INSTITUTO
UNIVERSITÁRIO
DE LISBOA

Explaining the S&P 500: How does certain commodities affect the index

Mailson Manuel Teixeira Varela

Master in Data Science

Supervisor:
Professor Ph.D, Diana E. Aldea Mendes, Associate Professor
Iscte - University Institute of Lisbon

September, 2023

BUSINESS SCHOOL

TECNOLOGIAS E ARQUITETURA

Department of Quantitative Methods for Management and Economics

Department of Information Science and Technology

Explaining the S&P 500: How does certain commodities affect the index

Mailson Manuel Teixeira Varela

Master in Data Science

Supervisor:
Professor Ph.D, Diana E. Aldea Mendes, Associate Professor
Department of Quantitative Methods,
Iscte - University Institute of Lisbon

September, 2023

Acknowledgment

I would like to express my sincere gratitude to all those who have contributed to the completion of this work. This journey has been both challenging and rewarding, and I couldn't have reached this milestone without the support and assistance of many individuals.

First and foremost, I extend my deepest appreciation to my thesis advisor, Diana Aldea Mendes, for their invaluable guidance, patience, and unwavering support throughout this research endeavor. Their expertise, insightful feedback, and dedication to my academic growth have been instrumental in shaping the quality of this thesis and I truly appreciate it.

I want to acknowledge the Instituto Universitário de Lisboa faculty and staff who have provided a conducive academic environment during my time here. Your dedication to fostering a culture of learning has been truly inspiring.

I extend my heartfelt gratitude to three remarkable teachers who have profoundly influenced my academic journey. Indira Brito, with their passion and unique way of teaching, sparked my interest and served as the reason I pursued a master's in data science. Carlos Araújo's unwavering belief in my abilities, and Elton Alves' guidance, have been instrumental in my decision to pursue higher education and have played a pivotal role in my academic growth. I am deeply thankful for their inspiration and mentorship.

Furthermore, I am indebted to my family and friends, especially my mother and father, for their unwavering belief in my abilities and their continuous encouragement throughout my educational journey. Their sacrifices and love have been my constant motivation.

Lastly, I wish to acknowledge the countless researchers and authors whose work laid the foundation for my own research. Their insights and discoveries have been essential in shaping the direction of this thesis.

Abstract

In recent years, the proliferation of Artificial Intelligence (A.I.) has revolutionized decision-making processes across various domains. Deep Learning algorithms, particularly LSTM and XGBoost models, have emerged as powerful tools for accurate predictions in complex contexts, such as financial markets. However, the inherent challenge of interpreting these models has led to a tradeoff between accuracy and transparency. The need for Explainable Artificial Intelligence (XAI) becomes paramount in critical domains like finance, where understanding the model's reasoning is crucial for informed decision-making.

Our study comprises two fundamental phases: model development and explanation. The initial phase focuses on crafting LSTM and XGBoost models, fine-tuning their hyperparameters, and optimizing their predictive performance for S&P 500 index forecasting. Rigorous evaluation metrics, encompassing MAE, MSE, MAPE and RMSE, guide our pursuit of accurate predictions. Dow Jones emerges as one of the most influential variables in forecasting S&P 500 along with Bitcoin, which, interestingly, wields a consistently negative impact in both models, penalizing both performance with its effects and unveiling its unique role. Our findings inform decision-making in finance, advocating for transparency and advancing predictive models and interpretability.

Key-Words: Machine Learning, Financial Market, Explainability, SHAP, LIME.

Resumo

Nos últimos anos, a proliferação da Inteligência Artificial (IA) revolucionou os processos de tomada de decisão em vários domínios. Algoritmos de Aprendizagem Profunda, particularmente modelos LSTM e XGBoost, emergiram como ferramentas poderosas para previsões precisas em contextos complexos, como os mercados financeiros. No entanto, o desafio inerente de interpretar esses modelos levou a um equilíbrio entre precisão e transparência. A necessidade de Inteligência Artificial Explicável (XAI) torna-se fundamental em domínios críticos, como finanças, onde entender o raciocínio do modelo é crucial para a tomada de decisões informadas.

