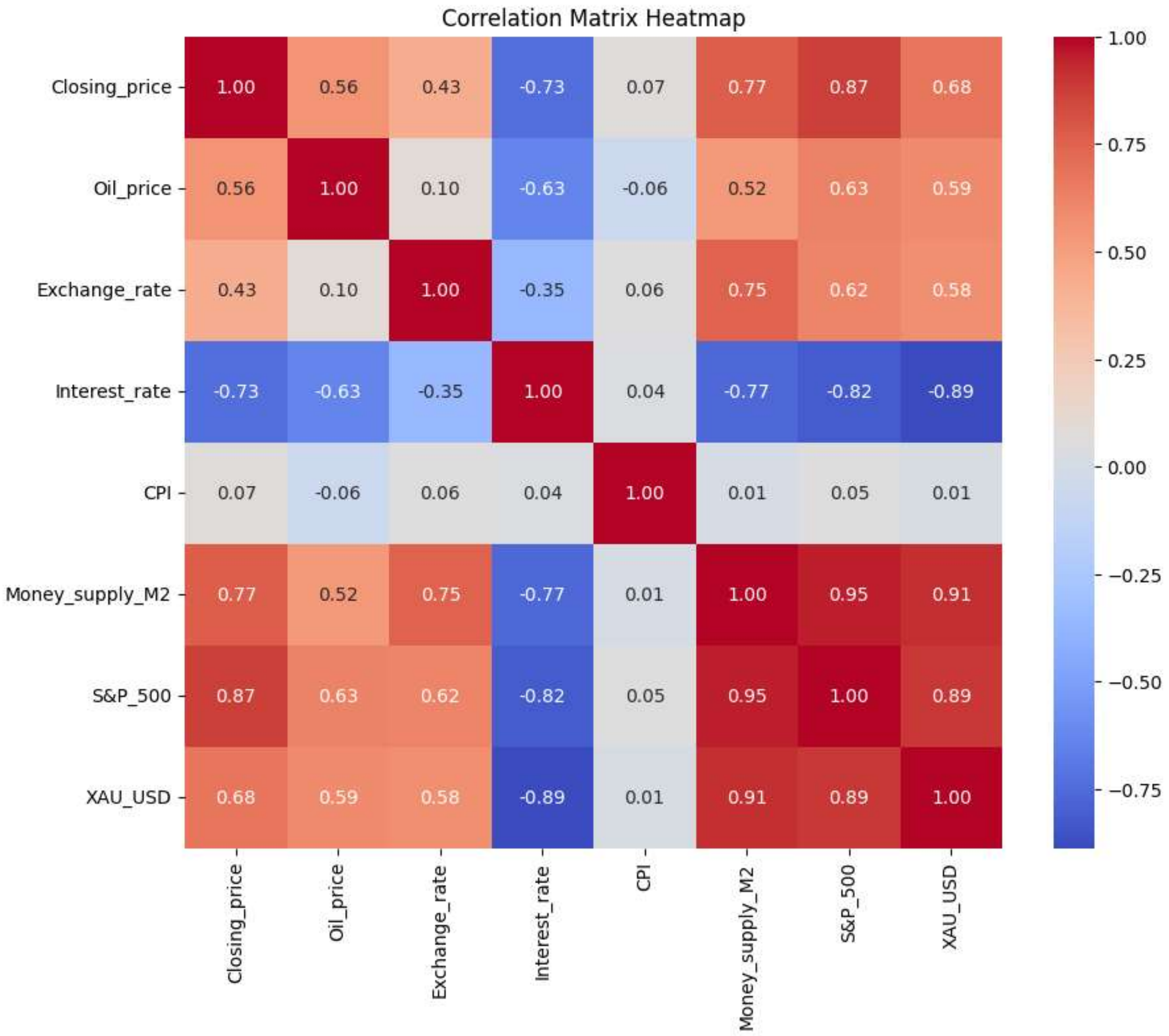

Figure 3. Example of correlation matrix between variables

Source: Author's summary

After the data preprocessing, the author proceeds with a descriptive analysis to outline the fundamental characteristics of the dataset. This descriptive examination serves to portray the dataset by summarizing essential information into basic measures without extending to conclusions or inferences beyond the provided data

Table 2. Example of descriptive statistics

	Closing_price	**Oil_price**	**Exchange_rate**	**Interest_rate**	**CPI**	**Money_supply_M2**	**S&P_500**	**XAU_USD**
count	2002.000000	2002.000000	2002.000000	2002.000000	2002.000000	2.002000e+03	2002.000000	2002.000000
mean	1008.562133	65.705604	23095.115897	0.053988	0.264266	1.089839e+07	3323.271973	1564.271224
std	228.499609	21.026712	612.182465	0.010220	0.440917	2.782728e+06	838.329327	280.207245
min	555.820000	7.790000	21779.000000	0.040000	-1.540000	6.267958e+06	2000.540000	1128.430000
25%	856.285000	55.670000	22727.500000	0.044000	0.010000	8.521098e+06	2642.247500	1284.822500
50%	988.315000	66.820000	23171.000000	0.060000	0.250000	1.075802e+07	3099.015000	1552.695000
75%	1166.967500	77.655000	23321.574707	0.062500	0.510000	1.376102e+07	4129.542500	1824.365000
max	1528.570000	128.260000	24871.000000	0.065000	1.520000	1.535412e+07	5088.800000	2063.810000

Source: Author's summary

The descriptive statistics table offers an overview of data related to the stock market and various macroeconomic factors. The data summary includes the number of observations, mean values, standard deviations, and the range of values from minimum to maximum, as well as significant percentiles (25%, 50%, 75%). Each variable, from closing stock prices to oil prices, exchange rates, interest rates, the Consumer Price Index (CPI), money supply M2, the S&P 500 index, and gold prices (XAU/USD), is analyzed across 2002 observations. The variability in the variables is reflected through the standard deviation values, with closing stock prices and oil prices showing significant disparities among observations, indicating the diversity in market reactions and commodity pricing. The fluctuations in exchange rates and M2 money supply also mirror the volatility in monetary policy and its economic impacts. The distribution of CPI values, ranging from negative to positive, reveals changes in living costs, while the disparity in values of the S&P 500 index and gold prices highlights their significance as indicators of the global financial market. This representation through minimum, percentile, and maximum values allows us to better understand the dispersion and range of each variable, aiding in the identification of trends and underlying patterns that may affect the stock market.

Subsequently, the author turns to visualization tools, aiming not only to identify trends and relationships with the target variable but also to detect anomalies and outliers within the data, thereby making more accurate assumptions for subsequent analyses. Visualization provides a comprehensive view of the data, allowing for a clear understanding of the impact of macroeconomic variables on the stock market over time, and supports the process of making data-based decisions.

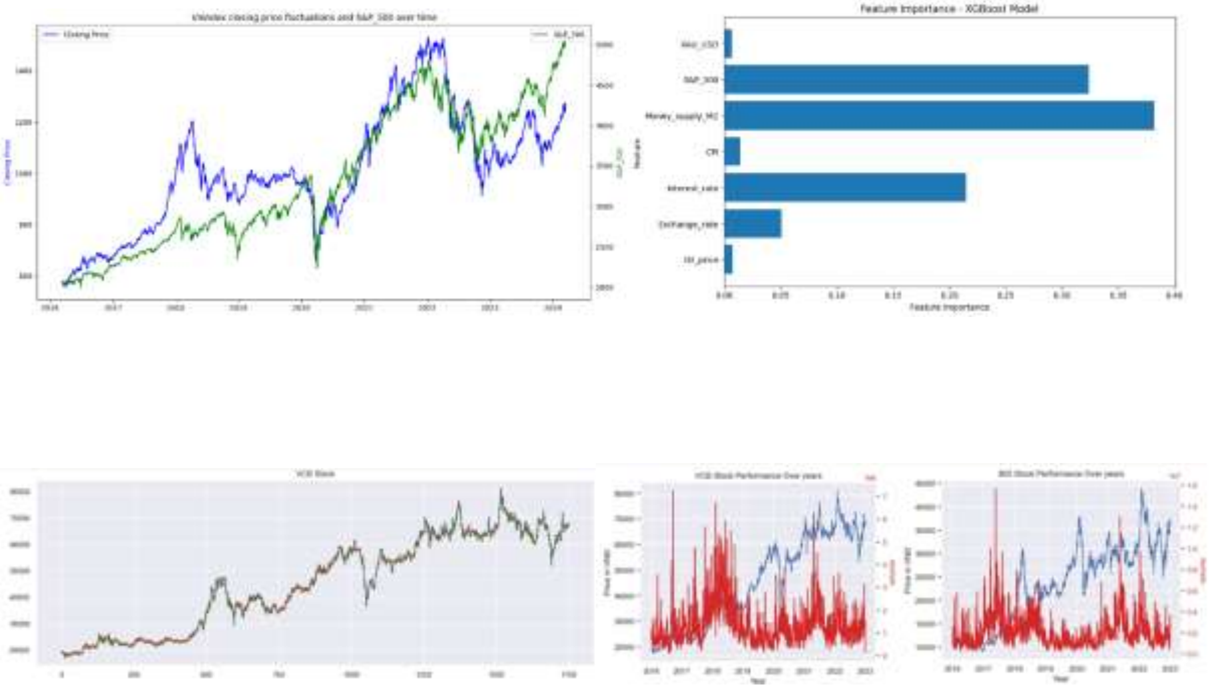

Figure 4. A few example of data visualization

Source: Author's summary

The author visualizes the data to highlight relationships and patterns among macroeconomic variables, providing a deeper understanding of the data through graphical representations. By employing charts such as line graphs to illustrate price trends, box plots to reflect the volatility and distribution of the exchange rate, scatter plots to demonstrate the relationship between interest rates and the consumer price index, and heat maps to summarize the correlation between the S&P 500 index and M2 money supply, this study offers a visual and comprehensible view of how macroeconomic factors affect the stock market.

3.6. Feature Importance tools

3.6.1. Feature Importance

Feature importance is a technique used in predictive modeling to assign a score to each input feature of the model, indicating the relative importance of each feature when making a prediction. These scores can be used to rank the features in order of importance, from most to least important. It makes complex models more understandable by highlighting which features are most influential in predicting the target variable. This insight can be particularly valuable in domains requiring clear explanations for decisions, such as finance and healthcare.

In the overall analysis, models like ANN and SVM can provide insights into feature importance through more complex techniques, whereas KNN does not directly offer information on its significant features. Therefore, the author chooses to utilize only four models which exhibit better predictive performance compared to the other four models: Random Forest, XGBoost, Linear Regression, and Decision Tree Regression. These selected models are used to identify the most important features based on their consistency across the models

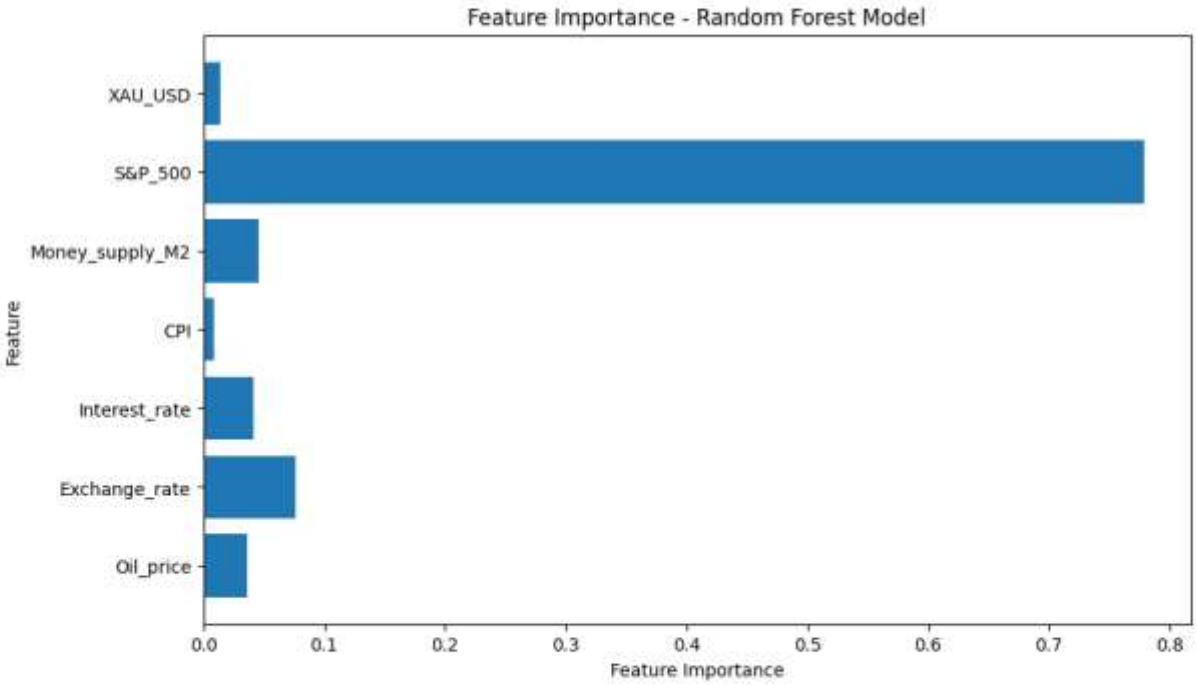

Figure 5. Feature Importance of Random Forest Model

Source: Author's summary

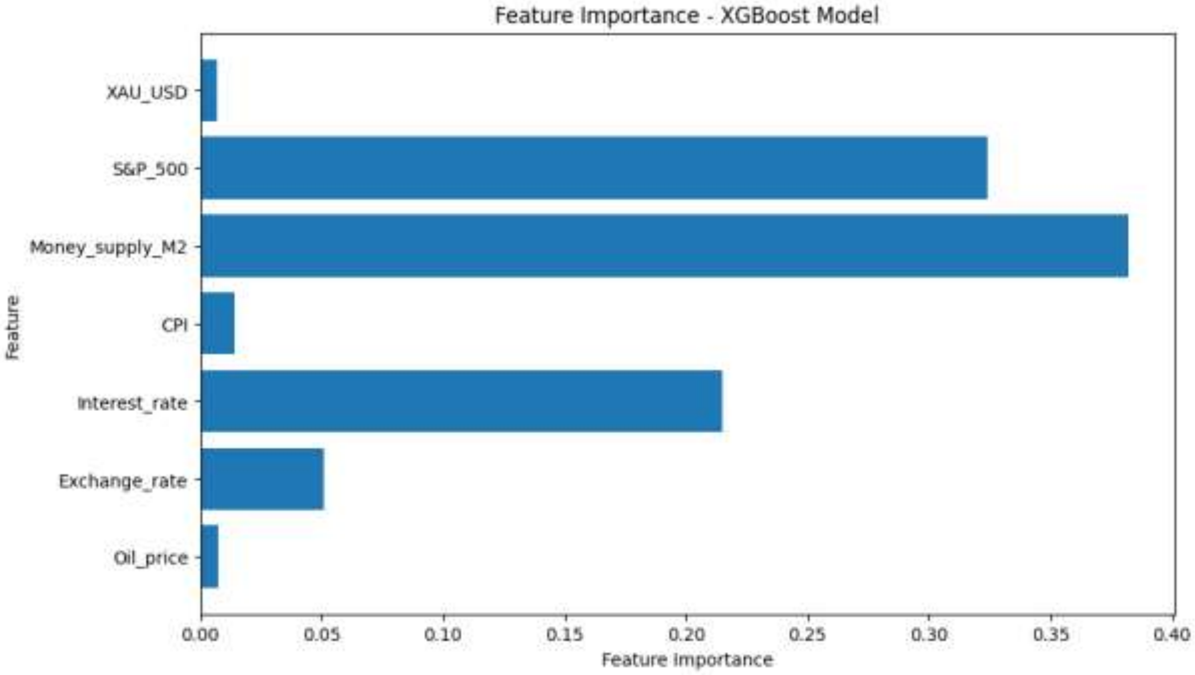

Figure 6. Feature Importance of XGBoost Model

Source: Author's summary

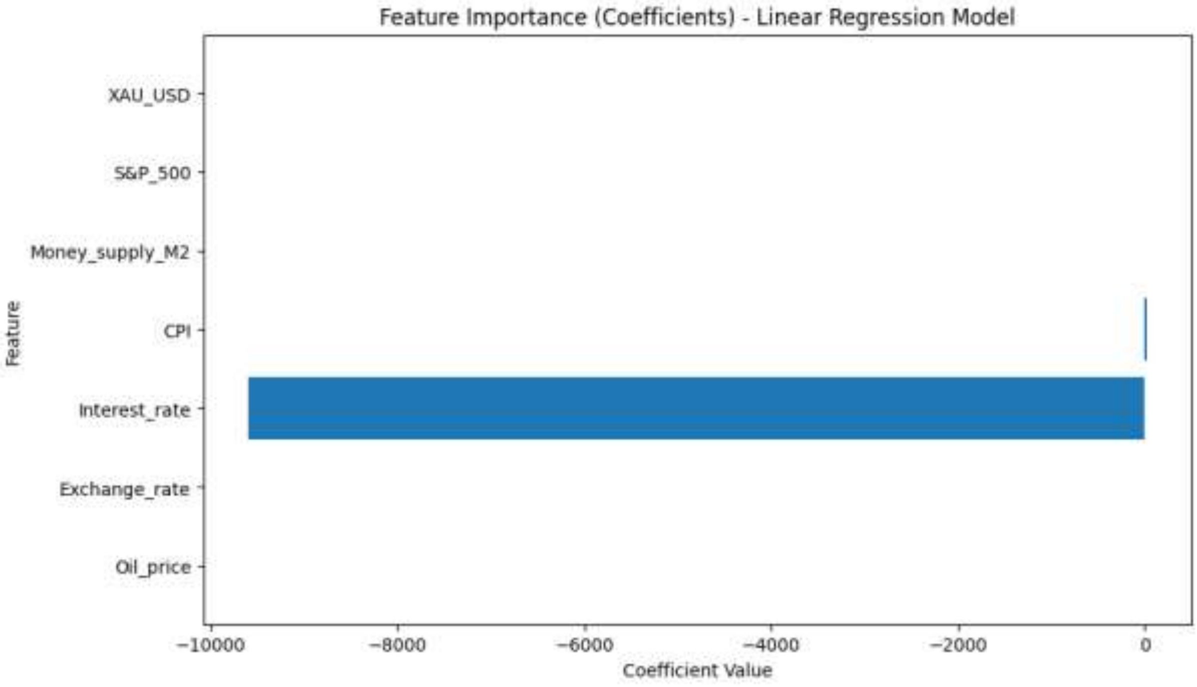

Figure 7. Feature Importance of Liner Regression Model

Source: Author's summary

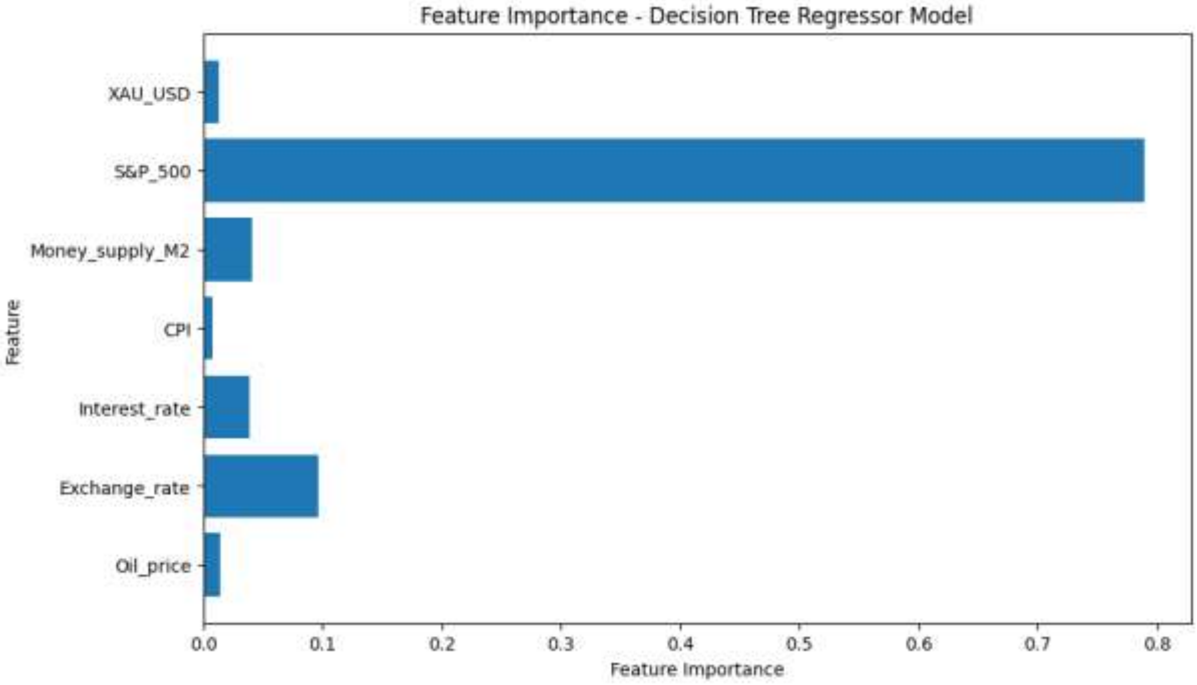

Figure 8. Feature Importance of Decision Tree Regression

Source: Author's summary

First, the S&P_500 feature is the most influential across all four models, indicating a strong correlation between the global stock market and the predicted index. The Money_supply_M2 feature emerges as an second important factor affecting the index, highlighting that monetary policy and the volume of money in circulation significantly impact the economy. Third, the exchange rate also proves to be an essential feature, affecting the VN-index. This reveals the stock market's sensitivity to fluctuations in the currency market. Fourth, While the importance of the interest rate varies among the models, it is identified as a factor impacting the index, especially in the Linear Regression model, which shows a large negative coefficient. This indicates an inverse relationship between interest rates and stock prices. Fifth, although the significance of crude oil shows little variation among the models, crude oil prices remain an influential factor, reflecting its importance in the global economy. Moreover, XAU_USD consistently holds lower importance and does not appear in the top five of all models. CPI is not displayed in the provided charts, hence it is not considered in this analysis.