Nosso estudo compreende duas fases fundamentais: desenvolvimento do modelo e explicação. A fase inicial concentra-se na criação de modelos LSTM e XGBoost, ajuste de seus hiperparâmetros e otimização do desempenho preditivo para a previsão do índice S&P 500. Métricas rigorosas de avaliação, incluindo MAE, MSE, MAPE e RMSE, orientam nossa busca por previsões precisas. O índice Dow Jones emerge como uma das variáveis mais influentes na previsão do S&P 500, juntamente com o Bitcoin, que, interessantemente, exerce um impacto consistentemente negativo em ambos os modelos, penalizando o desempenho com seus efeitos e revelando seu papel único. Nossas descobertas informam a tomada de decisões financeiras, advogando pela transparência e promovendo modelos preditivos e interpretabilidade avançados.

Palavras-chave: Machine Learning, Financial Market, Explainability, SHAP, LIME.

Contents

CHAPTER 1

Introduction

In the era of big data and Industry 4.0, Artificial Intelligence (A.I.) has emerged as a driving force behind industrial development and the integration of advanced technologies. Technologies such as graphic processing units, the Internet of Things (IoT), cloud computing, and blockchain have further propelled the application of A.I. in diverse systems (Lu, 2019). All this has sparked increased attention and investment in A.I., with the worldwide revenues for the A.I. market projected to exceed $432.8 billion by 2022, and a market size expected to surpass $500 billion in 2023 (International Data Corporation). As the A.I. market continues to expand, reaching an estimated $997.77 billion by 2028 (Plat A.I.[1]), the development and implementation of A.I. systems, particularly machine learning models, are also accelerating (Rai, 2020).

The rapid adoption of A.I. and Machine Learning (ML) in business decision-making has heightened the demand for XAI (Bhattacharya, 2022). Organizations and regulatory bodies recognize the need to demystify black-box algorithms, ensuring transparency, ethics, trust, and mitigating bias. Initiatives such as the European General Data Protection Regulation (GDPR) underscore the importance of A.I. ethics and the potential risks associated with biased or unexplainable A.I. systems (Das, 2020).

When it comes to A.I. and ML, to define the concepts of interpretability and explainability, there isn't yet a formal and concise explanation of what it is. Many authors have attempted to define what XAI is, and perhaps D. Gunning (Arrieta *et al.*, 2020) had the closest idea:

> "XAI will create a suite of machine learning techniques that enables human users to understand, appropriately trust, and effectively manage the emerging generation of artificially intelligent partners".

Adadi *et al*. defined XAI in 2018, stating that "XAI is a research field that aims to make A.I. systems results more understandable to humans". Simply put, both definitions don't fail to highlight that XAI's primary focus is to solidify the thrust of both non-technical and technical users further toward the system. Although they are direct and practical definitions, both of these fall under the problem pointed out by Arrieta *et al*. (2020), which is the lack of characterization of the term in question, ignoring critical aspects such as its mission. Still, the solution to that problem is beyond the scope of this work.

[1] https://plat.ai/blog/artificial-intelligence-market-size-and-growth/

As stated in the beginning, one of the main downsides of Artificial Intelligence and Machine Learning is their lack of transparency – therefore, the term "black box" associated with it – keeping them away from a broader adaptation, especially in critical fields where the "whys" are empirical. According to 56% of executives, the transparency and ethics of an A.I. system are important when it comes to the well-being of a successful business, and 41% of companies reported having to suspend the deployment of their A.I. systems due to potential ethical issues reflected by the lack of transparency of these systems (Dhinakaran, 2021).

The inability to interpret the judgments produced by cutting-edge ML and A.I. systems is caused by two factors: the gap between the research community and the business sectors; and the lack of expertise (Arrieta *et al*., 2020).

This thesis focuses on applying XAI in the context of financial data analysis. Financial markets present unique challenges due to their complexity and the criticality of decision-making. As a result, the demand for interpretable A.I. models in finance is particularly significant. The following section aims to provide a comprehensive review and analysis of XAI methodologies and strategies, emphasizing their application in financial data analysis.

The thesis is structured into 6 chapters, starting with the introduction, which provides an overview of the research. Chapter 2 comprises a comprehensive literature review, delving into essential subjects relevant to this thesis, including XAI and time series forecasting. Chapter 3 summarizes the data utilized in this research, encompassing details on data processing, model development, testing methodologies, and an investigation into model explainability. The subsequent chapters, from chapters 5 to 6, are reserved for presenting the results, engaging in detailed discussions, and concluding the study.

CHAPTER 2

Literature review

Explainability and interpretability are huge concerns for decision-makers who rely on artificial intelligence, analytics, and data science (Hoffman *et al*., 2018). Such issues can lead to an abrupt stop in the development and implementation of A.I. and ML systems. These issues also led to a significant amount of work from researchers over the years toward providing a better understanding of black-box models. As the decision-making process began to rely more and more on A.I. and ML systems, the necessity to understand the reasons behind the decisions also increased.

Through the years, many articles have been published on both explainable artificial intelligence and the financial market since they are both hot topics. Searching on Google Scholar for terms like "forecasting" and "financial market" reveals that this trend has received a significant amount of publications, with each decade providing almost ten times the results of its previous decade, as observed in Table 1.1.

Time frame	<2004	2004-2009	2009-2013	2013-2018	2018-2022
Results	17,000	5,950	6,430	9,480	11,300

Table 1.1 - Results of querying Google Scholar with "forecasting" and "financial market"

A related analysis shows a similar trend when querying for both "explainable ai" and "financial markets" on one search; the search engine returned 3,620 results for the current year alone. Querying and filtering by year, we can observe an enormous increase in the number of publications, especially in the past five years, as seen in Table 1.2. A search on google trends, shows, as illustrated in Figure 1, a slight trend in searching for keywords like explainable A.I. starting ruffly in the mid of 2015, which correlates with the number of publications that have been put out in the same timeframe. This trend can also be observed in Table 1.2, where we have an exponential increase in published articles over the past ten years. The amount of results for the past five years surpasses the accumulated results for the years before the time span, revealing that this trend will possibly continue for the years to come. The search query for "explainable ai", "financial market" and "forecasting" (Table 1.3) confirms the increase in the usage of explainable artificial intelligence for explaining forecasts of the financial market, with a total of 17,100 hits between 2018 and 2022. The query result indicates that using explainable artificial intelligence for financial market forecasting is a

research topic yet far from being exhausted as the need for understanding models' predictions grows simultaneously.

Time frame	<2004	2004-2009	2009-2013	2013-2018	2018-2022
Results	2,770	1,050	1,130	2,450	12,700

Table 1.2 - Results of querying Google Scholar with "explainable ai" and "financial market"

Time frame	<2004	2004-2009	2009-2013	2013-2018	2018-2022
Results	21,900	19,000	17,600	16,800	17,100

Table 1.3 - Results of querying Google Scholar with "explainable ai", "forecasting" and "financial market"