3.6.2. Shap value (Shapley Additive explanations)

SHAP values are a technique in the field of explainable AI (XAI) that provide insights into the contribution of each feature to the prediction of a particular instance by a machine learning model. This method draws from cooperative game theory, particularly from the concept of Shapley values, which were originally developed to fairly distribute the payout among players depending on their contribution to the game. SHAP importance offers important insight about the predictions created in experiments. It can help to understand which features are the most important to the prediction. SHAP values represent how much each feature contributes to the predicted value of the target, given all the other features of that row.

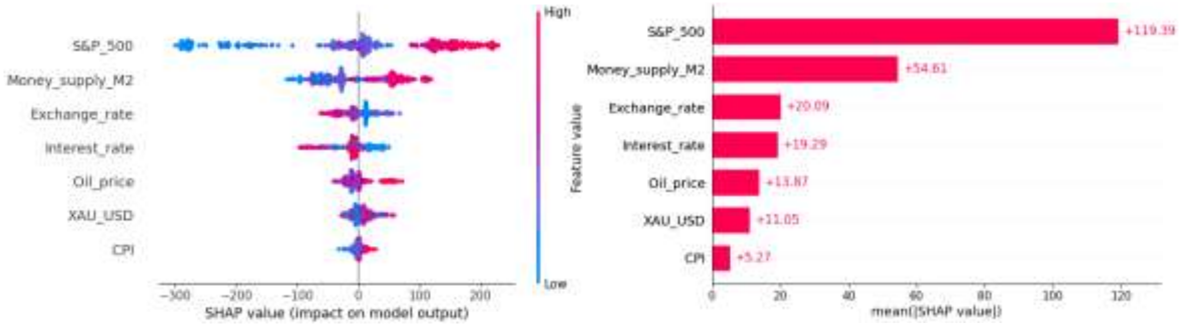

Figure 9. Example of SHAP feature importance

Source: Author's summary

We use the SHAP value to investigate the feature importance in order to analyze which indicators in the dataset have stronger effects on the Vietnam stock index. After comparing the seven models through the SHAP value analysis, we can deduce the importance of different features in predicting the closing price of the VN-index.

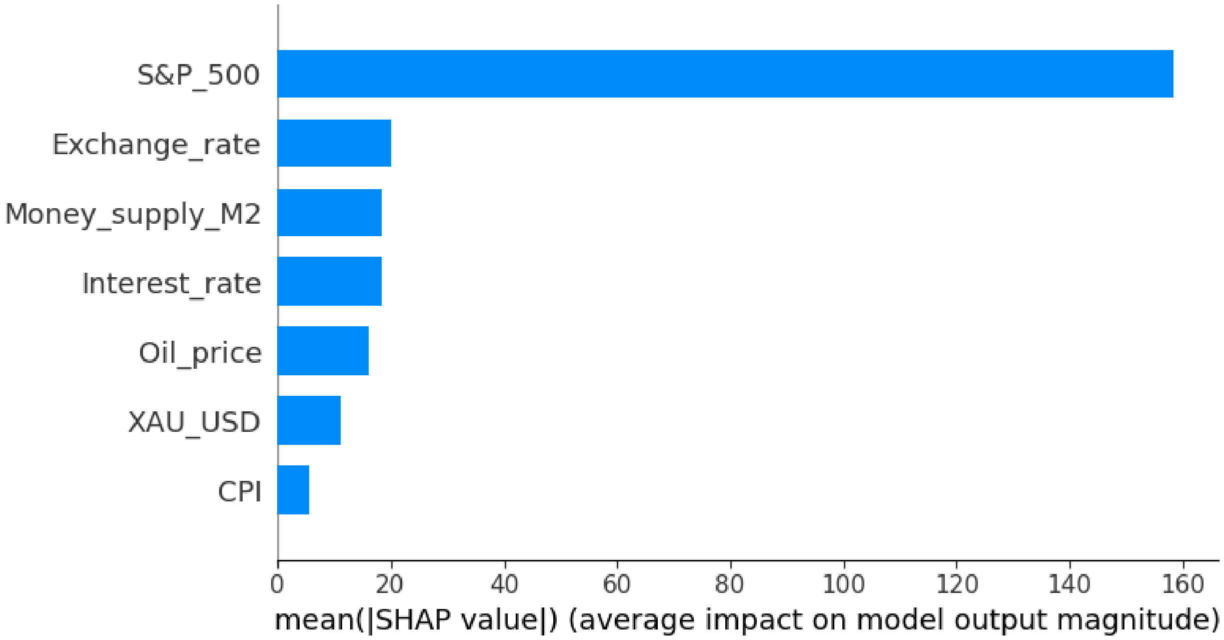

Figure 10 . SHAP feature importance of Random Forest

Source: Author's summary

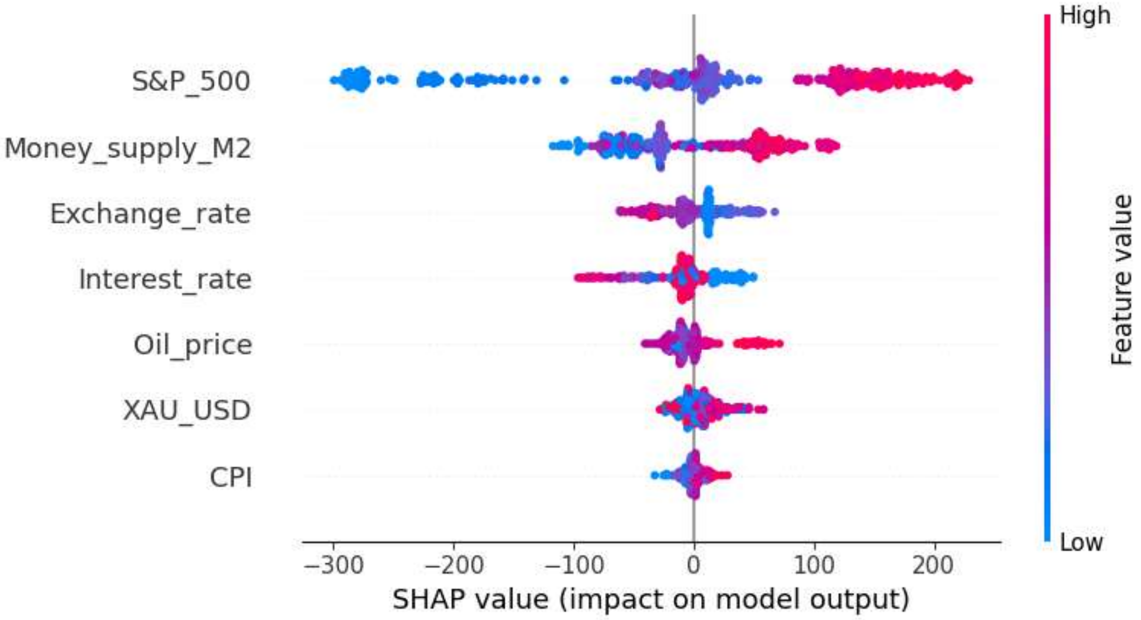

Figure 11 . SHAP feature importance of XG Boost

Source: Author's summary

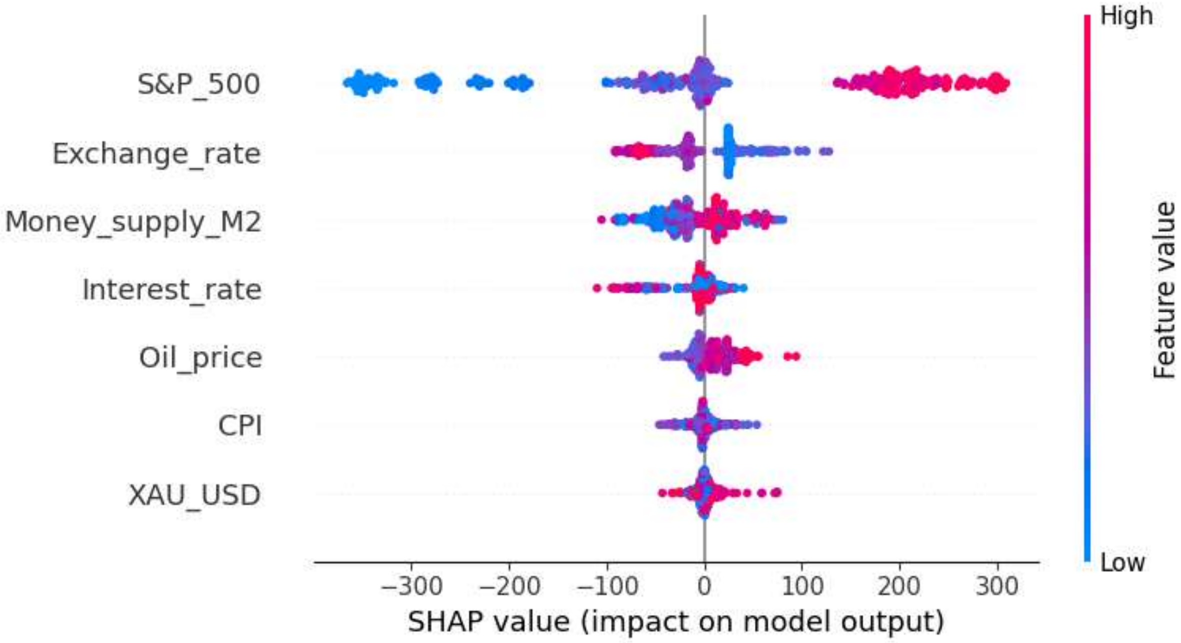

Figure 12 . SHAP feature importance of Decision Tree Regression

Source: Author's summary

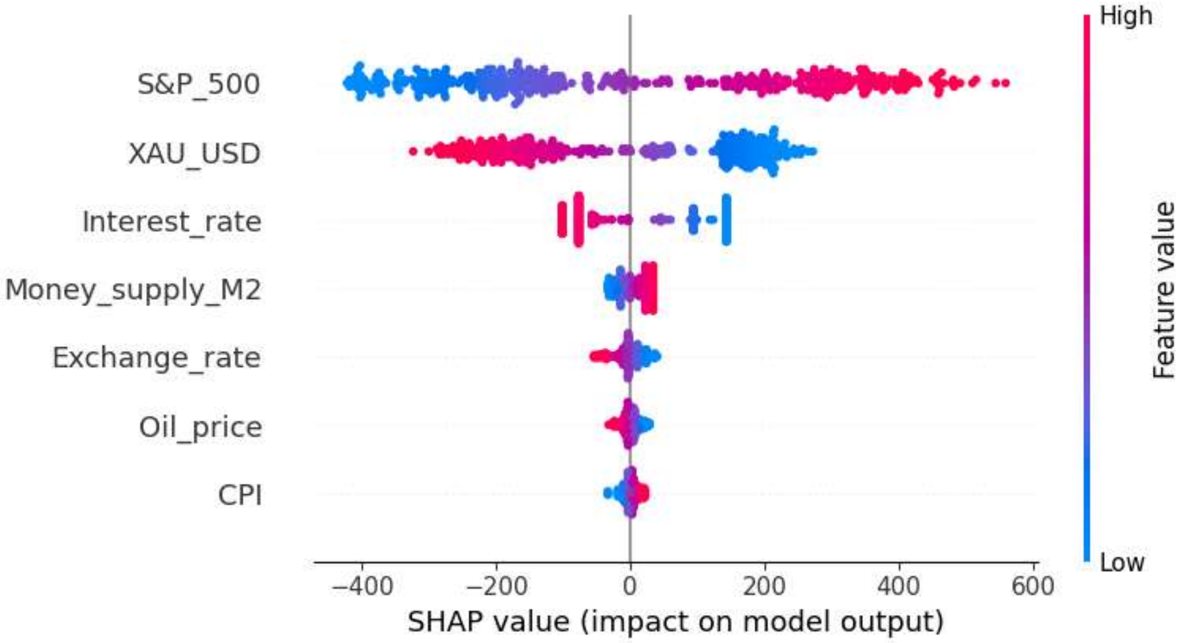

Figure 13 . SHAP feature importance of Liner Regression

Source: Author's summary

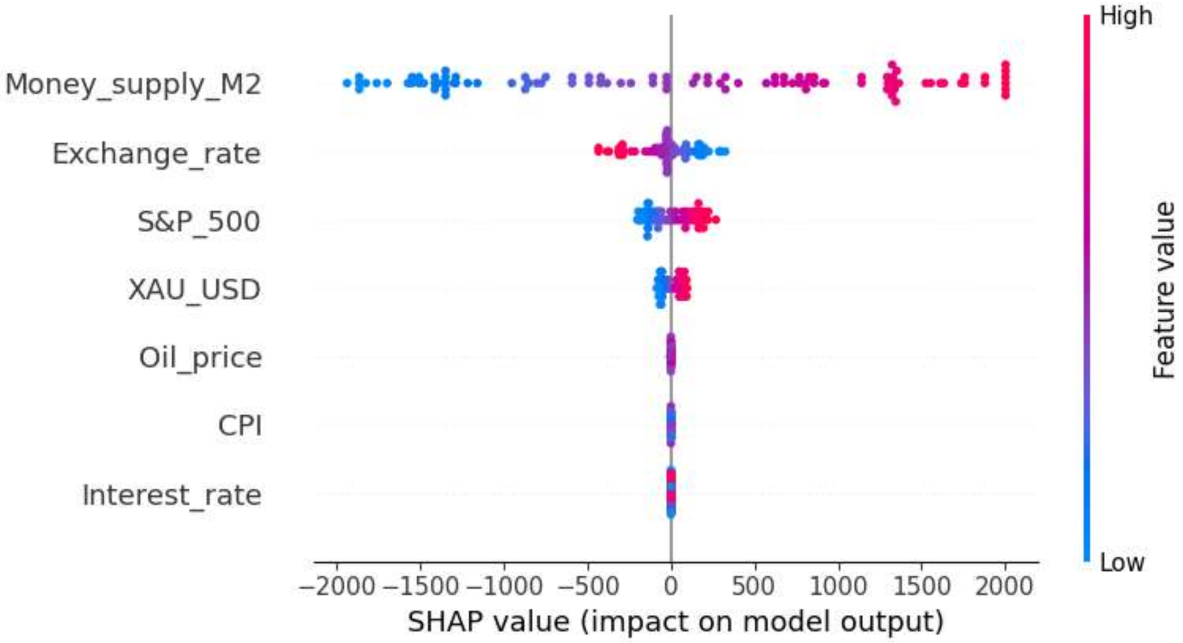

Figure 14 . SHAP feature importance of ANN

Source: Author's summary

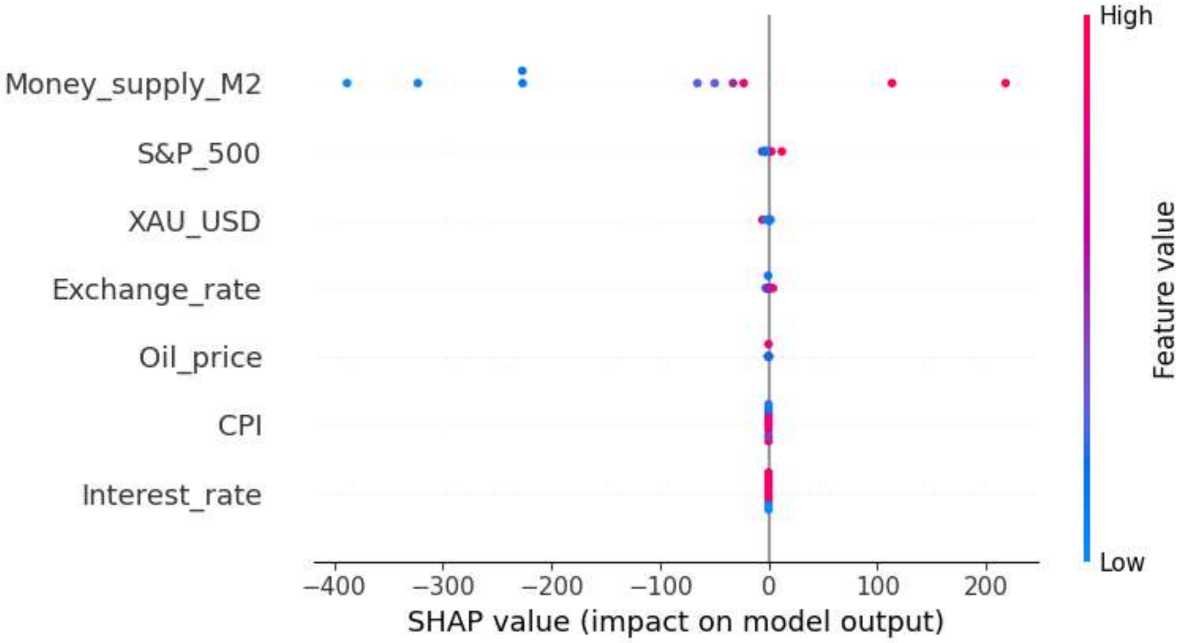

Figure 15. SHAP feature importance of KNN

Source: Author's summary

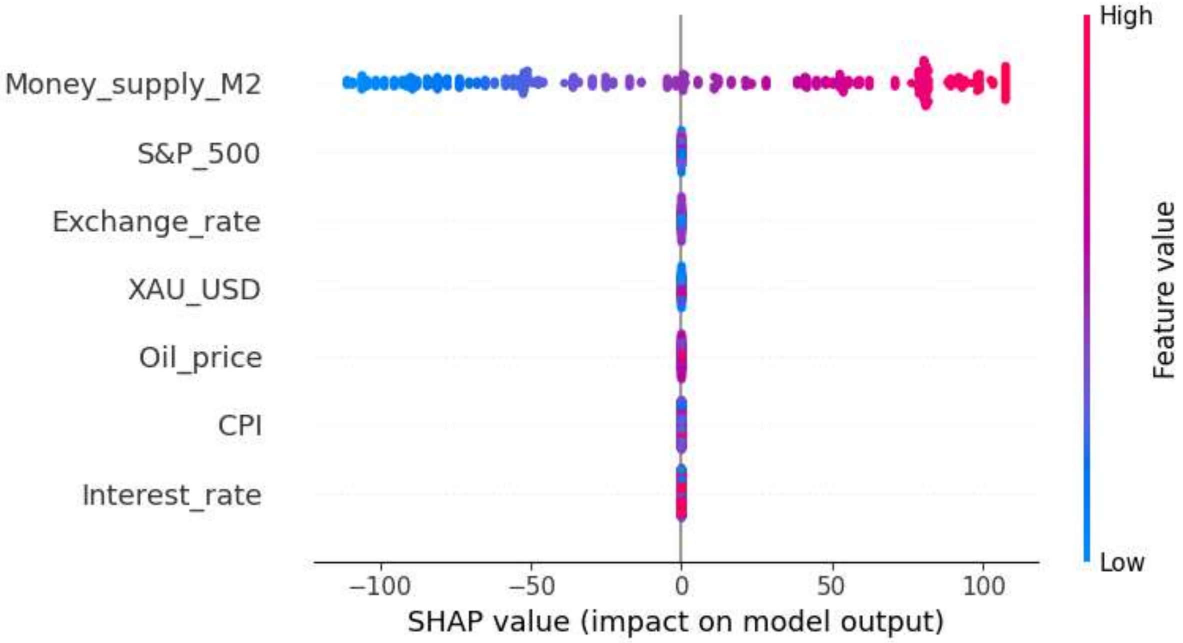

Figure 16. SHAP feature importance of SVM

Source: Author's summary

First, S&P 500 consistently shows the highest impact on the VN-index closing price across most models, indicating a strong correlation between global stock market trends and the VN-index. Second, the Money supply (M2) appears to be a significant factor, suggesting that liquidity in the economy has a notable effect on the stock market. Third, Fluctuations in the Exchange rate seem to have a considerable influence, reflecting the impact of foreign exchange on market confidence and investment decisions. Fourth, Interest rates typically have an inverse relationship with the stock market, and the SHAP values indicate that changes in interest rates can be a predictor of VN-index movement. Fifth, while not as strong as other features, oil prices do have a noticeable impact, possibly due to the economic significance of oil in global markets. Sixth, Gold price is another feature that shows a variable but significant impact on the VN-index, which could be due to its status as a safe-haven asset during market volatility. Seventh, the Consumer Price Index (CPI) has the least impact among the features analyzed, which might suggest that inflation is not as immediate a concern for stock market movements as the other factors.

In conclusion, while the models vary in their specifics, there is a clear trend that global economic indicators S&P 500, domestic monetary policy Money supply M2, and the international trade Exchange_rate factor are influential in predicting the closing price of the VN-index.

3.7. Reasonable choices of Machine learning and deep learning algorithm for this research

3.7.1. Random Forest

The Empirical Evidence shows prior research in financial market prediction has demonstrated the effectiveness of Random Forest in capturing the complex dynamics and nonlinear relationships inherent in stock price movements. Random

Forest is chosen for its robustness and ability to handle overfitting, which is crucial in analyzing financial markets where the data can be highly volatile. Its ensemble approach, by aggregating multiple decision trees, enhances the model's predictive accuracy and stability. Its effectiveness in feature selection makes it ideal for identifying and evaluating the significance of various macroeconomic factors on the VN-Index. Random Forest's ability to manage large datasets with a mixture of numerical and categorical data aligns well with the diverse range of variables considered in this study.

Random Forest operates as a supervised learning technique. This method functions by constructing numerous decision trees from various subsets of the dataset and determining the outcome based on the predominant or mean response of those trees. The essential steps involved in the Random Forest algorithm are depicted in Figure 17.

1. Initially, Random Forest selects “n” distinct subsets of the data, each containing “k” data points, randomly drawn from the total dataset.
2. For each of these subsets, a separate decision tree is developed independently.
3. Output is generated by every individual decision tree.
4. The overall output is then determined by aggregating these individual outputs through a process of majority or mean voting, depending on whether it’s a classification or regression task, respectively.

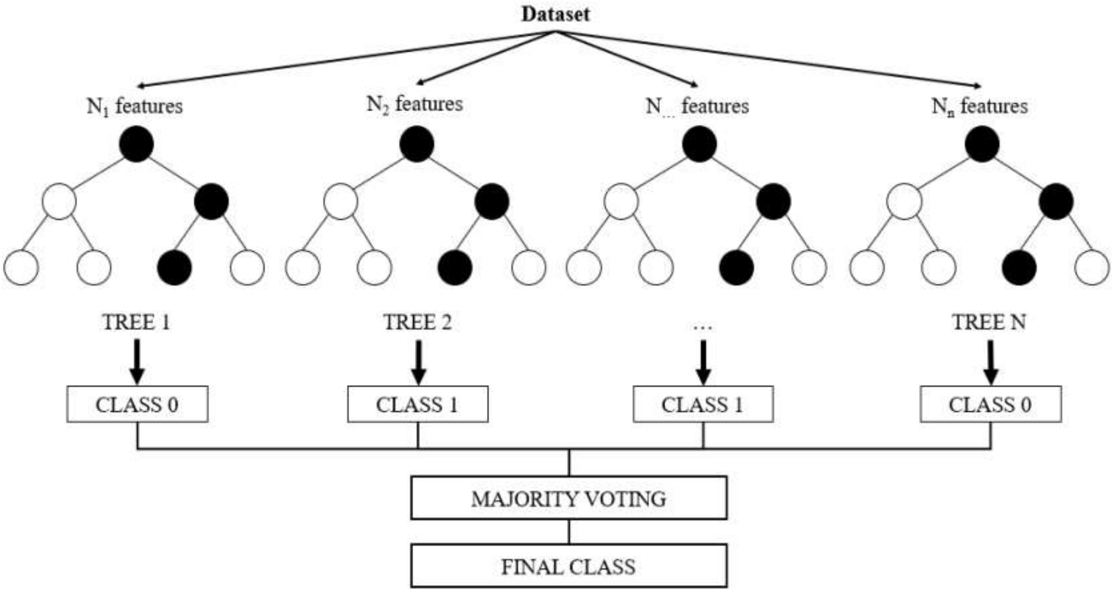

Figure 17. **Random Forest algorithm**

Source: Author's summary

3.7.2. XGBoost

This choice by the author is supported by a detailed scientific foundation, encompassing Enhanced Performance, Handling of Sparse Data, the incorporation of built-in L1 (LASSO) and L2 (Ridge) regularization techniques, and the use of a gradient boosting framework to systematically correct errors from previous trees in the sequence.

The selection of XGBoost is validated by its track record in financial market prediction. It also assesses the importance of features and is engineered to be scalable, efficiently handling large-scale and high-dimensional datasets, a common characteristic of financial data analysis projects. XGBoost is a supervised machine learning algorithm that utilizes a set of decision trees with the aim of optimizing the cost of the objective function.

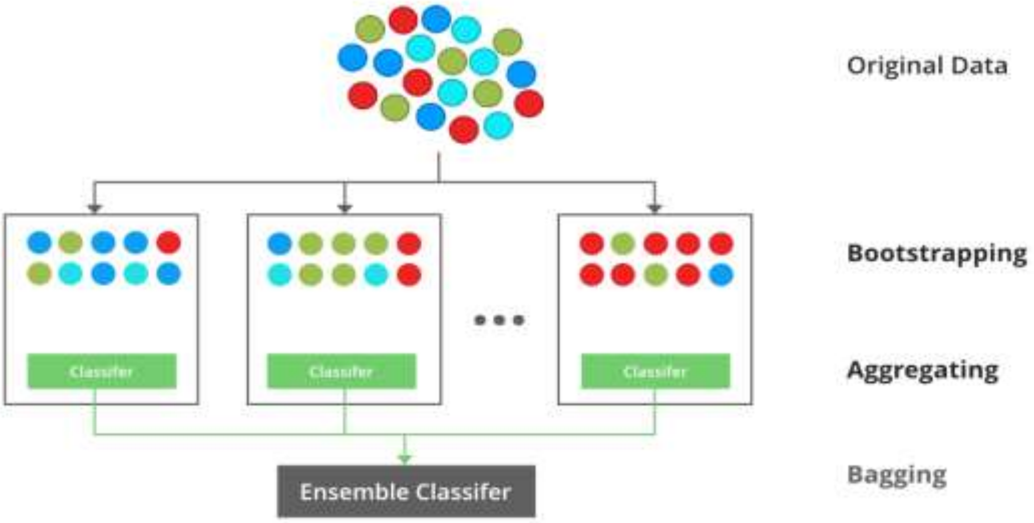

Figure 18. XGBoost model algorithm

Source: Author's summary

XGBoost consists of two main components, a loss function d and a regularization component β:

$$\Omega(Q) = \sum_{i=1}^{n} d(y_i, \hat{y}_i) + \sum_{k=1}^{K} \beta(f_k)$$

Where ŷi is the value to be predicted, n is the number of cases in the test dataset, K is the number of generated trees, and fk is a specific tree out of the total number of trees. The regularization part is expressed with the formula:

$$\beta(f_t) = yT + \frac{1}{2}\left[\alpha \sum_{f=1}^{T} |c_j| + \lambda \sum_{f=1}^{T} c_j^2\right]$$

With y being the minimum level of loss reduction achieved through splitting, λ represents the regularization term of the weight, and c denotes the weight associated with each leaf. Let ft (xi) = cq(xi), where q is in the interval [1, T], T is the number of leaves. An approach is then taken to select the share that has the greatest impact on growth

3.7.3. Decision Tree Regression

For the scientific basis, Decision Tree Regression boasts an algorithm that excels in capturing nonlinear relationships between variables, a common trait of financial market data. It can also handle heterogeneous data types (numeric and categorical) without extensive preprocessing. Furthermore, it serves as a simple and effective tool for preliminary analysis. Another significant reason for the author's choice of this model is that financial datasets often exhibit skewed distributions and outliers. Decision Trees inherently manage such irregularities efficiently, reducing the need for extensive data cleaning and transformation.

While the CART (Classification And Regression Tree) method applies the Gini index to partition data in classification situations, it switches to the least squares method when solving regression problems. This method selects the dividing points to minimize the sum of squared errors (residuals) between predicted and actual values, making the model more accurate. The mathematics of residual sum of squares (RSS) plays an important role in evaluating the effectiveness of each split, guiding the tree-building process to ensure the highest accuracy:

$$RSS = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$$

In which, n is the number of observations, Yi is actual values, Ŷi is predicted values

3.7.4. Linear Regression

The scientific foundation for incorporating this model into the study is based on three elements: Linear Regression is renowned for its simplicity and clarity through a transparent approach. Due to its simplicity, linear regression models can be quickly deployed and interpreted. It excels at quantifying the strength of relationships between variables. The model operates under the assumption of linearity between predictor factors and outcomes. Linear Regression has a long history of application in financial research, where it has been used to model the relationships between stock indices and various economic indicators. This comparison helps to determine the added value of complex algorithms in predicting the movements of the VN-Index. Its proven utility in such contexts reinforces its suitability for this research.

Linear Regression is one of the most basic and popular statistical techniques in Machine Learning, used to predict a continuous variable based on the linear relationship between the target variable and one or more variables. independent (predictors). There are two main types: Simple linear regression (one independent variable) and multivariate linear regression (many independent variables)

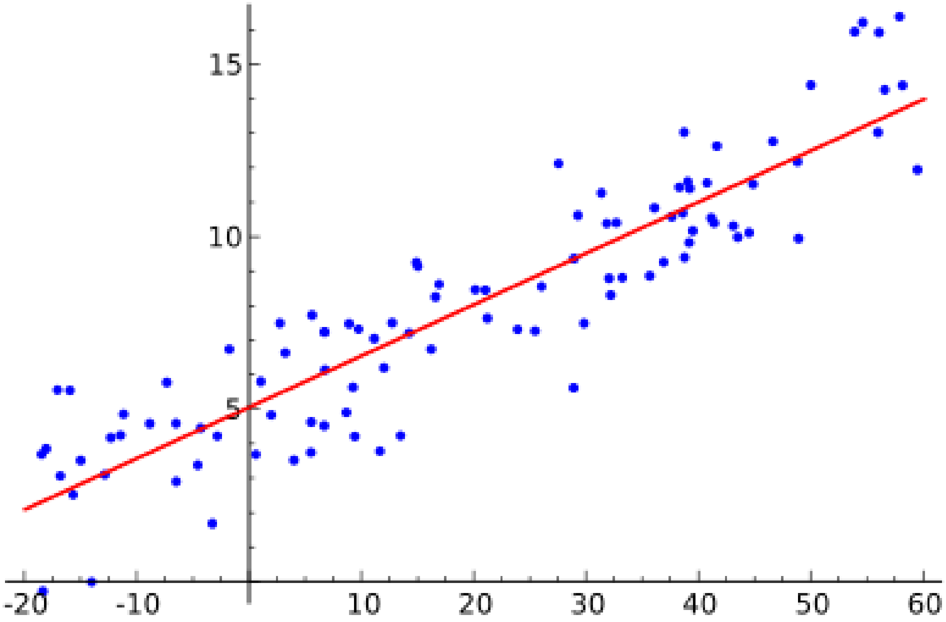

Figure 19. Linear Regression algorithm

Source: Author's summary

Given a data set $\{y_i,\, x_{i1},\ldots,x_{ip}\}_{i=1}^{n}$, of n statistical units, a linear regression model assumes that the relationship between the dependent variable y and the vector of regressors x is linear. This relationship is modeled through a disturbance term or error variable ε — an unobserved random variable that adds "noise" to the linear relationship between the dependent variable and regressors. Thus the model takes the form

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^{\mathsf{T}}\boldsymbol{\beta} + \varepsilon_i, \qquad i = 1,\ldots,n,$$

where $^{\mathsf{T}}$ denotes the transpose so that $\mathbf{x}_i^{\mathsf{T}}\boldsymbol{\beta}$ is the inner product between vectors $\mathbf{x}_i$ and $\boldsymbol{\beta}$. Often these n equations are stacked together and written in matrix notation as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

where

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix},$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^{\mathsf{T}} \\ \mathbf{x}_2^{\mathsf{T}} \\ \vdots \\ \mathbf{x}_n^{\mathsf{T}} \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1p} \\ 1 & x_{21} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{bmatrix},$$

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}.$$

The goal of linear regression is to minimize the squared error (i.e., the distance from the data points to the regression line), thereby finding the coefficients *b* reflects the best linear relationship between the target variable and the independent variable.

3.7.5. Support Vector Machine

The scientific rationale behind the author's choice of the SVM model is recognized for its ability to identify the optimal hyperplane that maximizes the margin between different classes or minimizes error in predictions. The kernel trick enables SVM to transform linearly inseparable data in lower-dimensional space into a higher-dimensional space, facilitating the modeling of complex, nonlinear relationships. Given the volatility of the VN-Index, a model that prevents overfitting is essential for making reliable predictions, such as SVM. Financial datasets often comprise a large number of variables. SVM is efficient in high-dimensional spaces, even when the number of dimensions exceeds the number of samples. The effectiveness of SVM has been proven in applications like financial market prediction.

Support Vector Machine (SVM) is a supervised machine learning algorithm utilized for both classification and regression tasks. In regression problems, the aim of SVM is to find a line (or hyperplane in higher-dimensional spaces) such that the distance from the data points to that line is maximized. In regression problems, SVM is represented as a mathematical equation:

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + b$$

Here: α_i are Lagrange multipliers, y_i represents the actual value of the data point x_i, K(x_i,x) is the kernel function that maps the original space to a higher dimensional space, common kernels include linear, polynomial, radial basis function (RBF), or sigmoid kernel, and b is the bias term.

If f(x) > 0, the data point x is assigned to one hyperplane, whereas if f(x) < 0, the data point x is assigned to the opposite side of the hyperplane. Data points lying on the hyperplane will have f(x) = 0. To predict the value for a new data point, we use the hyperplane equation f(x) previously determined. If f(x) > 0, the model predicts a positive value. Conversely, if f(x) < 0, the model predicts a negative value.

By optimizing ω and b, we aim to minimize the error between predicted values and actual values. To achieve this, we need to optimize a loss function. The loss function for SVM is:

$$L\big(y, f(x)\big) = \max(0, 1 - y \cdot f(x))$$

In which, y is the actual value of the data point, f(x) is the predicted value by the SVM model for the data point x, max(0,1-y·f(x)) is the hinge loss function.

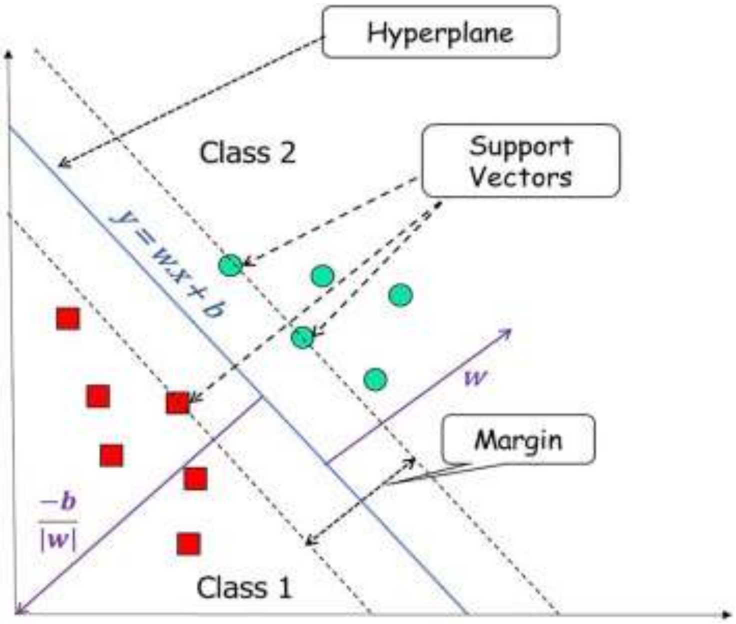

Figure 20. Support Vector Machine algorithm

Source: Author's summary

3.7.6. Artificial Neural Network

The financial market is characterized by its complexity, with numerous variables interacting in non-linear ways that can significantly influence the VN-Index. ANNs are capable of processing these interactions by learning from historical data, making them especially suitable for forecasting tasks in this domain. Their flexibility in architecture (number of layers and neurons) allows for fine-tuning the model to best capture the dynamics of the VN-Index, considering macroeconomic news and technical analysis indicators.

ANN have been selected due to their unparalleled ability to learn and model complex and non-linear relationships from large datasets. The architecture of ANN, inspired by the biological neural networks, enables it to capture intricate patterns in the data, which might not be apparent or accessible to more traditional statistical or machine learning methods. The adaptability of ANNs to different types of data (structured and unstructured) further justifies their choice. Given that this research aims to identify the factors influencing the VN-Index through both quantitative (technical analysis) and qualitative (macroeconomic news) data, ANNs' ability to seamlessly integrate and learn from these diverse data types is invaluable.

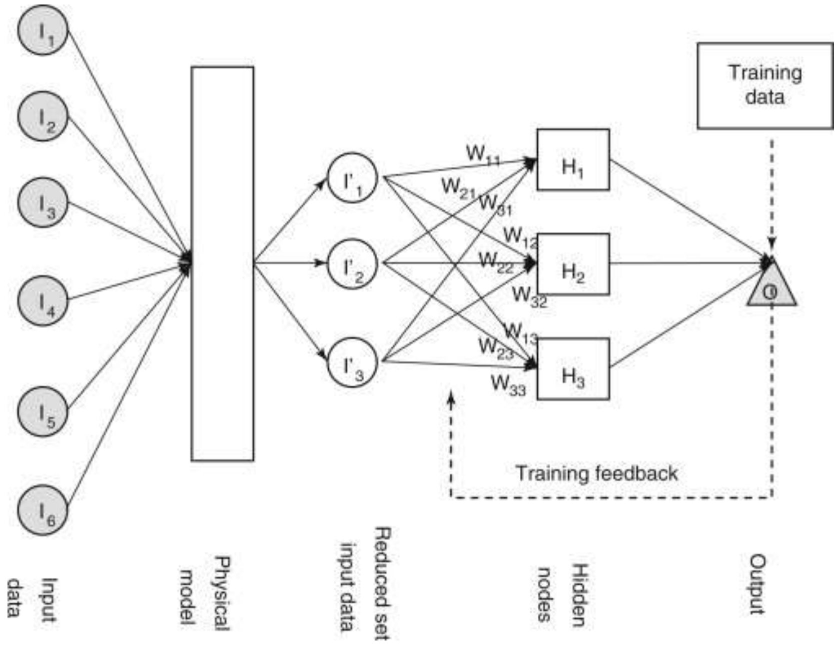

Figure 21 . Artificial Neural Network algorithm

Source: Author's summary

Within the architecture of the ANN model, layers are intricately interconnected, hosting numerous neurons tasked with processing incoming data. The neuron stands as the fundamental component of the ANN. The operational mechanism of neurons within the ANN framework is detailed in Figures 21 and 22. Each neuron is designed to receive an input "x" associated with a weight "w" from "n" neurons positioned in a preceding layer. The primary function of these neurons involves utilizing an activation function denoted as f(x) to yield an output Yj = f(Pj), where Pj represents the aggregate input function at the jth neuron. This process is pivotal for deciphering the underlying connections between input variables and their corresponding outputs (Phong, Tam, & Thanh, 2022)

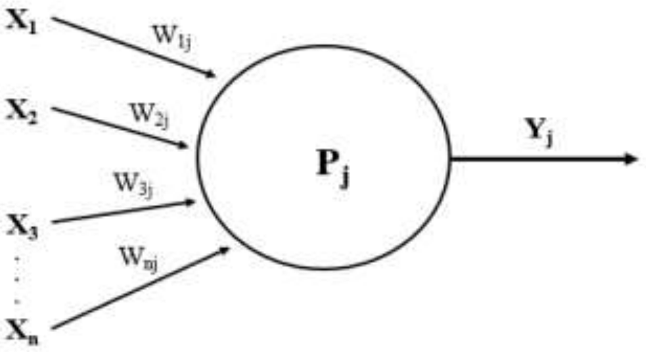

Figure 22. **A neuron in ANN network**

Source: Author’s summary

3.8. Models’ performance check

3.8.1. Coefficient of Determination or R-Squared (R2)

R-squared (R^2), is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model. The R^2 value ranges from 0 to 1. In which, 0 indicates that the model does not explain any of the variability of the response data around its mean, 1 indicates that the model explains all the variability of the response data around its mean. Mathematically, R^2 is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n} (Y_i - \bar{Y}_i)^2}$$

In which, SSres is the sum of squares of residuals (also known as the residual sum of squares). SStot is the total sum of squares (proportional to the variance of the data). R² can be used to gauge the quality of a linear regression model; however, it does not indicate whether the regression estimates and predictions are biased, which is why it should not be used as the sole measure of goodness of fit. Additionally, R² can be inflated by adding more variables to the model, regardless of whether those variables are relevant to the prediction. This is why adjusted R² is sometimes used, as it accounts for the number of variables and the number of observations

3.8.2. Root Mean Square Error (RMSE)

Root Mean Square Error (RMSE) is a standard way to measure the error of a model in predicting quantitative data. It is commonly used in regression analysis to validate experimental results. The formula for calculating RMSE is:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n} \varepsilon_t^2}{n}} = \sqrt{\frac{\sum_{t=1}^{n} \left(Y_t - \hat{Y}_t\right)^2}{n}}$$

In which, yt represents the observed values, y^t represents the predicted values, and n is the number of observations. RMSE gives you the standard deviation of the residuals, which are the differences between the observed and predicted values. Residuals are a measure of how far from the regression line data points are, and RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit. A lower RMSE is generally better than a higher one, as it indicates that the data is closer to the fitted line.