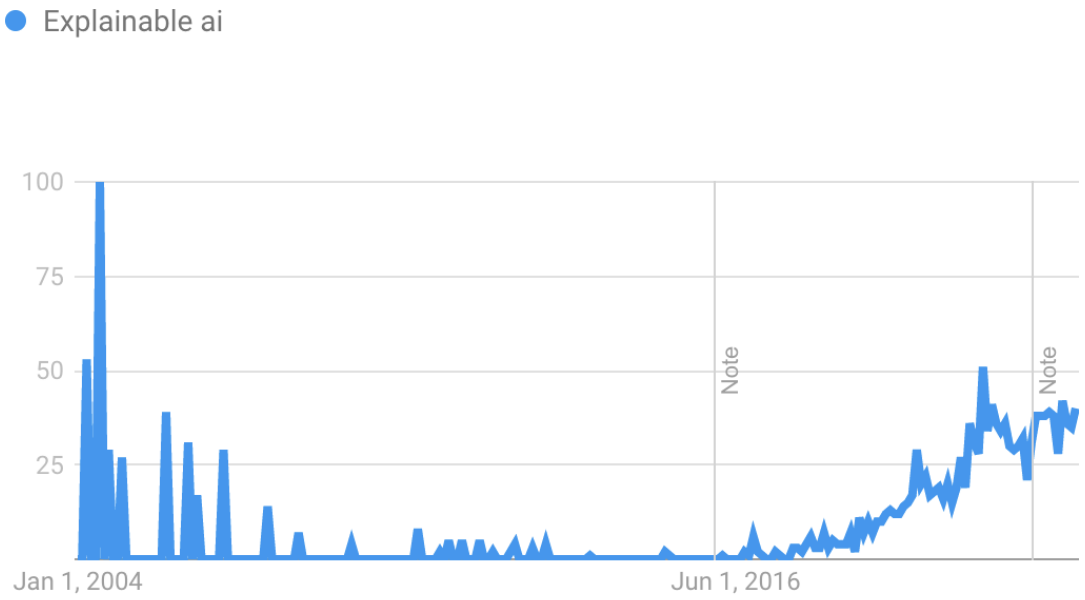


Figure 1.1 - Results of searching Explainable A.I. on google trends

Although it might seem like a new research field – since it has been gaining more popularity recently – the first notions of explainability in Artificial Intelligence had subsided in expert systems in the mid of 1980s (Buchanan & Shortliffe, 1984; Wick & Thompson, 1992 as cited in Confalonieri *et al.*, 2021). When it comes to explainable artificial intelligence, there is plenty of work done, but not as much in the financial area. Regarding the

financial area, some articles were revealed to be relevant to this study and worthy of mention, considering they also used the SHAP values, an approach based on game theory to describe the performance of a machine-learning model (Mangalathu *et al*., 2020) and also one of the most widely used techniques for model explanations, and LIME methodology alongside ML models such as XGBoost, Random Forest, Neural Networks (N.N.) and Logistic Regression (L.R.).

By using gradient boosting decision trees (GBDT), Ohana *et al*. (2021) conducted an experiment in which they tried to predict the prices of the S&P 500 using 150 features and then used the SHAP values to explain why and how the model got the final decision.

Misheva et al. (2021) trained a machine learning-based credit score prediction model on the P2P Lending Platform (Lending Club) dataset. The paper attempted to secure explanatory power by applying post hoc XAI techniques (LIME and SHAP explanations). It used LIME to explain instances locally and SHAP to obtain local and global explanations.

In their work, Agarwal *et al*. (2022), used a financial time series model based on XGBoost and Random Forest algorithms to predict the close values of each stock by using the past 20 days of close values as input, for data range from January 1, 2012, to January 1, 2020. The authors then explained the models' predictions using LIME and SHAP. In conclusion, the models used by the authors gave a high accuracy (98%), but due to their nature - labeled as black box models - they lack explainability and interpretability, which makes them an unreliable tool for a domain where explainability is valued. The explainability issue was solved using LIME and SHAP. As stated by the authors, when used alongside empirically sound machine learning models, you get a forecasting model supported by its empirical results and intuitive explanations.

Balagopalan et al. (2022) surveyed the quality of explanations for different protected subgroups using data from finance, healthcare, college admissions, and the U.S. justice system settings using four explainability methods, both local and global methods, with two different black-box model architectures, Neural Networks and Logistic Regression. The four explainability models (two local and two global) were trained, and they found that fidelity gaps exist between subgroups in many settings. This was done by dividing the data sets randomly into four subsets. The division was made as follows: a training set for black-box models (50%), a training set for explanation models (30%), a validation set for explanation models (10%), and a held-out test set for evaluating both black-box and explanation models (10%). The models were trained on each dataset to serve as black-box models. As a result of this fidelity gap experiment over local explainability, the authors found that LIME favors

disproportionately different subgroups with maximum accuracy (with a threshold of 0.5) gaps varying from 0.1-21.4%, confirming that explanation can differ dramatically by subgroups. Also, with the Area under the ROC Curve **(**AUC)/Accuracy metric varying significantly (0-6%) between subgroups and, as stated by the author, when explanations are judged to be "high quality" based on average fidelity, it might be misleading and lead to errors in decision-making. Secondly, due to the identical features on which the models were trained, the SHAP's gaps were consistently zero. But, using a subset of features to train the explanation model may lead to more valuable explanations, increasing the gaps significantly, indicating that SHAP can also suffer from significant gaps in fidelity when used in practice. The authors used two surrogate models for global explanations: Generalized Additive Model (GAM) and a sparse decision tree (Tree). Then, they evaluated the fidelity of each global method with the original black-box. This experiment showed that the gaps in accuracy between subgroups differed significantly from the global explanatory model means due to an imbalanced proportion between subgroups. Thirdly, they found that using fewer features may lead to more significant gaps in performance between subgroups in sparse decision trees (Trees).

Furthermore, in attempts to improve the explainability, it was found that strong local and global models were more effective for a local explanation by reducing the gaps in three out of the four datasets. Applying Just Train Twice, a more robust version of LIME, achieved this reduction. As for global explanation, rebalancing the subgroups by randomly oversampling minority groups, fixing what failed on global explanation, but this common rebalancing method proved not to be the optimal solution for every case. In conclusion, the authors found that significant performance gaps exist between groups, indicating that some groups may receive better explanations than others, and using robust methods can improve the fairness of explanation models.

One work proposed by Maree *et al*. (2020), on explaining a deep neural network for financial transaction classification featured the application of feature extraction using Shapley values and a hybrid approach of text clustering and decision tree classifier. In this work, the authors combined and compared explanations from surrogate models (i.e., decision tree, which was used as a baseline for explanation) and SHAP. In conclusion, surrogate models used under their conditions showed a tradeoff between explainability and interpretability due to many nodes. A clear understanding of which features bias the model the SHAP technique provided the most. The model was tested in terms of robustness via exposure to targeted evasion attacks.

Also, Çelik *et al*. (2023) introduced a successful framework for direct daily direction prediction in capital markets, emphasizing portfolio diversification opportunities. Machlev *et al*. (2022) highlighted the potential of XAI in power systems, addressing challenges and suggesting future research directions. Freeborough & Van Zyl (2022) explored XAI's transferability to financial time series prediction, revealing insights into model architectures and recommending improvements. Park and Yang (2022) proposed an interpretable LSTM model for economic growth and crisis prediction, considering the impact of COVID-19. Rojat *et al*. (2021) provided an overview of XAI methods applied to time series, emphasizing the need for domain-specific explainable approaches. Dang *et al*. (2020) presented DeepVix, a unified system combining machine learning and visual analytics, enabling visual explainability for multivariate time-series predictions. These studies collectively demonstrate the growing importance of XAI in diverse applications and highlight opportunities for future research and development.

All works described here only used XAI or surrogate models to explain a model's prediction. In this study, we analyzed the most recent period until August 2023 to identify if the novel developments in explaining predictions over financial markets have produced changes.

CHAPTER 3

Methodology and Data Processing

We follow the conventional ML methodology (CRISP-DM), adding the XAI part along the way. The proposed methodology can be observed in the flowchart illustrated in Figure 3.1.