3.8.3. Mean Absolute Error (MAE)

Mean Absolute Error (MAE) is a metric used to evaluate the accuracy of a regression model. It calculates the average absolute difference between the predicted values and the observed (actual) values. Unlike the Mean Squared Error (MSE), MAE does not square the errors before averaging, which means it doesn't disproportionately punish larger errors. The formula for MAE is:

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

In which, yi is the actual observed value from the dataset, y^i is the predicted value from the model, n is the number of observations or data points. The MAE gives an average of how far the model's predictions are from the actual values in a scale that is the same as the data, making it a very interpretable measure of model quality.

3.8.4. Mean Squared Error (MSE)

The Mean Squared Error (MSE) is a common metric used in statistics and machine learning to measure the average squared difference between the estimated values and the actual value. The formula is given by:

$$\mathrm{MSE} = \frac{1}{n} \sum (Y_i - \hat{Y}_i)^2$$

In which, n is the number of observations, y^i is the predicted value, yi is the actual value. MSE is widely used to evaluate the performance of regression models. A lower MSE indicates a better fit of the model to the data. In machine learning, MSE can be used as a loss function called the quadratic loss or L2 loss, guiding the training process to minimize the difference between the predicted and actual values.

3.8.5. Mean Squared Error (MAPE)

MAPE, on the other hand, measures the average absolute percent error for each observation or prediction error. The formula for MAPE is:

$$MAPE = \frac{\sum_{t=1}^{n} \frac{|\varepsilon_t|}{Y_t}}{n} = \frac{\sum_{t=1}^{n} \frac{|Y_t - \hat{Y}_t|}{Y_t}}{n}$$

In which, n is the number of observations, y^i is the predicted value, yi is the actual value. MAPE is often more interpretable to non-technical stakeholders because it expresses errors in percentage terms, showing how large the prediction errors are relative to the actual values. MAPE is beneficial when communicate the error relative to the scale of the data. MAPE is less sensitive to outliers in the prediction but can be heavily influenced by small actual values (since errors are divided by the actual values)

3.8.6. Learning curves

Learning curves in Machine Learning are graphical representations that show how the performance of a machine learning model improves over time with experience, i.e., as it is exposed to more data. These curves are crucial for diagnosing the behavior of a model during the training process. By plotting both the training and validation performance on a graph, learning curves help in understanding:

- How well the model is learning.
- The presence of bias or variance in the model.
- When the model has stopped improving and if it’s overfitting or underfitting.

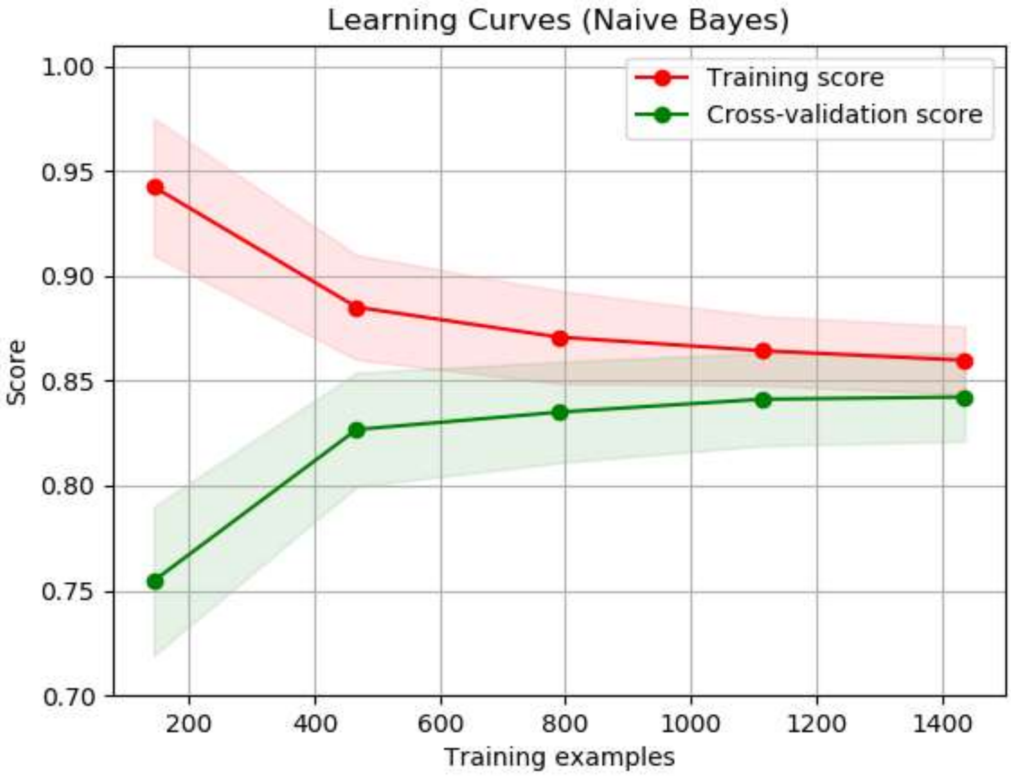

Figure 23. Example of learning curves

Source: Author's summary

If there is a significant gap between the training and validation performance, the model has high variance and is overfitting the training data. In this case, adding more training data can help improve the model. An ideal learning curve is when both the training and validation curves converge at a high level of performance. The gap between the curves should be minimal, indicating that the model generalizes well to unseen data.

3.8.7. K-Fold cross-validation

K-Fold cross-validation technique is a popular method applied to evaluate the reliability of machine learning models. This procedure divides the data set into k parts or "folds", ensuring each part is of equal size.

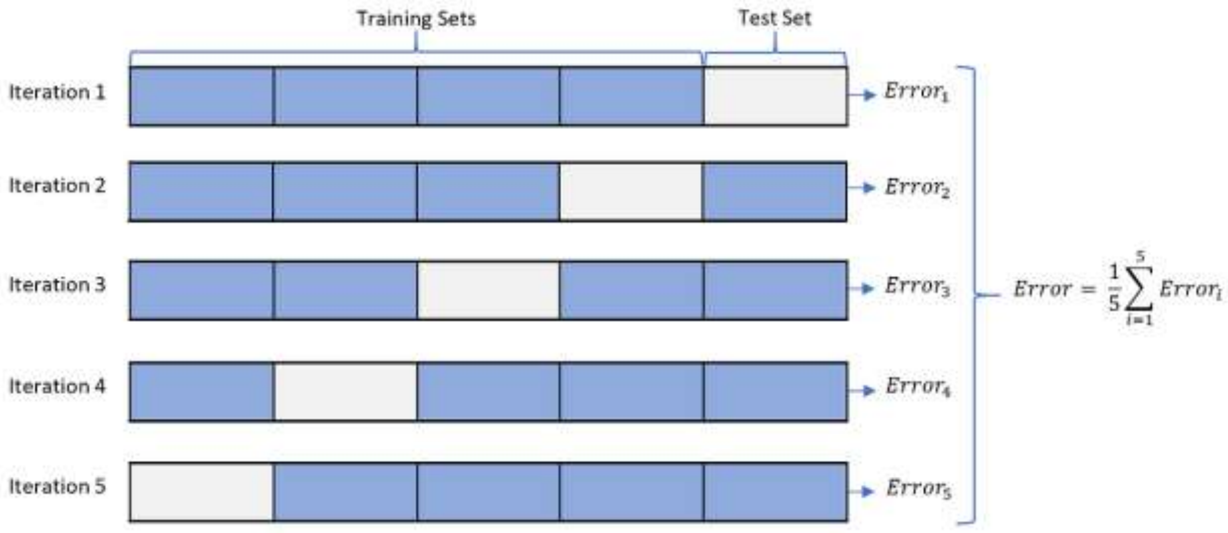

Figure 24. 5-fold cross validation

(Source: (Phung & Rhee, 2019))

During this process, the model will be trained and tested k times, with the condition that each time a different fold will be chosen as the validation data set, and the remaining folds will be used as training data. training. The goal is to evaluate the model on each validation set once, thereby obtaining the results from each iteration. This result includes evaluation metrics such as accuracy and F1 score, from which the average value of them can be calculated across all iterations to estimate the overall performance of the model. K-Fold cross-validation is a flexible and reliable approach, especially important in cases of small data sizes or when many parameters need to be tuned, helping to limit the risk of overfitting and providing provides an overview of the model's predictive ability.

4. Results and discussions

4.1. Results of the Exploratory Data Analysis (EDA)

4.1.1. Descriptive Statistics

Table 3. **Statistical description of numberic variable**

	Closing_price	**Oil_price**	**Exchange_rate**	**Interest_rate**	**CPI**	**Money_supply_M2**	**S&P_500**	**XAU_USD**
count	2002.000000	2002.000000	2002.000000	2002.000000	2002.000000	2.002000e+03	2002.000000	2002.000000
mean	1008.562133	65.705604	23095.115897	0.053988	0.264266	1.089839e+07	3323.271973	1564.271224
std	228.499609	21.026712	612.182465	0.010220	0.440917	2.782728e+06	838.329327	280.207245
min	555.820000	7.790000	21779.000000	0.040000	-1.540000	6.267958e+06	2000.540000	1128.430000
25%	856.285000	55.670000	22727.500000	0.044000	0.010000	8.521098e+06	2642.247500	1284.822500
50%	988.315000	66.820000	23171.000000	0.060000	0.250000	1.075802e+07	3099.015000	1552.695000
75%	1166.967500	77.655000	23321.574707	0.062500	0.510000	1.376102e+07	4129.542500	1824.365000
max	1528.570000	128.260000	24871.000000	0.065000	1.520000	1.535412e+07	5088.800000	2063.810000

Source: Author's summary

The average closing price over the dataset is approximately 1008.56 units, with a standard deviation of 228.50, indicating variability. The range is quite broad, from a minimum of 555.82 to a maximum of 1528.57 units. The mean oil price is around 65.71 units. The oil price has fluctuated significantly, as suggested by its standard deviation of 21.03 and a wide range between the minimum and maximum prices. This indicator has an average value of approximately 23095.12, with variations reflected by the standard deviation of 612.18. The data suggests that the exchange rate has a narrower spread compared to the closing and oil prices, as seen by the smaller standard deviation relative to its mean. Interest rates have a mean of 0.0539 (5.39%) and a very low standard deviation, which indicates a relatively stable interest rate environment within the range of the dataset. The average CPI is 0.264, but with a negative minimum value, it implies there were periods of deflation.

The standard deviation is 0.44, indicating variability in inflation rates. The mean money supply stands at approximately 10.9 million units, and the standard deviation is about 2.78 million, showing substantial changes in the money supply over time. For the S&P 500 index, the average is 3323.27 with a standard deviation of 838.33, reflecting the volatility in the stock market. The average price of gold is around 1564.27 units, with a relatively high standard deviation of 280.21, showing that gold prices can vary widely. Overall, oil price, and S&P 500 index suggest that these markets experience considerable fluctuations, while others, such as the interest rate, show more stability.

4.1.2. Pearson correlation coefficient

In Figure 25, we can see that the Pearson correlation coefficients reveal some interesting insights about the relationship between the stock price and the various features under consideration.

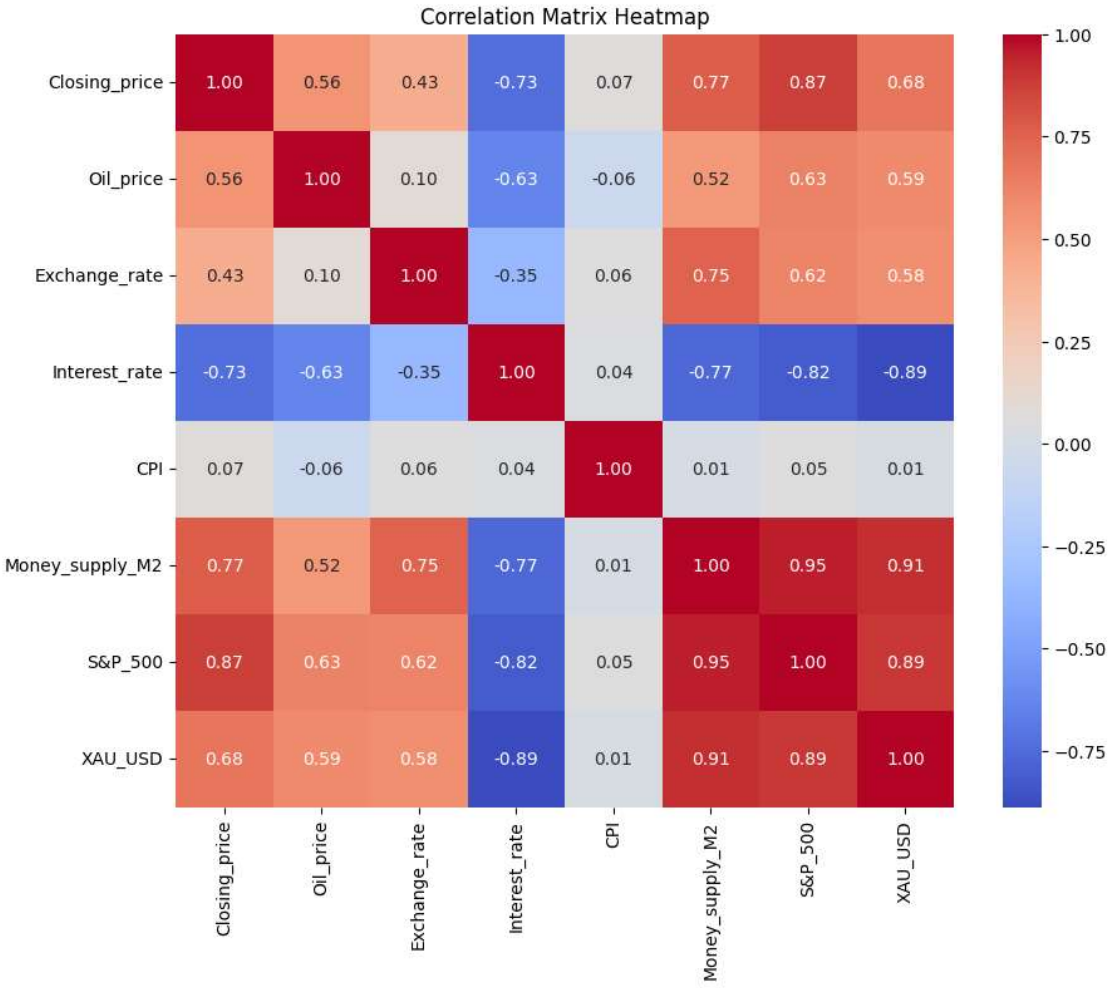

Figure 25. Correlation matrix between variables

Source: Author's summary

The strong positive correlation between Closing_price and S&P_500, as well as Money_supply_M2, reflects the impact of money supply and investor sentiment on stock prices. When the money supply increases, it can lead to higher demand for stock investment, thus driving prices up. Conversely, the strong negative correlation between Interest_rate and Closing_price, S&P_500 and XAU_USD, suggests that interest rates tend to move inversely with stock market indices and gold prices. When interest rates rise, it may lead to a decrease in stock and gold prices, as higher interest rates often increase borrowing costs, prompting investors to shift funds away from stocks and gold. Notably, the strong correlation between S&P_500 and Money_supply_M2 indicates that fluctuations in the money supply may significantly impact the stock market, aligning with theories of securities and monetary policy. XAU_USD (gold price) exhibits a moderately positive correlation with Closing_price and S&P_500, indicating that gold is often considered a safe haven investment when stock markets rise or used as a hedging tool. The weaker correlation of oil price with other indices may be attributed to the influence of various factors on oil price volatility, such as OPEC policies, production levels, and global demand, not solely dependent on overall economic fluctuations. CPI appears to have less clear correlations with other variables. CPI measures inflation, which may not immediately affect stock or gold prices, or due to the time lag between inflation and market reactions.

4.1.3. Data visualization

Based on the descriptive statistics and the level of correlation mentioned earlier, the author seeks to gain a broader overview of the fluctuations in macroeconomic variables in relation to the closing price to gain deeper insights into the Vietnamese stock market as well as the domestic and international economy.

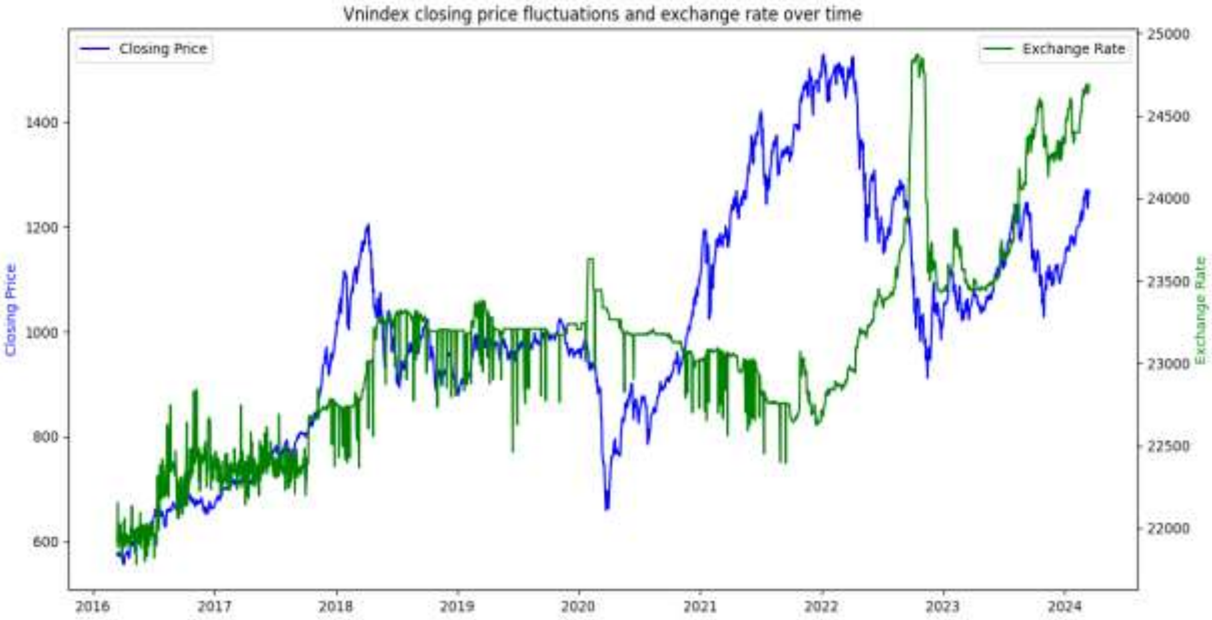

Figure 26. Fluctuations in closing prices and exchange rate over time

Source: Author’s summary

.

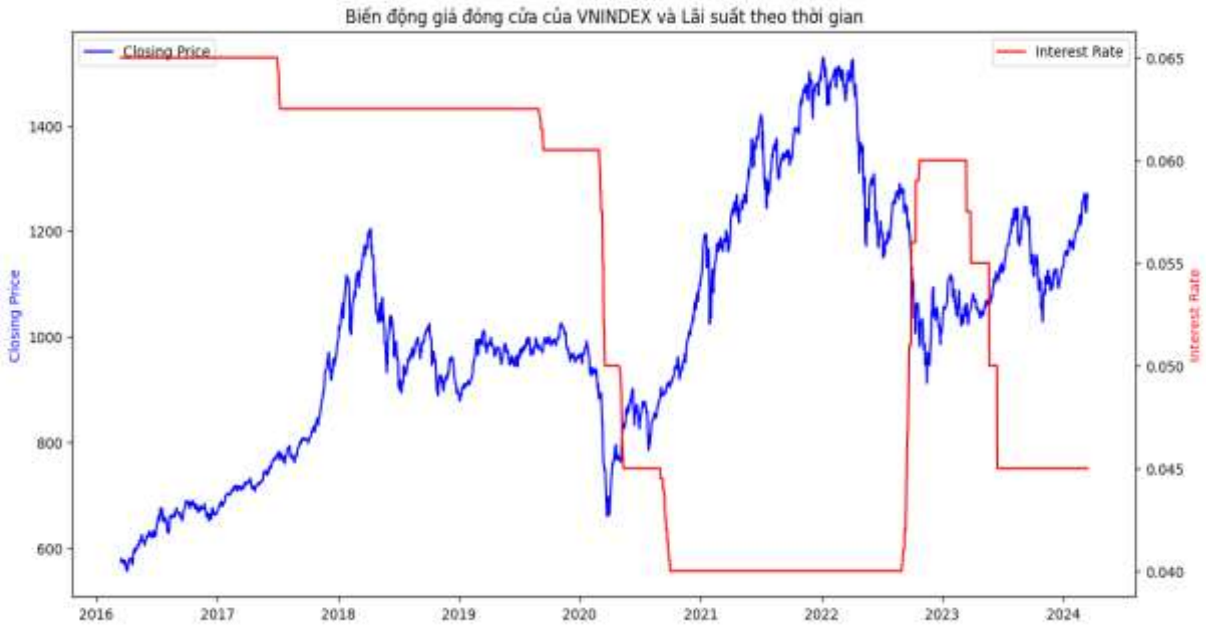

Figure 27. Fluctuations in closing prices and interest rates over time

Source: Author’s summary

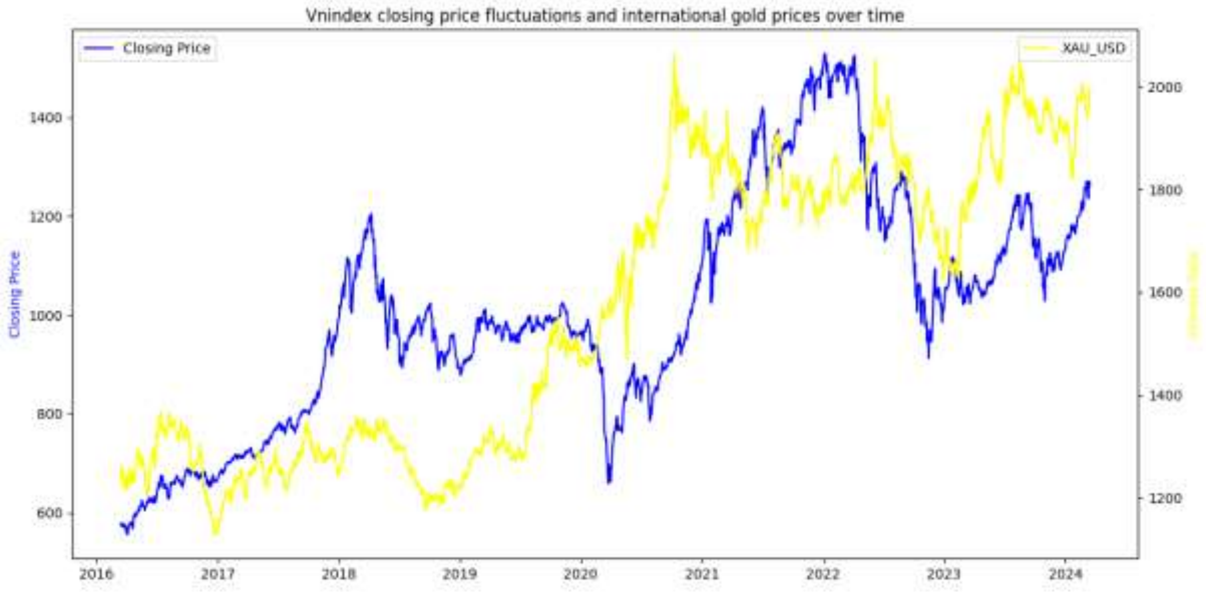

Figure 28. Fluctuations in closing prices and international gold over time

Source: Author’s summary

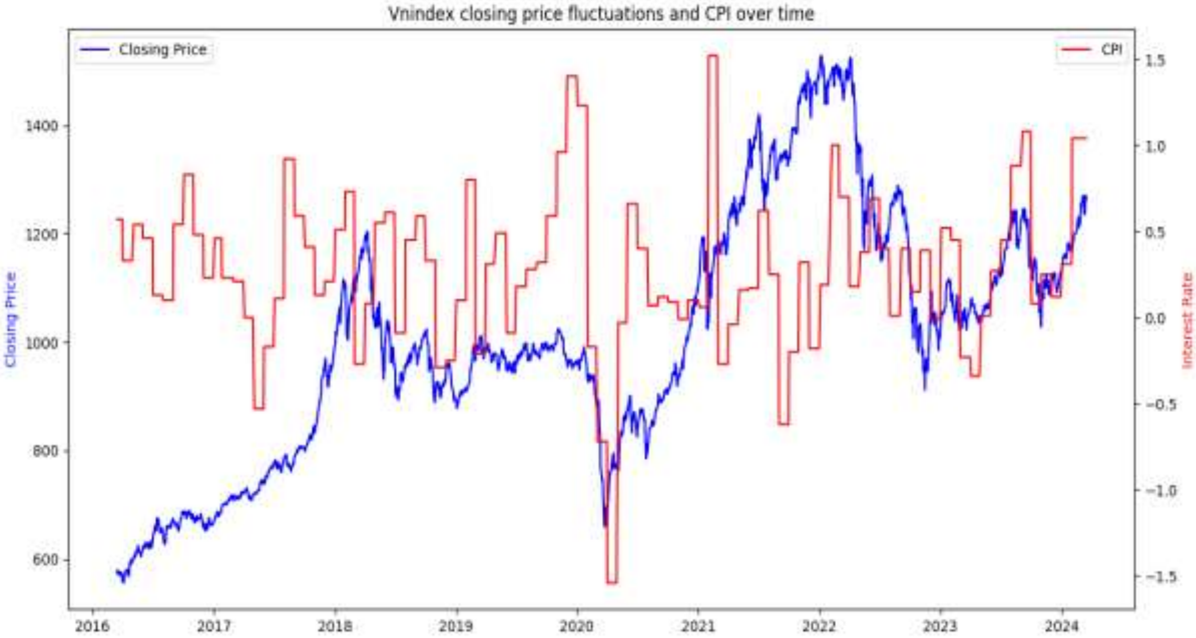

Figure 29. Fluctuations in closing prices and CPI over time

Source: Author's summary

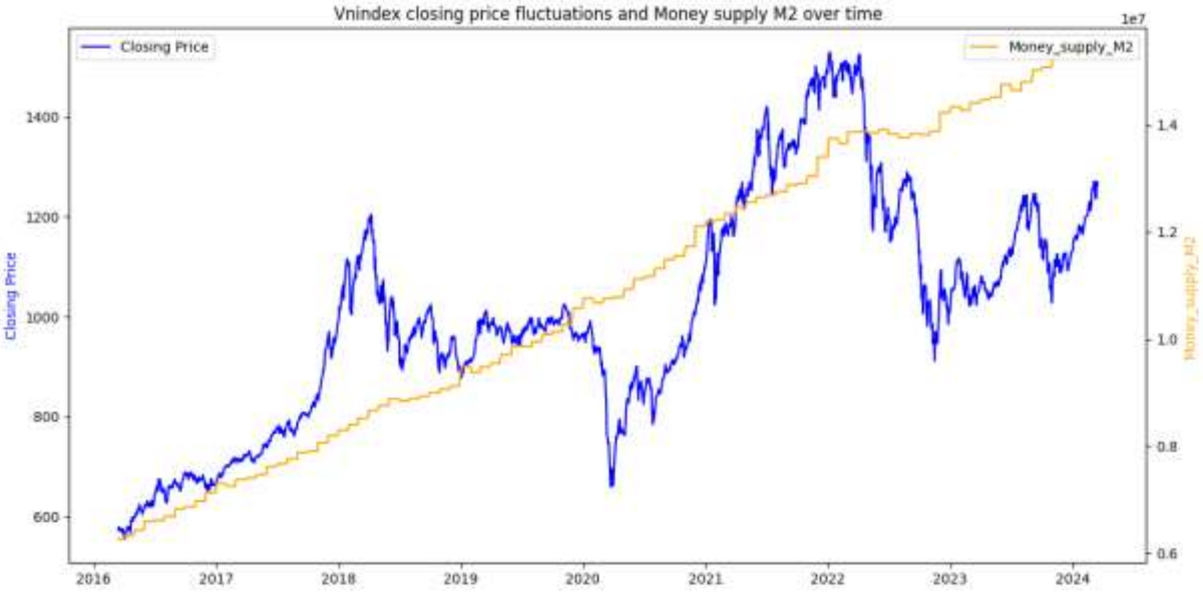

Figure 30. Fluctuations in Closing prices and Money supply M2 over time

Source: Author's summary

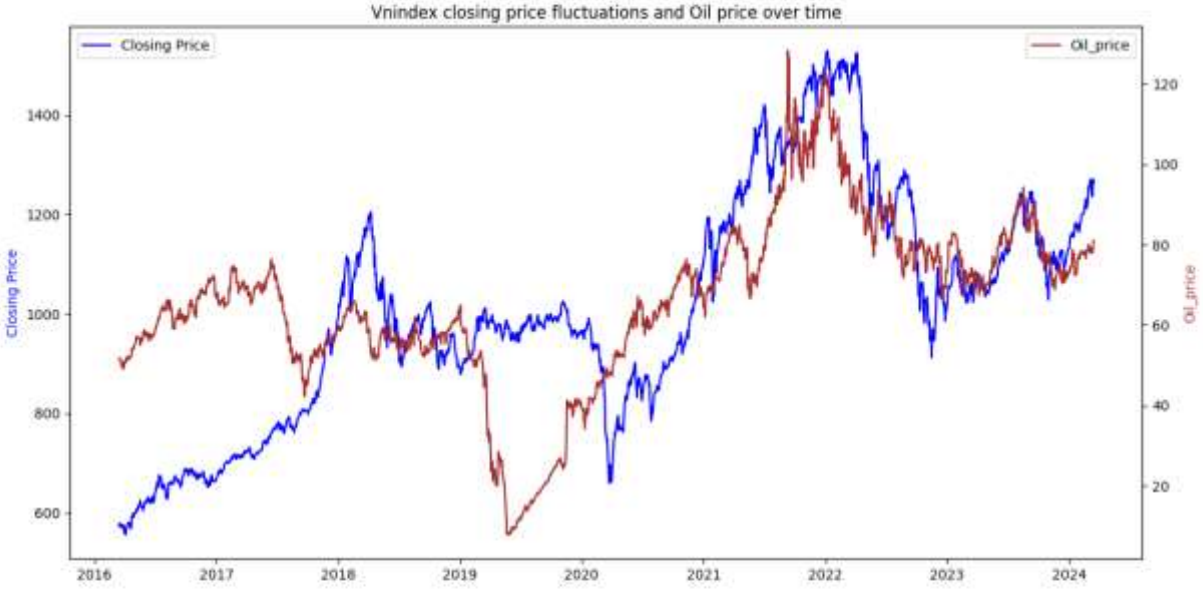

Figure 31. Fluctuations in Closing prices and Oil price over time

Source: Author's summary

The visualization results indicate that the Closing Price tends to increase over the years, with significant fluctuations related to specific economic or political events. There was a significant decline around the year 2020 due to the impact of the COVID-19 pandemic on the stock market. The exchange rate shows strong fluctuations and a trend of increase over time. Particularly, the exchange rate seems to recover after each downturn, which may reflect the intervention of central banks or changes in monetary policy. When interest rates decrease, the VNINDEX tends to rise, and vice versa. This aligns with the fundamental financial theory that lower interest rates may encourage investment in the stock market due to reduced capital borrowing costs. The VNINDEX and world gold prices appear to move inconsistently. There are periods when the VNINDEX rises while gold prices fall, and vice versa, indicating no close and stable relationship between the two markets. However, in some periods, such as in 2020 and early 2021, both show an upward trend, possibly due to concerns about inflation or global economic instability prompting investors to seek gold as a safe haven investment.

The CPI fluctuates but does not follow a clear trend like the VNI. The CPI has periods of increase as well as decrease. In some periods, the VNI and CPI seem to move in sync, meaning that when the CPI rises, the VNI also increases, which may partly indicate that stock price increases are related to inflation. In Figure 30, it can be observed that when the money supply M2 increases, the VNI also tends to increase. There are periods when the increase in the money supply M2 is not accompanied by an increase in the VNI, and vice versa, indicating that there are other factors influencing the stock market besides the money supply.

It is noteworthy that the VNI and S&P 500 show synchronous movements, with clear peaks and troughs over the years. The most evident observation is the deep simultaneous decline of both indices during a global economic event, namely the period when the world was heavily affected by the COVID-19 pandemic for over a year. This demonstrates that both indices reflect the global macroeconomic situation, and significant fluctuations can cause simultaneous impacts across multiple markets. Although the world oil price affects the VNI, the extent of influence and the relationship between them are not entirely consistent across different periods.

4.2. Results of the performance metrics

Based on the computed performance metrics for the models, the author proceeds to compare the three models with the most effective predictive performance with the aim of selecting the most optimal model for this study.

Table 4. Performance and accuracies of models

	Model	MSE	RMSE	MAE	MAPE	R-Squared
1	Random Forest	4.197507e+02	20.487818	12.420641	1.209722	0.991750
2	XGBoost	4.959086e+02	22.269005	14.214132	1.391971	0.990253

	Model	MSE	RMSE	MAE	MAPE	R-Squared
3	Linear Regression	7.935919e+03	89.083775	66.476096	6.860058	0.844021
4	Decision Tree	7.247153e+02	26.920537	15.114913	1.489434	0.985756
5	KNN	4.476300e+02	21.157269	13.684214	1.350528	0.991202
6	ANN	5.519568e+06	2349.376165	1844.781844	168.855271	-107.485870
7	SVM	3.122309e+04	176.700570	136.190948	14.281501	0.386317

Source: Author's summary

This model yielded the best results among the tested models, boasting a high R-Squared value (0.991750) and the lowest errors (MSE, RMSE, MAE). The low MAPE indicates the model's accurate predictive ability, with an average prediction error of only about 1.21%. Similar to the Random Forest, XGBoost also performs highly effectively but with slightly higher error indicators and a marginally lower R-Squared. This model is not as efficient as decision tree-based models. The error indicators are very high (MSE, RMSE, and MAE are all elevated) and the R-Squared is low (0.844021), indicating poor predictive accuracy. Decision Tree Regression is a fairly good model with an R-Squared of 0.985756; however, its error metrics are higher compared to Random Forest and XGBoost. KNN demonstrates good predictive capability with an R-Squared of 0.991202, only slightly lower than Random Forest, but with higher error metrics. This model performed very poorly, with extremely high error metrics and a negative R-Squared (-107.485870). This might indicate overfitting, excessive complexity, or errors in the training process. This model has the poorest predictive performance among the considered models, with an R-Squared of only 0.386317 and high error metrics.

In conclusion, tree-based models (Random Forest, XGBoost, and Decision Tree) show good predictive performance, with Random Forest emerging as the best model. Linear Regression is not suited for this data type or the problem at hand. ANN and SVM do not align with the data or current approach and need to be adjusted or replaced with alternative models.

4.3. Results of Model Comparison

Based on the calculated performance metrics for the models, the author proceeds to compare the three models with the most effective predictive performance with the aim of selecting the most optimal model for this study

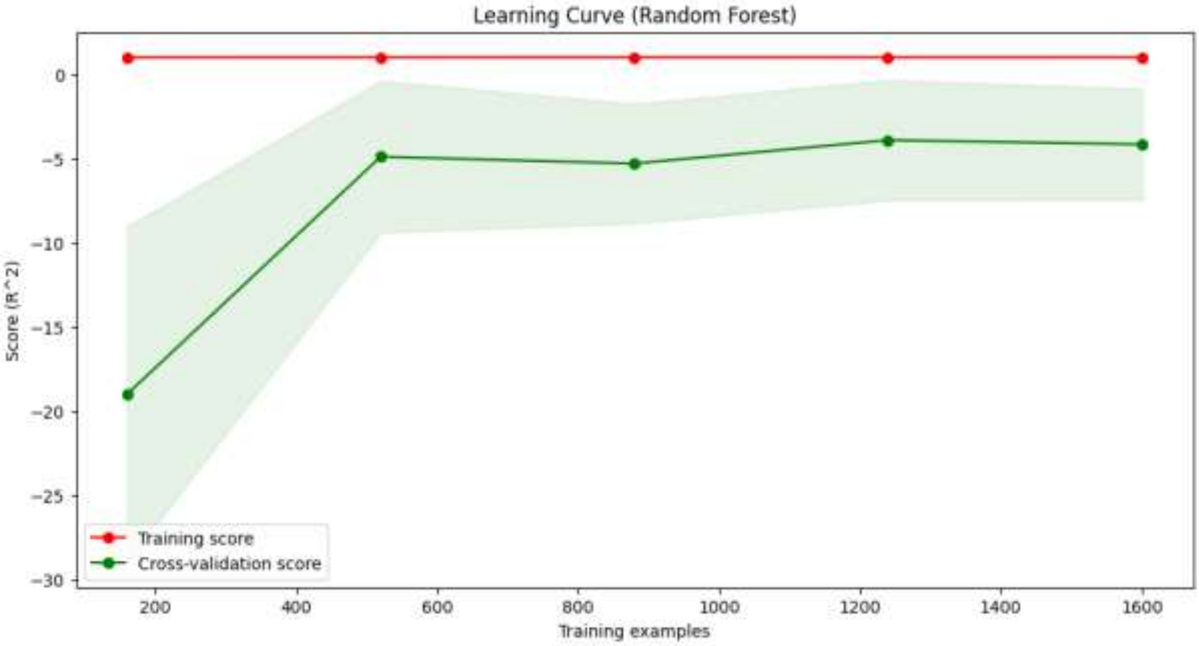

Figure 32. Learning Curve of Random Forest

Source: Author's summary

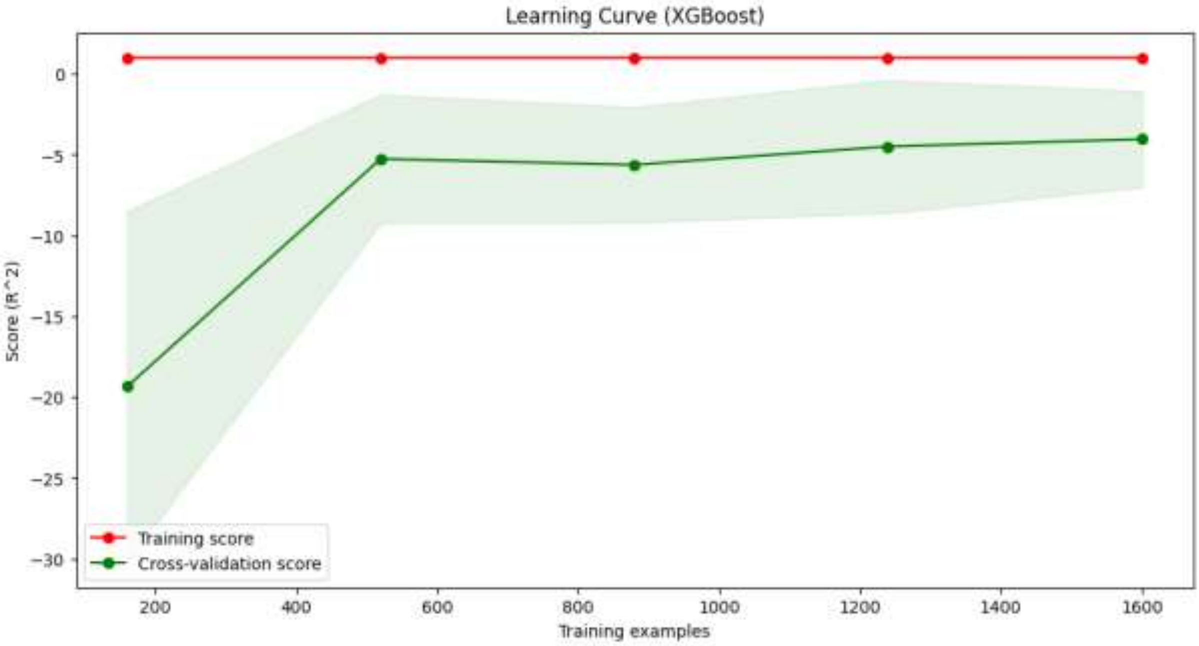

Figure 33. Learning Curve of XGBoost

Source: Author's summary

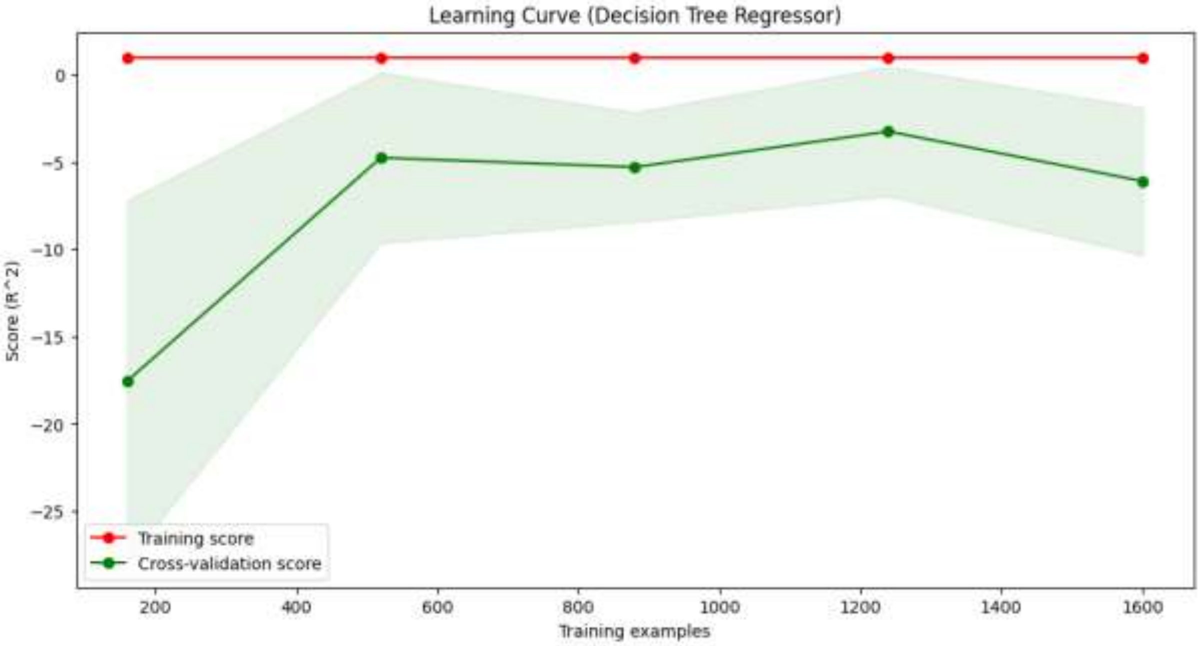

Figure 34 . Learning Curve of Decision Tree Regression

Source: Author's summary

Regard as Random Forest, the cross-validation score also increases and starts to plateau, suggesting that adding more data beyond a certain point may not lead to significant improvements in model performance. The gap between the training and cross-validation scores suggests the model could be overfitting the training data since it performs well on training but slightly worse on unseen data. Random Forest seems to have the best generalization ability among the three, with the highest R^2 and Explained Variance scores. Based on the learning curves for XGBoost, the cross-validation score improves as more data is added, and the convergence between the training and validation scores is a good sign, indicating a well-generalizing model. However, there's still a noticeable gap between the training and validation scores, hinting at some overfitting. XGBoost also generalizes well but has more variance in performance across different folds compared to Random Forest. The cross-validation score of Decision Tree Regression improves significantly as more data is provided, but it doesn't quite reach the level of the training score, suggesting some overfitting. The variance between folds in cross-validation is quite significant, which might imply the model's performance is highly dependent on the data it's being tested on.