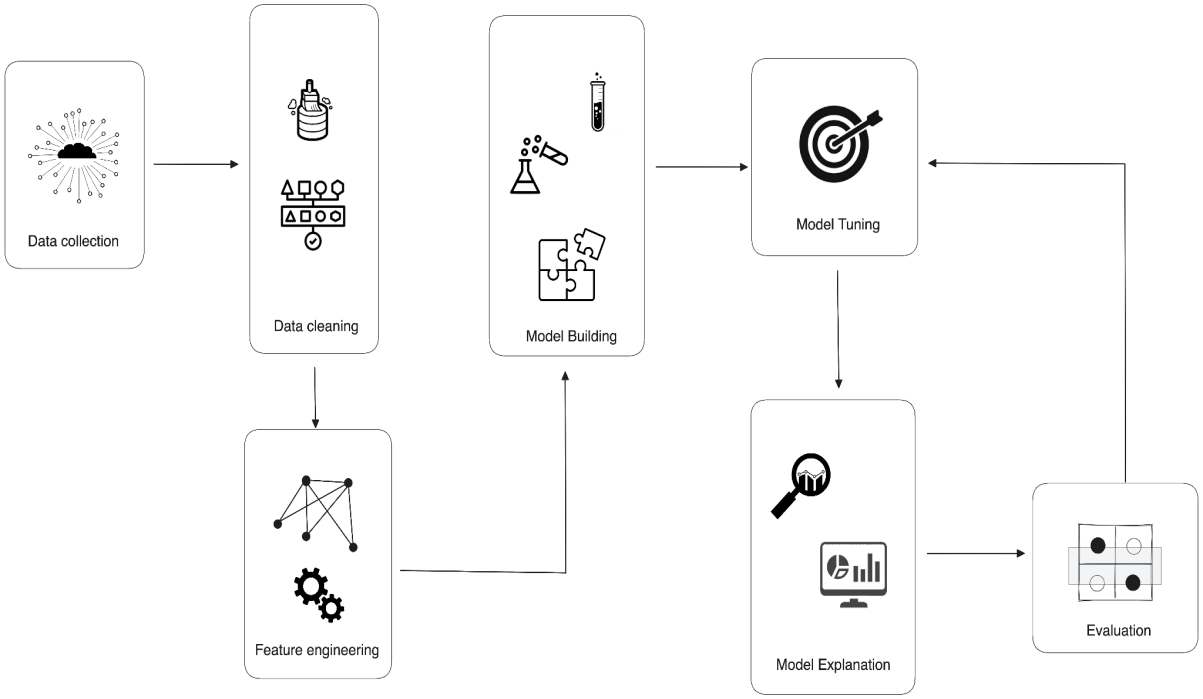


Figure 3.1 - Pipeline employed before and throughout the modeling and explanation phase

The data collection, cleaning, analysis, and feature engineering phases for preparing the data for predicting the price of the S&P 500 involved several steps (the modeling phase can be found in Chapter 4, along with the subsequent phases). The primary dataset used was the historical price of the S&P 500's index, which was acquired from the website *investing.com*. However, several other relevant variables were incorporated as additional/exogenous features to enhance the model's predictive power. These additional features included the Dow Jones index, Bitcoin, Ethereum, Gold, Oil, Silver, Copper, United States Dollar (USD), and Gas prices. Each dataset contained daily values from January 2017 to December 2022, resulting in a sample with 1563 observations. The data collection process involved direct access to the website, and navigating to the desired indices before downloading.

Once the data was obtained, the next step was consolidating all the individual datasets into one comprehensive dataset. For each index collected, only the price information was preserved and renamed to the corresponding index's name. By merging all the datasets, a unified dataset containing all the necessary information for modeling was created. Before proceeding with the modeling phase, the collected data underwent preprocessing to ensure its

quality and suitability. This included cleaning the data to remove any inconsistencies or errors that might affect the accuracy of the model, and this mainly consisted of dealing with null values and preserving only the days-of-the-week data points, as will be discussed further.

The dataset was partitioned into three sets, maintaining an 80-10-10(%) split for the training, validation and testing subsets. This partitioning strategy aimed to ensure adequate data for training the model while providing sufficient data for unbiased evaluation during the testing and validation stages. The training set, comprising 80% of the data, was utilized to train the model on historical information, allowing it to learn underlying patterns and relationships. The validation set, constituting 10% of the data, served to fine-tune the model and gauge its performance further, ensuring it could perform well on completely unseen data beyond the testing set. Lastly, the testing set, representing 10% of the data, was employed to assess the model's performance on unseen data, thereby providing an estimate of its generalization capabilities. This data split enabled a comprehensive evaluation of the model's effectiveness and facilitated the selection of a well-generalized and reliable predictive model.