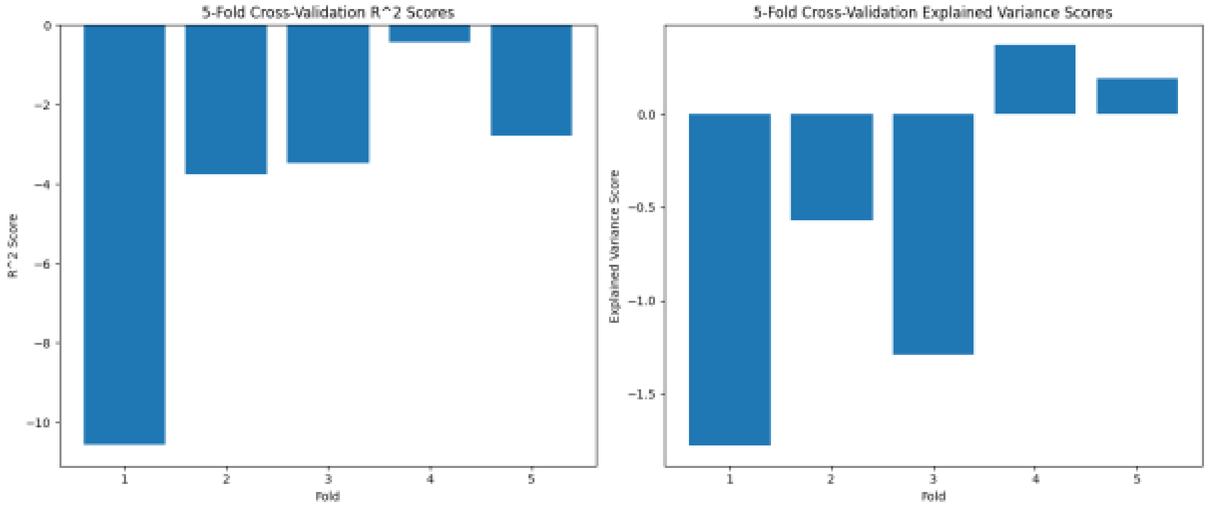

Figure 35 . Cross-validation score of Random Forest

Source: Author's summary

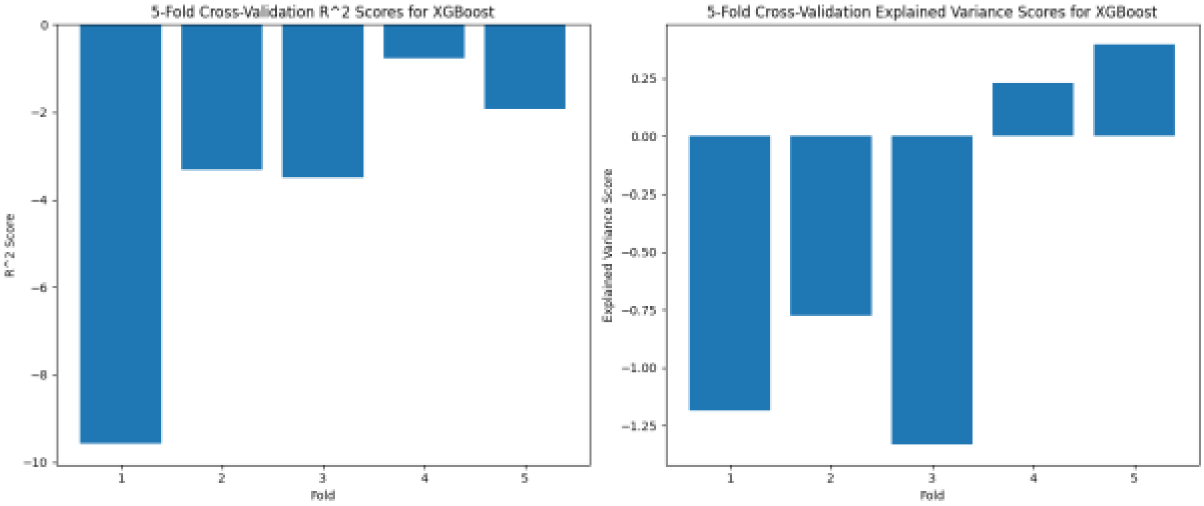

Figure 36. Cross-validation score of XGBoost

Source: Author's summary

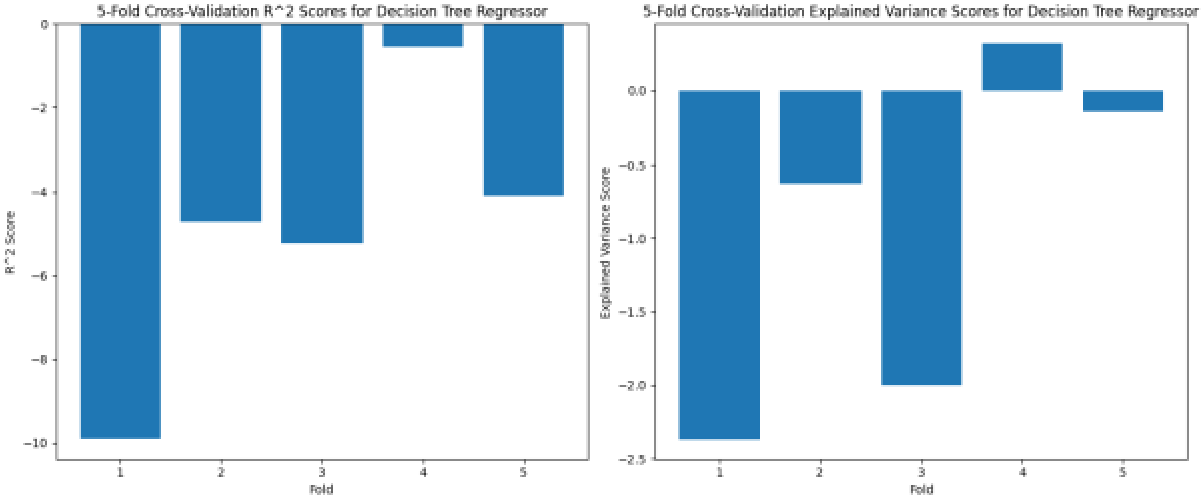

Figure 37. Cross-validation score of Decision Tree Regression

Source: Author's summary

According to the cross-validation R^2 and Explained Variance scores, they offer a view into how well the models predict unseen data. Random Forest seems to have the best generalization ability among the three, with the highest R^2 and Explained Variance scores. XGBoost also generalizes well but has more variance in performance across different folds compared to Random Forest. Decision Tree exhibits the most variability and potentially the poorest generalization, which is typical for this type of model because it tends to overfit the training data without proper tuning or ensemble methods.

Therefore, all three models show a good learning curve, but the Random Forest and XGBoost models display better generalization capabilities with less variance in the validation scores, suggesting they would perform better on new, unseen data compared to the Decision Tree Regressor. Finally, summarize the results, although both 2 models show excellent predictive capabilities, but the Random Forest seems to perform slightly better across most metrics and displays a more stable learning curve, suggesting better generalization. The R-Squared values from the cross-validation (as seen in the bar charts) are also consistently higher for Random Forest than for XGBoost, which further supports the conclusion that Random Forest is the better model out of the two for this particular dataset.

In summary, for this particular set of data, the Random Forest model is the recommended choice due to its superior ability to capture the underlying patterns in the data, making it the most effective model among those tested. The remaining two tree models, XGBoost and Decision Tree, also show good performance, but not as good as Random forest. Additionally, SVM and KNN, did not show good performance. Meanwhile, the Deep Learning model ANN does not seem to be suitable for the data set and approach applied in this study.

5. Conclusions and Recommendations

5.1. Conclusions

A variety of data analysis techniques can be employed to examine the impact of economic indicators. Previous studies predominantly utilized various statistical regression techniques to achieve this goal, often only testing the impact of selected indicators. This research aims to identify the key factors influencing the VN-Index by using machine learning techniques, focusing on macroeconomic news and technical analysis. Through comprehensive data analysis from March 2016 to March

2024, the author employs multiple machine learning models including Decision Tree, Random Forest, XGBoost, KNN, ANN, SVM, and Linear Regression. The findings indicate that among the evaluated models, Random Forest exhibits superior predictive performance, making it an excellent choice for predicting the movements of the VN-Index based on macroeconomic variables. SHAP analysis reveals that the S&P 500 index, money supply M2, exchange rates, and interest rates significantly impact the Vietnamese stock market. Conversely, global oil prices, the CPI, and global gold prices have a lesser impact. The study emphasizes the profound influence of global stock market trends, particularly the S&P 500, on the VN-Index, highlighting the interconnectedness of global financial markets. Moreover, domestic monetary policy, especially the money supply M2, plays a crucial role in the stock market dynamics, underscoring the importance of monetary stability for market health.

5.2. Recommendations

Successfully forecasting stock market movements can bring numerous benefits to managers, investors, as well as the Government. To effectively apply predictive models in practice, the author presents several recommendations suitable for different stakeholders.

For policymakers and regulatory bodies, they should enhance the financial market's resilience by strengthening legal frameworks to ensure transparency and stability. Furthermore, they need to implement policies to maintain an optimal money supply, thereby preventing inflation or deflation that could destabilize the market. Simultaneously, they should promote more rigorous monitoring and analysis of global economic trends to predict their impact on the domestic market.

For Investors, they should incorporate advanced analytical tools and machine learning models in investment strategies to better predict market movements.

Diversify investment portfolios to hedge against potential risks posed by global economic volatility and domestic monetary policy adjustments.

For researchers and academics, they should continue the exploration of machine learning applications in financial market analysis, with an emphasis on developing models that can adapt to the rapid changes in economic indicators. Encourage interdisciplinary studies that combine economics, finance, and data science to uncover deeper insights into market dynamics.

5.3. Limitations and further research

While this study provides valuable insights into the factors affecting the VN-Index and introduces a novel application of machine learning models for financial analysis, it acknowledges certain limitations

The first limitation of this study is that the reliance on data from March 2016 to March 2024, while extensive, might not capture the full spectrum of market behaviors, especially those influenced by rare or unprecedented economic events. The evolving nature of financial markets means that future conditions or crises might present challenges not observed in the current dataset.

The second limitation of this study is that the predictive models used, including Random Forest, XGBoost, and others, although effective, might exhibit limitations in generalizing their findings across different economic contexts or time periods. The performance of these models is contingent on the specific characteristics of the data and might not directly apply to other stock indices or financial markets without adjustments.

The thrid limitation of this study is that the study focuses on selected macroeconomic variables known to influence the stock market. However, the complex and interconnected nature of global financial systems suggests that other, unconsidered variables might also play significant roles. The omission of certain variables or the inability to capture the entirety of market influencers can lead to incomplete conclusions.

The fourth limitation of this study is that the rapid advancement in machine learning and data analysis techniques presents both an opportunity and a limitation. While the current study utilizes state-of-the-art methodologies, future developments could offer more refined tools for analysis, potentially rendering current methods less effective or obsolete

The fifth limitation of this study is that financial markets are occasionally influenced by unforeseen events or shocks, known as Black Swan events, which are difficult to predict and model. The ability of the current models to anticipate such events or their impacts on the VN-Index is limited, underscoring the need for models that can adapt to sudden market changes.

Future research should address these limitations by expanding the dataset to cover a broader time frame, including a wider range of variables, and exploring advanced modeling techniques. Investigating the impact of qualitative factors and developing models capable of adapting to unforeseen events could also enhance the robustness and applicability of future studies. This expanded approach will contribute to a more holistic understanding of the dynamics influencing the VN-Index and other financial indices.

APPENDIX

Appendix 1. Data

Date	Closing_price	Oil_price	Exchange_rate	Interest_rate	CPI	Money_supply_M2	S&P_500	XAU_USD
15/3/2016	574.04	51.74	21914	0.07	0.57	6267958	2015.93	1232.38
16/3/2016	577.07	51.75	22152	0.07	0.57	6267958	2027.22	1263.08
17/3/2016	579.26	51.28	21885	0.07	0.57	6267958	2040.59	1257.90
18/3/2016	575.82	51.34	22018	0.07	0.57	6267958	2049.58	1255.30
21/3/2016	572.27	51.36	21933	0.07	0.57	6267958	2051.60	1243.74
22/3/2016	570.91	50.30	21923	0.07	0.57	6267958	2049.80	1248.38
23/3/2016	574.71	49.83	22023	0.07	0.57	6267958	2036.71	1220.17
24/3/2016	570.66	49.66	21931	0.07	0.57	6267958	2035.94	1216.99
25/3/2016	572.08	50.56	21974	0.07	0.57	6267958	2037.05	1216.47
28/3/2016	575.72	49.03	21821	0.07	0.57	6267958	2055.01	1221.05
29/3/2016	568.28	49.28	21957	0.07	0.57	6267958	2063.95	1242.55
30/3/2016	569.91	49.43	21816	0.07	0.57	6267958	2059.74	1224.85
31/3/2016	561.22	50.74	21943	0.07	0.57	6267958	2072.78	1232.70
1/4/2016	558.43	50.97	21895	0.07	0.33	6326334	2066.13	1222.49
4/4/2016	555.82	50.60	22044	0.07	0.33	6326334	2045.17	1215.40
5/4/2016	560.32	51.31	21942	0.07	0.33	6326334	2066.66	1231.14
6/4/2016	567.79	51.82	22058	0.07	0.33	6326334	2041.91	1222.35
7/4/2016	571.60	51.79	21934	0.07	0.33	6326334	2047.60	1240.69
8/4/2016	572.34	51.98	21882	0.07	0.33	6326334	2041.99	1240.39
11/4/2016	579.27	52.08	21924	0.07	0.33	6326334	2061.72	1257.95
12/4/2016	579.84	51.39	21964	0.07	0.33	6326334	2082.42	1255.94
13/4/2016	578.02	51.84	21922	0.07	0.33	6326334	2082.78	1243.14
14/4/2016	579.49	51.91	21913	0.07	0.33	6326334	2080.73	1227.39
15/4/2016	579.86	51.72	21951	0.07	0.33	6326334	2094.34	1234.05
19/4/2016	568.28	52.27	21940	0.07	0.33	6326334	2100.80	1231.55
20/4/2016	568.03	51.95	21896	0.07	0.33	6326334	2102.40	1250.50
21/4/2016	575.73	52.50	21949	0.07	0.33	6326334	2091.48	1244.05
22/4/2016	592.48	54.00	21891	0.07	0.33	6326334	2091.58	1248.85
25/4/2016	591.58	53.91	22008	0.07	0.33	6326334	2087.79	1232.30
26/4/2016	598.48	53.91	21977	0.07	0.33	6326334	2091.70	1237.90
27/4/2016	593.96	54.43	21960	0.07	0.33	6326334	2095.15	1243.44
28/4/2016	591.67	54.11	22134	0.07	0.33	6326334	2075.81	1246.10
29/4/2016	598.37	54.79	21934	0.07	0.33	6326334	2065.30	1266.48
4/5/2016	599.07	55.59	21989	0.07	0.54	6427397	2081.43	1293.74
5/5/2016	601.51	55.72	21917	0.07	0.54	6427397	2063.37	1291.60
6/5/2016	606.52	57.12	21852	0.07	0.54	6427397	2051.12	1286.00
9/5/2016	603.85	56.91	21991	0.07	0.54	6427397	2050.63	1279.49
10/5/2016	605.05	56.77	21973	0.07	0.54	6427397	2057.14	1278.05
11/5/2016	614.06	56.96	21779	0.07	0.54	6427397	2058.69	1288.70

12/5/2016	612.12	56.81	21963	0.07	0.54	6427397	2084.39	1263.85
13/5/2016	610.82	56.84	21998	0.07	0.54	6427397	2064.46	1266.40
16/5/2016	615.78	56.67	22004	0.07	0.54	6427397	2064.11	1277.69
17/5/2016	624.75	55.02	21916	0.07	0.54	6427397	2046.61	1263.49
18/5/2016	622.45	55.25	21975	0.07	0.54	6427397	2066.66	1273.60
19/5/2016	619.20	55.17	21902	0.07	0.54	6427397	2047.21	1273.86
20/5/2016	614.81	56.61	21940	0.07	0.54	6427397	2047.63	1280.00
23/5/2016	611.03	56.58	22092	0.07	0.54	6427397	2040.04	1258.30
24/5/2016	611.62	56.16	21920	0.07	0.54	6427397	2052.32	1255.10
25/5/2016	611.89	56.88	22020	0.07	0.54	6427397	2048.04	1252.30
26/5/2016	604.34	57.79	21878	0.07	0.54	6427397	2076.06	1249.24
27/5/2016	608.11	58.22	22003	0.07	0.54	6427397	2090.54	1227.00
30/5/2016	614.50	58.68	21950	0.07	0.54	6427397	2090.10	1224.18
31/5/2016	618.44	58.71	21940	0.07	0.54	6427397	2099.06	1220.15
1/6/2016	619.86	57.71	22018	0.07	0.46	6592394	2096.96	1212.40
2/6/2016	623.37	57.56	21956	0.07	0.46	6592394	2099.33	1205.58
3/6/2016	621.88	57.23	21945	0.07	0.46	6592394	2105.26	1215.24
6/6/2016	620.05	57.24	21797	0.07	0.46	6592394	2099.13	1213.35
7/6/2016	624.65	58.17	22004	0.07	0.46	6592394	2109.41	1210.95
8/6/2016	627.87	58.09	21888	0.07	0.46	6592394	2112.13	1244.49
9/6/2016	631.26	57.35	21983	0.07	0.46	6592394	2119.12	1245.24
10/6/2016	629.84	57.34	22026	0.07	0.46	6592394	2115.48	1243.85
….	….	….	….	…	….	…	…	….
21/2/2024	1230.04	76.61	24525	0.05	1.04	15354119	4906.19	2005.78
22/2/2024	1227.31	77.81	24525	0.05	1.04	15354119	4958.61	1995.88
23/2/2024	1212.00	78.58	24590	0.05	1.04	15354119	4942.81	1982.70
26/2/2024	1224.17	78.30	24630	0.05	1.04	15354119	4954.23	1982.15
27/2/2024	1237.46	78.41	24660	0.05	1.04	15354119	4995.06	1985.51
28/2/2024	1254.55	79.96	24625	0.05	1.04	15354119	4997.91	1992.27
29/2/2024	1252.73	80.06	24620	0.05	1.04	15354119	5026.61	1977.68
1/3/2024	1258.28	78.81	24640	0.05	1.04	15354119	5021.84	1968.40
4/3/2024	1261.41	78.34	24640	0.05	1.04	15354119	4953.17	1949.79
5/3/2024	1269.98	79.17	24676	0.05	1.04	15354119	5000.62	1958.19
6/3/2024	1262.73	79.31	24690	0.05	1.04	15354119	5029.73	1936.79
7/3/2024	1268.46	77.93	24665	0.05	1.04	15354119	5005.57	1945.89
8/3/2024	1247.35	77.89	24680	0.05	1.04	15354119	4975.51	1962.67
11/3/2024	1235.49	78.16	24640	0.05	1.04	15354119	4981.80	1959.09
12/3/2024	1245.00	78.06	24635	0.05	1.04	15354119	5087.03	1980.85
13/3/2024	1270.51	79.75	24645	0.05	1.04	15354119	5088.80	1980.01
14/3/2024	1264.26	80.97	24675	0.05	1.04	15354119	5069.53	1977.19
15/3/2024	1263.78	81.03	24690	0.05	1.04	15354119	5078.20	1998.37

Source: Author's summary

Appendix 2. Source code

```
import pandas as pd
# Load the data from the provided Excel file
data_path = '/content/gdrive/MyDrive/Dữ liệu Đồ án tốt nghiệp (Autosaved).xlsx'
data= pd.read_excel(data_path)
# Display the first few rows of the dataframe to understand its structure
data.head()
data.isna().sum()
data.duplicated().sum()
# check features (quantity, data type)
# displays a list of column names for checking and debugging
data.columns.tolist()
data.info()
# Perform descriptive statistics for the variables in the data
description = data.describe()
description
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
# Assuming 'data' is your DataFrame and it has already been defined earlier in your code.
correlation_matrix = data.corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
```

```
plt.title('Correlation Matrix Heatmap')

plt.show()

# Make sure that the 'date' column is properly formatted as datetime

data['Date'] = pd.to_datetime(data['Date'])

# Sort data in date order

data_sorted = data.sort_values(by='Date')

# visualize data between 'date' and 'closing price'

plt.figure(figsize=(14, 7))

plt.plot(data_sorted['Date'], data_sorted['Closing_price'], marker='.', linestyle='-',
linewidth=0.5, label='Closing Price')

plt.title('Vnindex closing price fluctuations over time')

plt.xlabel('Date')

plt.ylabel('Closing price')

plt.legend()

plt.show()

# visualize data between 'date', 'closing_price' and 'exchange_rate'

plt.figure(figsize=(14, 7))

plt.plot(data_sorted['Date'], data_sorted['Closing_price'], label='Closing Price',
color='blue')

plt.ylabel('Closing Price', color='blue')

plt.legend(loc='upper left')

# create secondary y-axis for exchange_rate

ax2 = plt.gca().twinx()
```

```
ax2.plot(data_sorted['Date'], data_sorted['Exchange_rate'], label='Exchange Rate', color='green')
ax2.set_ylabel('Exchange Rate', color='green')
ax2.legend(loc='upper right')
plt.title('Vnindex closing price fluctuations and exchange rate over time')
plt.xlabel('Date')
plt.show()
# visualize data between 'date', 'closing_price' and 'interest_rate'
plt.figure(figsize=(14, 7))
plt.plot(data_sorted['Date'], data_sorted['Closing_price'], label='Closing Price', color='blue')
plt.ylabel('Closing Price', color='blue')
plt.legend(loc='upper left')
# create secondary y-axis for 'interest_rate'
ax2 = plt.gca().twinx()
ax2.plot(data_sorted['Date'], data_sorted['Interest_rate'], label='Interest Rate', color='red')
ax2.set_ylabel('Interest Rate', color='red')
ax2.legend(loc='upper right')
plt.title('vnindex's closing price fluctuations and interest rates over time')
plt.xlabel('Ngày')
plt.show()
# visualize data between 'date', 'closing_price' and 'interest_rate'
plt.figure(figsize=(14, 7))
plt.plot(data_sorted['Date'], data_sorted['Closing_price'], label='Closing Price', color='blue')

plt.ylabel('Closing Price', color='blue')

plt.legend(loc='upper left')

# create secondary y-axis for 'interest_rate'

ax2 = plt.gca().twinx()

ax2.plot(data_sorted['Date'], data_sorted['XAU_USD'], label='XAU_USD', color='yellow')

ax2.set_ylabel('Interest Rate', color='yellow')

ax2.legend(loc='upper right')

plt.title('Vnindex closing price fluctuations and international gold prices over time')

plt.xlabel('Date')

plt.show()

#visualize data between 'date', 'closing_price' and 'interest_rate'

plt.figure(figsize=(14, 7))

plt.plot(data_sorted['Date'], data_sorted['Closing_price'], label='Closing Price', color='blue')

plt.ylabel('Closing Price', color='blue')

plt.legend(loc='upper left')

# create secondary y-axis for 'interest_rate'

ax2 = plt.gca().twinx()

ax2.plot(data_sorted['Date'], data_sorted['CPI'], label='CPI', color='red')

ax2.set_ylabel('Interest Rate', color='red')
```

```
ax2.legend(loc='upper right')

plt.title('Vnindex closing price fluctuations and CPI over time')

plt.xlabel('Date')

plt.show()

# visualize data between 'date', 'closing_price' and 'interest_rate'

plt.figure(figsize=(14, 7))

plt.plot(data_sorted['Date'], data_sorted['Closing_price'], label='Closing Price',
color='blue')

plt.ylabel('Closing Price', color='blue')

plt.legend(loc='upper left')

# create secondary y-axis for 'interest_rate'

ax2 = plt.gca().twinx()

ax2.plot(data_sorted['Date'], data_sorted['Money_supply_M2'],
label='Money_supply_M2', color='orange')

ax2.set_ylabel('Money_supply_M2', color='orange')

ax2.legend(loc='upper right')

plt.title('Vnindex closing price fluctuations and Money supply M2 over time')

plt.xlabel('Date')

plt.show()

# visualize data between 'date', 'closing_price' and 'interest_rate'

plt.plot(data_sorted['Date'], data_sorted['Closing_price'], label='Closing Price',
color='blue')

plt.ylabel('Closing Price', color='blue')

plt.legend(loc='upper left')
```

```
# create secondary y-axis for 'interest_rate'
ax2 = plt.gca().twinx()
ax2.plot(data_sorted['Date'], data_sorted['S&P_500'], label='S&P_500', color='green')
ax2.set_ylabel('S&P_500', color='green')
ax2.legend(loc='upper right')
plt.title('Vnindex closing price fluctuations and S&P_500 over time')
plt.xlabel('Date')
plt.show()
# visualize data between 'date', 'closing_price' and 'interest_rate'
plt.figure(figsize=(14, 7))
plt.plot(data_sorted['Date'], data_sorted['Closing_price'], label='Closing Price', color='blue')
plt.ylabel('Closing Price', color='blue')
plt.legend(loc='upper left')
# create secondary y-axis for 'interest_rate'
ax2 = plt.gca().twinx()
ax2.plot(data_sorted['Date'], data_sorted['Oil_price'], label='Oil_price', color='brown')
ax2.set_ylabel('Oil_price', color='brown')
ax2.legend(loc='upper right')
plt.title('Vnindex closing price fluctuations and Oil price over time')
plt.xlabel('Date')
plt.show()
```

```
from sklearn.model_selection import train_test_split

from sklearn.ensemble import RandomForestRegressor

from sklearn.linear_model import LinearRegression

from sklearn.tree import DecisionTreeRegressor

from sklearn.neighbors import KNeighborsRegressor

from sklearn.neural_network import MLPRegressor

from sklearn.svm import SVR

from xgboost import XGBRegressor

from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score

import numpy as np

# Preparing the data

X = data.drop(['Date', 'Closing_price'], axis=1)

y = data['Closing_price']

# Splitting the dataset into training and testing sets

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

from sklearn.ensemble import RandomForestRegressor

import matplotlib.pyplot as plt

# Re-training the Random Forest model with the entire dataset for feature importance analysis

rf_model = RandomForestRegressor()

rf_model.fit(X, y)

# Calculating feature importance

feature_importances = rf_model.feature_importances_
```

```
# Plotting feature importances

plt.figure(figsize=(10, 6))

plt.barh(y=X.columns, width=feature_importances)

plt.xlabel('Feature Importance')

plt.ylabel('Feature')

plt.title('Feature Importance - Random Forest Model')

plt.show()

import numpy as np

import matplotlib.pyplot as plt

from sklearn.model_selection import learning_curve

from sklearn.ensemble import RandomForestRegressor

# Assuming 'X' and 'y' are your features and target variable and have been defined earlier in your code.

# Computing learning curves with the scoring method adjusted to R^2 for better interpretation

train_sizes, train_scores, validation_scores = learning_curve(

    estimator=RandomForestRegressor(),

    X=X,

    y=y,

    train_sizes=np.linspace(0.1, 1.0, 5),

    cv=5,

    scoring='r2'

)

# Calculating the mean and standard deviation for training and validation scores
```

```
train_scores_mean = np.mean(train_scores, axis=1)
train_scores_std = np.std(train_scores, axis=1)
validation_scores_mean = np.mean(validation_scores, axis=1)
validation_scores_std = np.std(validation_scores, axis=1)
# Plotting the learning curves
plt.figure(figsize=(12, 6))
plt.fill_between(train_sizes, train_scores_mean - train_scores_std,
                 train_scores_mean + train_scores_std, alpha=0.1,
                 color="r")
plt.fill_between(train_sizes, validation_scores_mean - validation_scores_std,
                 validation_scores_mean + validation_scores_std, alpha=0.1, color="g")
plt.plot(train_sizes, train_scores_mean, 'o-', color="r",
         label="Training score")
plt.plot(train_sizes, validation_scores_mean, 'o-', color="g",
         label="Cross-validation score")
plt.title("Learning Curve (Random Forest)")
plt.xlabel("Training examples")
plt.ylabel("Score (R^2)")
plt.legend(loc="best")
plt.show()
from sklearn.model_selection import cross_val_score
import matplotlib.pyplot as plt
from sklearn.ensemble import RandomForestRegressor
import numpy as np
```

```
# Your code assumes that 'X' and 'y' are already defined as your feature set and target variable, respectively

# Recomputing cross-validation and explained variance scores for visualization

cv_scores_r2 = cross_val_score(RandomForestRegressor(), X, y, cv=5, scoring='r2')

explained_variance_scores = cross_val_score(RandomForestRegressor(), X, y, cv=5, scoring='explained_variance')

# Plotting the cross-validation R^2 scores

plt.figure(figsize=(14, 6))

plt.subplot(1, 2, 1)

plt.bar(range(1, 6), cv_scores_r2)

plt.xlabel('Fold')

plt.ylabel('R^2 Score')

plt.title('5-Fold Cross-Validation R^2 Scores')

# Plotting the explained variance scores

plt.subplot(1, 2, 2)

plt.bar(range(1, 6), explained_variance_scores)

plt.xlabel('Fold')

plt.ylabel('Explained Variance Score')

plt.title('5-Fold Cross-Validation Explained Variance Scores')

plt.tight_layout()

plt.show()
# Training the XGBoost model to calculate and visualize feature importance

xgb_model = XGBRegressor()

xgb_model.fit(X, y)

# Calculating feature importance from the XGBoost model
```

```
xgb_feature_importances = xgb_model.feature_importances_

# Plotting feature importances

plt.figure(figsize=(10, 6))

plt.barh(y=X.columns, width=xgb_feature_importances)

plt.xlabel('Feature Importance')

plt.ylabel('Feature')

plt.title('Feature Importance - XGBoost Model')

plt.show()

# Computing learning curves for the XGBoost model with the scoring method adjusted to r2

train_sizes_xgb, train_scores_xgb, validation_scores_xgb = learning_curve(

    estimator=XGBRegressor(),

    X=X,

    y=y,

    train_sizes=np.linspace(0.1, 1.0, 5),

    cv=5,

    scoring='r2'

)

# Calculating the mean and standard deviation for training and validation scores for XGBoost

train_scores_mean_xgb = np.mean(train_scores_xgb, axis=1)

train_scores_std_xgb = np.std(train_scores_xgb, axis=1)

validation_scores_mean_xgb = np.mean(validation_scores_xgb, axis=1)

validation_scores_std_xgb = np.std(validation_scores_xgb, axis=1)
```

```
# Plotting the learning curves for XGBoost
plt.figure(figsize=(12, 6))
plt.fill_between(train_sizes_xgb, train_scores_mean_xgb - train_scores_std_xgb,
                 train_scores_mean_xgb + train_scores_std_xgb, alpha=0.1,
                 color="r")
plt.fill_between(train_sizes_xgb, validation_scores_mean_xgb -
validation_scores_std_xgb,
                 validation_scores_mean_xgb + validation_scores_std_xgb, alpha=0.1,
color="g")
plt.plot(train_sizes_xgb, train_scores_mean_xgb, 'o-', color="r",
         label="Training score")
plt.plot(train_sizes_xgb, validation_scores_mean_xgb, 'o-', color="g",
         label="Cross-validation score")
plt.title("Learning Curve (XGBoost)")
plt.xlabel("Training examples")
plt.ylabel("Score (R^2)")
plt.legend(loc="best")
plt.show()
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import learning_curve, cross_val_score
from sklearn.ensemble import RandomForestRegressor
from xgboost import XGBRegressor
# Assuming X and y are already defined in your dataset
# Corrected code for learning_curve with RandomForestRegressor
train_sizes, train_scores, validation_scores = learning_curve(
    estimator=RandomForestRegressor(),
    X=X,
    y=y,
    train_sizes=np.linspace(0.1, 1.0, 5),
    cv=5,
    scoring='r2'
)
# Now calculating and plotting for the XGBoost model without errors
cv_scores_r2_xgb = cross_val_score(XGBRegressor(), X, y, cv=5, scoring='r2')
explained_variance_scores_xgb = cross_val_score(XGBRegressor(), X, y, cv=5, scoring='explained_variance')
# Corrected plotting code for XGBoost results
plt.figure(figsize=(14, 6))
# Plot for cross-validation R^2 scores
plt.subplot(1, 2, 1)
plt.bar(range(1, 6), cv_scores_r2_xgb)
plt.xlabel('Fold')
plt.ylabel('R^2 Score')
plt.title('5-Fold Cross-Validation R^2 Scores for XGBoost')
# Plot for explained variance scores
plt.subplot(1, 2, 2)
plt.bar(range(1, 6), explained_variance_scores_xgb)
plt.xlabel('Fold')

plt.ylabel('Explained Variance Score')

plt.title('5-Fold Cross-Validation Explained Variance Scores for XGBoost')

plt.tight_layout()

plt.show()

# Training the Linear Regression model

lr_model = LinearRegression()

lr_model.fit(X, y)

# Extracting coefficients

lr_coefficients = lr_model.coef_

# Plotting the coefficients as feature importance

plt.figure(figsize=(10, 6))

plt.barh(y=X.columns, width=lr_coefficients)

plt.xlabel('Coefficient Value')

plt.ylabel('Feature')

plt.title('Feature Importance (Coefficients) - Linear Regression Model')

plt.show()

# Computing learning curves for the Linear Regression model with the scoring method adjusted to r2

train_sizes_lr, train_scores_lr, validation_scores_lr = learning_curve(

    estimator=LinearRegression(),

    X=X,

    y=y,

    train_sizes=np.linspace(0.1, 1.0, 5),
    cv=5,
    scoring='r2'
)
# Calculating the mean and standard deviation for training and validation scores for Linear Regression
train_scores_mean_lr = np.mean(train_scores_lr, axis=1)
train_scores_std_lr = np.std(train_scores_lr, axis=1)
validation_scores_mean_lr = np.mean(validation_scores_lr, axis=1)
validation_scores_std_lr = np.std(validation_scores_lr, axis=1)
# Plotting the learning curves for Linear Regression
plt.figure(figsize=(12, 6))
plt.fill_between(train_sizes_lr, train_scores_mean_lr - train_scores_std_lr,
                 train_scores_mean_lr + train_scores_std_lr, alpha=0.1,
                 color="r")
plt.fill_between(train_sizes_lr, validation_scores_mean_lr - validation_scores_std_lr,
                 validation_scores_mean_lr + validation_scores_std_lr, alpha=0.1, color="g")
plt.plot(train_sizes_lr, train_scores_mean_lr, 'o-', color="r",
         label="Training score")
plt.plot(train_sizes_lr, validation_scores_mean_lr, 'o-', color="g",
         label="Cross-validation score")
plt.title("Learning Curve (Linear Regression)")
plt.xlabel("Training examples")
plt.ylabel("Score (R^2)")
plt.legend(loc="best")
plt.show()
# Calculating cross-validation R^2 scores for the Linear Regression model
cv_scores_r2_lr = cross_val_score(LinearRegression(), X, y, cv=5, scoring='r2'
# Calculating explained variance scores for the Linear Regression model
explained_variance_scores_lr = cross_val_score(LinearRegression(), X, y, cv=5, scoring='explained_variance')
# Plotting the results for visualization
plt.figure(figsize=(14, 6))
# Plotting the cross-validation R^2 scores for Linear Regression
plt.subplot(1, 2, 1)
plt.bar(range(1, 6), cv_scores_r2_lr)
plt.xlabel('Fold')
plt.ylabel('R^2 Score')
plt.title('5-Fold Cross-Validation R^2 Scores for Linear Regression')
# Plotting the explained variance scores for Linear Regression
plt.subplot(1, 2, 2)
plt.bar(range(1, 6), explained_variance_scores_lr)
plt.xlabel('Fold')
plt.ylabel('Explained Variance Score')
plt.title('5-Fold Cross-Validation Explained Variance Scores for Linear Regression')
plt.tight_layout()
plt.show()
# Training the Decision Tree Regressor model
```

```
dt_model = DecisionTreeRegressor()

dt_model.fit(X, y)

# Calculating feature importance from the Decision Tree model

dt_feature_importances = dt_model.feature_importances_

# Plotting feature importances

plt.figure(figsize=(10, 6))

plt.barh(y=X.columns, width=dt_feature_importances)

plt.xlabel('Feature Importance')

plt.ylabel('Feature')

plt.title('Feature Importance - Decision Tree Regressor Model')

plt.show()

# Computing learning curves for the Decision Tree Regressor model with the scoring method adjusted to r2

train_sizes_dt, train_scores_dt, validation_scores_dt = learning_curve(

    estimator=DecisionTreeRegressor(),

    X=X,

    y=y,

    train_sizes=np.linspace(0.1, 1.0, 5),

    cv=5,

    scoring='r2'

)

# Calculating the mean and standard deviation for training and validation scores for Decision Tree Regressor

train_scores_mean_dt = np.mean(train_scores_dt, axis=1)
```

```
train_scores_std_dt = np.std(train_scores_dt, axis=1)

validation_scores_mean_dt = np.mean(validation_scores_dt, axis=1)

validation_scores_std_dt = np.std(validation_scores_dt, axis=1)

# Plotting the learning curves for Decision Tree Regressor

plt.figure(figsize=(12, 6))

plt.fill_between(train_sizes_dt, train_scores_mean_dt - train_scores_std_dt,

                 train_scores_mean_dt + train_scores_std_dt, alpha=0.1,

                 color="r")

plt.fill_between(train_sizes_dt, validation_scores_mean_dt - validation_scores_std_dt,

                 validation_scores_mean_dt + validation_scores_std_dt, alpha=0.1,
color="g")

plt.plot(train_sizes_dt, train_scores_mean_dt, 'o-', color="r",

         label="Training score")

plt.plot(train_sizes_dt, validation_scores_mean_dt, 'o-', color="g",

         label="Cross-validation score")

plt.title("Learning Curve (Decision Tree Regressor)")

plt.xlabel("Training examples")

plt.ylabel("Score (R^2)")

plt.legend(loc="best")

plt.show(

# Calculating cross-validation R^2 scores for the Decision Tree Regressor model

cv_scores_r2_dt = cross_val_score(DecisionTreeRegressor(), X, y, cv=5, scoring='r2')

# Calculating explained variance scores for the Decision Tree Regressor model
```

```
explained_variance_scores_dt = cross_val_score(DecisionTreeRegressor(), X, y, cv=5, scoring='explained_variance')
# Plotting the results for visualization
plt.figure(figsize=(14, 6))
# Plotting the cross-validation R^2 scores for Decision Tree Regressor
plt.subplot(1, 2, 1)
plt.bar(range(1, 6), cv_scores_r2_dt)
plt.xlabel('Fold')
plt.ylabel('R^2 Score')
plt.title('5-Fold Cross-Validation R^2 Scores for Decision Tree Regressor')
# Plotting the explained variance scores for Decision Tree Regressor
plt.subplot(1, 2, 2)
plt.bar(range(1, 6), explained_variance_scores_dt)
plt.xlabel('Fold')
plt.ylabel('Explained Variance Score')
plt.title('5-Fold Cross-Validation Explained Variance Scores for Decision Tree Regressor')
plt.tight_layout()
plt.show(
# Computing learning curves for the SVM model with the scoring method adjusted to r2
train_sizes_svm, train_scores_svm, validation_scores_svm = learning_curve(
    estimator=SVR(),
    X=X,
    y=y,
    train_sizes=np.linspace(0.1, 1.0, 5),
    cv=5,
    scoring='r2'
)
# Calculating the mean and standard deviation for training and validation scores for
SVM
train_scores_mean_svm = np.mean(train_scores_svm, axis=1)
train_scores_std_svm = np.std(train_scores_svm, axis=1)
validation_scores_mean_svm = np.mean(validation_scores_svm, axis=1)
validation_scores_std_svm = np.std(validation_scores_svm, axis=1)
# Plotting the learning curves for SVM
plt.figure(figsize=(12, 6))
plt.fill_between(train_sizes_svm, train_scores_mean_svm - train_scores_std_svm,
                 train_scores_mean_svm + train_scores_std_svm, alpha=0.1,
                 color="r")
plt.fill_between(train_sizes_svm, validation_scores_mean_svm -
validation_scores_std_svm,
                 validation_scores_mean_svm + validation_scores_std_svm, alpha=0.1,
color="g")
plt.plot(train_sizes_svm, train_scores_mean_svm, 'o-', color="r",
         label="Training score")
plt.plot(train_sizes_svm, validation_scores_mean_svm, 'o-', color="g",
         label="Cross-validation score")
plt.title("Learning Curve (SVM)")
plt.xlabel("Training examples")
plt.ylabel("Score (R^2)")

plt.legend(loc="best")

plt.show()

# Calculating cross-validation R^2 scores for the SVM model

cv_scores_r2_svm = cross_val_score(SVR(), X, y, cv=5, scoring='r2'

# Calculating explained variance scores for the SVM model

explained_variance_scores_svm = cross_val_score(SVR(), X, y, cv=5, scoring='explained_variance')

# Plotting the results for visualization

plt.figure(figsize=(14, 6))

# Plotting the cross-validation R^2 scores for SVM

plt.subplot(1, 2, 1)

plt.bar(range(1, 6), cv_scores_r2_svm)

plt.xlabel('Fold')

plt.ylabel('R^2 Score')

plt.title('5-Fold Cross-Validation R^2 Scores for SVM')

# Plotting the explained variance scores for SVM

plt.subplot(1, 2, 2)

plt.bar(range(1, 6), explained_variance_scores_svm)

plt.xlabel('Fold')

plt.ylabel('Explained Variance Score')

plt.title('5-Fold Cross-Validation Explained Variance Scores for SVM'

plt.tight_layout()

plt.show()
```

```
# Computing learning curves for the KNN model with the scoring method adjusted to
r2
train_sizes_knn, train_scores_knn, validation_scores_knn = learning_curve(
    estimator=KNeighborsRegressor(),
    X=X,
    y=y,
    train_sizes=np.linspace(0.1, 1.0, 5),
    cv=5,
    scoring='r2'
)
# Calculating the mean and standard deviation for training and validation scores for
KNN
train_scores_mean_knn = np.mean(train_scores_knn, axis=1)
train_scores_std_knn = np.std(train_scores_knn, axis=1)
validation_scores_mean_knn = np.mean(validation_scores_knn, axis=1)
validation_scores_std_knn = np.std(validation_scores_knn, axis=1)
# Plotting the learning curves for KNN
plt.figure(figsize=(12, 6))
plt.fill_between(train_sizes_knn, train_scores_mean_knn - train_scores_std_knn,
                 train_scores_mean_knn + train_scores_std_knn, alpha=0.1,
                 color="r")
plt.fill_between(train_sizes_knn, validation_scores_mean_knn -
validation_scores_std_knn,
                 validation_scores_mean_knn + validation_scores_std_knn, alpha=0.1,
color="g")
```

```
plt.plot(train_sizes_knn, train_scores_mean_knn, 'o-', color="r",

         label="Training score")

plt.plot(train_sizes_knn, validation_scores_mean_knn, 'o-', color="g",

         label="Cross-validation score")

plt.title("Learning Curve (KNN)")

plt.xlabel("Training examples")

plt.ylabel("Score (R^2)")

plt.legend(loc="best")

plt.show()

# Calculating cross-validation R^2 scores for the KNN model

cv_scores_r2_knn = cross_val_score(KNeighborsRegressor(), X, y, cv=5, scoring='r2')

# Calculating explained variance scores for the KNN model

explained_variance_scores_knn = cross_val_score(KNeighborsRegressor(), X, y,
cv=5, scoring='explained_variance')

# Plotting the results for visualization

plt.figure(figsize=(14, 6))

# Plotting the cross-validation R^2 scores for KNN

plt.subplot(1, 2, 1)

plt.bar(range(1, 6), cv_scores_r2_knn)

plt.xlabel('Fold')

plt.ylabel('R^2 Score')

plt.title('5-Fold Cross-Validation R^2 Scores for KNN')

# Plotting the explained variance scores for KNN

plt.subplot(1, 2, 2)
plt.bar(range(1, 6), explained_variance_scores_knn)

plt.xlabel('Fold')

plt.ylabel('Explained Variance Score')

plt.title('5-Fold Cross-Validation Explained Variance Scores for KNN')

plt.tight_layout()

plt.show()

# Computing learning curves for the ANN model with the scoring method adjusted to r2

train_sizes_ann, train_scores_ann, validation_scores_ann = learning_curve(

    estimator=MLPRegressor(max_iter=1000),

    X=X,

    y=y,

    train_sizes=np.linspace(0.1, 1.0, 5),

    cv=5,

    scoring='r2'

)

# Calculating the mean and standard deviation for training and validation scores for ANN

train_scores_mean_ann = np.mean(train_scores_ann, axis=1)

train_scores_std_ann = np.std(train_scores_ann, axis=1)

validation_scores_mean_ann = np.mean(validation_scores_ann, axis=1)

validation_scores_std_ann = np.std(validation_scores_ann, axis=1)

# Plotting the learning curves for ANN

plt.figure(figsize=(12, 6))
plt.fill_between(train_sizes_ann, train_scores_mean_ann - train_scores_std_ann,
                 train_scores_mean_ann + train_scores_std_ann, alpha=0.1,
                 color="r")
plt.fill_between(train_sizes_ann, validation_scores_mean_ann -
validation_scores_std_ann,
                 validation_scores_mean_ann + validation_scores_std_ann, alpha=0.1,
color="g")
plt.plot(train_sizes_ann, train_scores_mean_ann, 'o-', color="r",
         label="Training score")
plt.plot(train_sizes_ann, validation_scores_mean_ann, 'o-', color="g",
         label="Cross-validation score")
plt.title("Learning Curve (ANN)")
plt.xlabel("Training examples")
plt.ylabel("Score (R^2)")
plt.legend(loc="best")
plt.show()
# Calculating cross-validation R^2 scores for the ANN model
cv_scores_r2_ann = cross_val_score(MLPRegressor(max_iter=1000), X, y, cv=5,
scoring='r2', n_jobs=-1)
# Calculating explained variance scores for the ANN model
explained_variance_scores_ann = cross_val_score(MLPRegressor(max_iter=1000), X,
y, cv=5, scoring='explained_variance', n_jobs=-1)
# Plotting the results for visualization
plt.figure(figsize=(14, 6))
# Plotting the cross-validation R^2 scores for ANN
```

```
plt.subplot(1, 2, 1)
plt.bar(range(1, 6), cv_scores_r2_ann)
plt.xlabel('Fold')
plt.ylabel('R^2 Score')
plt.title('5-Fold Cross-Validation R^2 Scores for ANN')
# Plotting the explained variance scores for ANN
plt.subplot(1, 2, 2)
plt.bar(range(1, 6), explained_variance_scores_ann)
plt.xlabel('Fold')
plt.ylabel('Explained Variance Score')
plt.title('5-Fold Cross-Validation Explained Variance Scores for ANN')
plt.tight_layout()
plt.show()
# Defining the models
models = {
    "Random Forest": RandomForestRegressor(),
    "XGBoost": XGBRegressor(),
    "Linear Regression": LinearRegression(),
    "Decision Tree": DecisionTreeRegressor(),
    "KNN": KNeighborsRegressor(),
    "ANN": MLPRegressor(max_iter=1000),
    "SVM": SVR()
}
# Training the models and making predictions
```

```
results = []

for name, model in models.items():

    model.fit(X_train, y_train)

    predictions = model.predict(X_test)

    mse = mean_squared_error(y_test, predictions)

    rmse = np.sqrt(mse)

    mae = mean_absolute_error(y_test, predictions)

    mape = np.mean(np.abs((y_test - predictions) / y_test)) * 100

    r2 = r2_score(y_test, predictions)

    results.append({

        "Model": name,

        "MSE": mse,

        "RMSE": rmse,

        "MAE": mae,

        "MAPE": mape,

        "R-Squared": r2

    })

# Creating a DataFrame to display the results

results_df = pd.DataFrame(results)

results_df
!pip install shap

import shap

from sklearn.ensemble import RandomForestRegressor

# train the random forest model
```

```
model = RandomForestRegressor()

model.fit(X_train, y_train)

# initialize shape explainer

explainer = shap.TreeExplainer(model)

# calculate shape values for x_test

shap_values = explainer.shap_values(X_test)

# Visualize the importance of features

shap.summary_plot(shap_values, X_test, plot_type="bar")

from xgboost import XGBRegressor

xgb_model = XGBRegressor()

xgb_model.fit(X_train, y_train

import shap

# initialize shape explainer

explainer = shap.Explainer(xgb_model)

# calculate shape values for x_test

shap_values = explainer(X_test)

# Visualize an overview of the importance of features

shap.summary_plot(shap_values, X_test)

# Visualize the importance of features as a bar chart

from sklearn.tree import DecisionTreeRegressor

# Let's say x_train, y_train are your training data

dt_model = DecisionTreeRegressor()

dt_model.fit(X_train, y_train)

import shap
```

```
# initialize shape explainer
explainer = shap.Explainer(dt_model)
# calculate shape values for x_test
shap_values = explainer(X_test)
# Visualize an overview of the importance of features
shap.summary_plot(shap_values, X_test)
# Visualize the importance of features as a bar chart
shap.plots.bar(shap_values)
from sklearn.linear_model import LinearRegression
lr_model = LinearRegression()
lr_model.fit(X_train, y_train)
import shap
explainer = shap.Explainer(lr_model, X_train)
shap_values = explainer(X_test)
shap.summary_plot(shap_values, X_test)
shap.plots.bar(shap_values)
from sklearn.neural_network import MLPRegressor
ann_model = MLPRegressor(hidden_layer_sizes=(100,), max_iter=1000)
ann_model.fit(X_train, y_train)
import shap
explainer = shap.KernelExplainer(ann_model.predict, shap.sample(X_train, 100))
shap_values = explainer.shap_values(shap.sample(X_test, 100))
shap.summary_plot(shap_values, shap.sample(X_test, 100))
from sklearn.neighbors import KNeighborsRegressor
```

```
knn_model = KNeighborsRegressor(n_neighbors=5)
knn_model.fit(X_train, y_train)
import shap
explainer = shap.KernelExplainer(knn_model.predict, shap.sample(X_train, 100))
shap_values = explainer.shap_values(shap.sample(X_test, 10))
shap.summary_plot(shap_values, shap.sample(X_test, 10))
from sklearn.svm import SVR
import shap
import numpy as np
svm_model = SVR()
svm_model.fit(X_train, y_train)
background_data = shap.utils.sample(X_train, 100)
explainer = shap.KernelExplainer(svm_model.predict, background_data)
shap_values = explainer.shap_values(X_test, nsamples=100)
expected_value = np.array([explainer.expected_value]) if
isinstance(explainer.expected_value, float) else explainer.expected_value
shap_explanation = shap.Explanation(values=shap_values,
base_values=expected_value, data=X_test)
shap.summary_plot(shap_explanation, X_test)
shap.plots.bar(shap_explanation)
```

REFERENCES

Vietnamese References

1. Trương, T. T. D., & Lê, H. T. (2023). Ứng dụng phương pháp học máy trong dự báo rủi ro phá sản của các doanh nghiệp Việt Nam. *Học viện Ngân hàng*. DOI: 10.33301/JED.VI.1066
2. Nguyen, K. M., & Nguyen, D. V. (2013). Quan hệ giữa các yếu tố kinh tế vĩ mô và biến động thị trường chứng khoán: Bằng chứng nghiên cứu từ thị trường Việt Nam [Relationship between macroeconomic factors and stock market volatility: Research evidence from Vietnam market]. *Tạp chí Phát triển Khoa học và Công nghệ*, 16(3Q), 86-100.
3. Truong, L. D. (2014). Các nhân tố ảnh hưởng đến sự thay đổi giá của cổ phiếu: Các bằng chứng từ Sở Giao dịch Chứng khoán Thành Phố Hồ Chí Minh [Factors influencing stock price changes: Evidence from Ho Chi Minh City Stock Exchange]. *Tạp chí Khoa học Trường Đại học Cần Thơ,* 72-78.
4. Tran, H. H. (2017). Tác động của giá dầu thế giới đến thị trường chứng khoán và các biến vĩ mô của nền kinh tế: Trường hợp Việt Nam [The impact of world oil prices on the stock market and macroeconomic variables: The case of Vietnam]. Retrieved July 20, 2019, from *Tạp chí Công thương* website: http://tapchicongthuong.vn/bai-viet/tac-dong-cuagia-dau-the-gioi-den-thi-truong-chung-khoan-va-cac-bien-vi-mo-trong-nen-kinh-tetruong-hop-viet-nam-51084.htm
5. Dinh, L. T. T., & Nguyen, T. T. T. (2008). Tác động của tỷ giá, bất động sản, giá vàng lên thị trường chứng khoán Việt Nam thời gian qua [The impact of exchange rates, real estate, and gold prices on Vietnam's stock market recently]. *Tạp chí Ngân hàng*, 17, 26-30.

English References

1. Dao, H. T., Vu, L. H., Pham, T. L., & Nguyen, K. T. (2022). Macro-Economic Factors Affecting the Vietnam Stock Price Index: An Application of the ARDL Model. *The Journal of Asian Finance, Economics and Business*, *9*(5), 285-294.
2. Nguyen, T. D., Bui, Q. H., & Nguyen, T. T. (2016). Causal Correlation between Exchange Rate and Stock Index: Evidence from VN-Index. *Asian Social Science*, 12(8), 43.

3. Vuong, Q. D. (2016). The Impact of Macroeconomic Factors on Stock Price Index, VN-Index. *IJISET - International Journal of Innovative Science, Engineering & Technology*, 3(7). ISSN (Online) 2348 – 7968.
4. Tran, K. L., Le, H. A., Lieu, C. P., & Nguyen, D. T. (2023). Machine Learning to Forecast Financial Bubbles in Stock Markets: Evidence from Vietnam. *International Journal of Financial Studies*, 11(4), 133.
5. Abdalla, I. S., & Murinde, V. (1997). Exchange rate and stock price interactions in emerging financial markets: Evidence on India, Korea, Pakistan and the Philippines. *Applied Financial Economics*, 7(1), 25-35.
6. Le, T. M. H., Jian, Z., & Zhican, Z. (2019). Impact of Macroeconomic Variables on Stock Price Index: Evidence from Vietnam Stock Market. *Research Journal of Finance and Accounting*, 10(12).
7. Sangsawaia, N., & Sutivong, D. (2023). Analyzing Impact of Economic Indicators on Vietnam Stock Market Using Machine Learning Techniques. In L.-C. Tang (Ed.), *Industrial Engineering and Applications* (Vol. 230054).
8. Sangsawai, Nuttawan, "Analyzing impact of economic indicators on Vietnam stock market with machine learning techniques" (2022). Chulalongkorn University Theses and Dissertations (Chula ETD). 5905.
9. Geetha, C., Mohidin, R., Chandran, V. V., & Chong, V. (2011). The relationship between inflation and stock market: Evidence from Malaysia, United States and China. International *Journal of Economics and Management Sciences*, 1(2), 1-16.
10. González, M., Nave, J., & Rubio, G. (2018). Macroeconomic determinants of stock market betas. *Journal of Empirical Finance*, 45, 26-44.
11. Ho, L.-C., & Huang, C.-H. (2015). The nonlinear relationships between stock indexes and exchange rates. *Japan and the World Economy*, 33, 20-27.
12. Smyth, R., & Narayan, P. K. (2018). What do we know about oil prices and stock returns? *International Review of Financial Analysis*, 57, 148-156
13. Rjoub, H. (2012). Stock prices and exchange rates dynamics: Evidence from emerging markets. *African Journal of Business Management*, 6(13), 4728.
14. Maysami, R. C., & Koh, T. S. (2000). A vector error correction model of the Singapore stock market. *International Review of Economics & Finance*, 9(1), 79-96.
15. Barboza, Flavio, Herbert Kimura, and Edward Altman. 2017. Machine learning models and bankruptcy prediction. *Expert Systems with Applications* 83: 405–17
16. Beutel, Johannes, Sophia List, and Gregor von Schweinitz. 2019. Does machine learning help us predict banking crises? Journal of Financial Stability 45: 100693.

17. Breiman, Leo. 2001. Random forests. Machine learning 45: 5–32.

18. Chatzis, Sotirios P., Vassilis Siakoulis, Anastasios Petropoulos, Evangelos Stavroulakis, and Nikos Vlachogiannakis. 2018. Forecasting stock market crisis events using deep and statistical machine learning techniques. *Expert Systems with Applications* 112: 353–71.

19. McKinney, Wes. 2010. Data structures for statistical computing in python. Paper presented at the 9th Python in Science Conference, Austin, TX, USA, June 28–July 3.

20. Başoğlu Kabran, Fatma, and Kamil Demirberk Ünlü. 2021. A two-step machine learning approach to predict S&P 500 bubbles. *Journal of Applied Statistics* 48: 2776–94.

21. Fuster, Andreas, Paul Goldsmith-Pinkham, Tarun Ramadorai, and Ansgar Walther. 2022. Predictably unequal. The Effects of Machine Learning on Credit Markets. *Journal of Finance* 77: 1–808.

22. Richards, N., Simpson, J., & Evans, J. (2009). The interaction between exchange rates and stock prices: An Australian context. *International Journal of Economics and Finance*, 1(1), 3-23.

23. Sujit, K. S., & Kumar, B. R. (2011). Study on the dynamic relationship among gold price, oil price, exchange rate, and stock market returns. *International Journal of Applied Business and Economic Research*, 9(2), 145-165.

24. Adam, A. M., Tweneboah, G. (2008). Macroeconomic factors and stock market movement: Evidence from Ghana. *SSRN Journal*, 1, 79-86.

25. Alam, M. M., & Uddin, M. G. S. (2009). Relationship between interest rate and stock price: Empirical evidence from developed and developing countries. *International Journal of Business and Management*, 4(3), 43-51.

26. Alqattan, A. A., & Alhayky, A. (2016). Impact of oil prices on stock markets: Evidence from Gulf Cooperation Council (GCC) financial markets. *Amity Journal of Finance*, 1(1), 1-8.

27. Asprem, M. (1989). Stock prices, asset portfolios, and macroeconomic variables in ten European countries. *Journal of Banking and Finance*, 13(4-5), 589-612.

28. Degiannakis, S., Filis, G., & Arora, V. (2018). Oil prices and stock markets: A review of the theory and empirical evidence. *Energy Journal*, 39(1).

29. Geetha, C., Mohidin, R., Chandran, V. V., & Chong, V. (2011). The relationship between inflation and stock market: Evidence from Malaysia, United States, and China. *International Journal of Economics and Management Sciences*, 1(2), 1-16.

30. Gonzalez, M., Nave, J., & Rubio, G. (2018). Macroeconomic determinants of stock market betas. *Journal of Empirical Finance*, 45, 26-44.

31. Kieu, N. M., & Diep, V. N. (2013). Relationship between macroeconomic factors and the stock market fluctuations: Empirical evidence from the Vietnam stock market, *Science and Technology Development Journal*, 16(3), 86-100.

32. Lee, J. W., & Brahmasrene, T. (2018). An exploration of dynamical relationships between macroeconomic variables and stock prices in Korea. *Journal of Asian Finance, Economics, and Business*, 5(3), 7-17.

33. Lee, J. W., & Brahmasrene, T. (2020). An exploration of dynamical relationships between macroeconomic variables and stock prices in Korea revisited. *Journal of Asian Finance, Economics, and Business*, 7(10), 23-34.

34. Mukherjee, T. K., & Naka, A. (1995). Dynamic relations between macroeconomic variables and the Japanese stock market: An application of a vector error correction model. *Journal of Financial Research*, 18(2), 223-237.

35. Nandha, M., & Singh, H. (2011). Short-run and long-run oil price sensitivity of Chinese stocks. *Corporate Ownership and Control*, 8, 163-169.

36. Nhu Quynh, N. T., & Huong Linh, V. T. (2019). Impact of some macroeconomic factors on the stock price index in Vietnam. Journal of Science of Ho Chi Minh City Open University - *Economics and Business Administration*, 14(3), 47-63.

37. Obstfeld, M., & Rogoff, K. (1995). Exchange rate dynamics redux. *Journal of Political Economy*, 103(3), 624-660.

38. Phong, L. H. & Bach Van, D. T. (2015). Verifying the impact of macro factors on the Vietnam stock index by the ARDL model. *Journal of Development and Integration*, 20(30), 61-66.

39. Rahman, A. A., Sidek, N. Z. M., & Tafri, F. H. (2009). Macroeconomic determinants of Malaysian stock market. *African Journal of Business Management*, 3(3), 095-106.

40. Trinh, T. P. T, & Linh Dan, V. L. (2020). Asymmetric impact of oil price fluctuations on the Vietnam stock market. Journal of Asian Finance, Economics, and Business, 8(6), 553-562.

41. Hussainey, K. and Khanh Ngoc, L. (2009) ‘The impact of macroeconomic indicators on Vietnamese stock prices’, *Journal of Risk Finance*, 10(4), pp. 321–332.