In preparing the data for analysis, normalization techniques were crucial to ensure that the variables representing various aspects of the financial and economic landscape were brought to a consistent scale. Applying z-score normalization allowed us to standardize the data, making it comparable across different features. Moreover, certain adjustments were made to guarantee compatibility with the model training and evaluation process. Type casting was performed to ensure the appropriate data types were assigned to each feature, facilitating accurate analysis. Additionally, we accounted for potential discrepancies caused by weekends, holidays, or non-trading days through a business day adjustment. This comprehensive preparation allows us to clearly understand the significance of each variable and its distinct roles in the subsequent analyses.

The variables chosen for analysis represent various aspects of the financial and economic landscape. Each variable plays a distinct role and carries specific significance in the study context. The descriptions aim to clearly understand the variables and their role in the subsequent analyses.

- Spx: This variable represents the value of the S&P 500 index. The S&P 500 is a stock market index that measures the performance of 500 large companies listed on stock exchanges in the United States. It is widely regarded as a benchmark for the overall performance of the U.S. stock market.
- Gold: This variable refers to the price or value of Gold. Gold is a precious metal and a popular investment asset. It is known for its intrinsic value, durability, and use as a hedge

against inflation or economic uncertainties. Various factors, such as supply and demand dynamics, geopolitical events, and market sentiment, influence the price of Gold.

- Btc: This variable represents the price or value of Bitcoin. Bitcoin is a decentralized digital currency that operates on a peer-to-peer network. It is the first and most well-known cryptocurrency, and its value is determined by market forces of supply and demand. Bitcoin has gained significant attention as a speculative investment and a potential alternative to traditional fiat currencies.
- Dji: This variable represents the performance or value of the Dow Jones Industrial Average Index (Dji). The Dji is a stock market index that tracks the performance of 30 large, publicly-owned companies in the United States. It is one of the oldest and most widely recognized stock market indexes, often used to indicate the overall health of the United States(U.S.) stock market.
- Eth: This variable refers to the price or value of Ethereum. Ethereum is a decentralized, open-source blockchain platform that enables the development of decentralized applications and smart contracts. Ether (Eth) is the native cryptocurrency of the Ethereum platform and is used to power transactions and computational operations within the network.
- Copper: This variable represents the price or value of copper. Copper is a widely used industrial metal with various applications, particularly in construction, electronics, and manufacturing. The price of copper is influenced by factors such as global demand, supply dynamics, economic indicators, and geopolitical events.
- Oil: This variable refers to the price or value of crude oil. Crude oil is a fossil fuel and a primary energy source refined into various petroleum products. Oil prices are influenced by global supply and demand dynamics, geopolitical tensions, production levels, and market speculation.
- Gas: This variable represents the price or value of natural gas. Natural gas is a fossil fuel for heating, electricity generation, and industrial processes. Supply and demand dynamics, weather conditions, storage levels, and global energy markets influence natural gas prices.
- Silver: This variable refers to the price or value of silver. Silver is a precious metal often used in jewelry, silverware, and industrial applications. It is also considered an investment asset influenced by factors such as supply and demand dynamics, economic indicators, geopolitical events, and market sentiment.
- USD: This variable represents the exchange rate or value of the United States Dollar (USD). The USD is the official currency of the United States and is widely used as a global

reserve currency. The exchange rate of the USD against other currencies is influenced by factors such as interest rates, economic indicators, geopolitical events, and monetary policies.

Additionally, a more detailed exploration of the dataset's characteristics unveils valuable insights through summary statistics derived from 1563 observations. The mean values range widely, from 3.28 to 3282.71, highlighting the diverse central tendencies within the dataset. Likewise, the standard deviation values, varying from 260.51 to 716.96, indicate fluctuations in data spread. The dataset's minimum and maximum values showcase substantial data ranges, for instance, ranging from 1163.20 to 2058.40. Furthermore, the percentiles (25%, 50%, 75%) offer a clear picture of data distribution. The full spectrum of the dataset can be found in Appendix A.

A distinctive pattern emerges when examining Figure 3.2, which represents the S&P 500's closing prices from 2017 to 2022. The graph depicts a series of fluctuations, reflecting the market's inherent volatility during this period. While intermittent downward movements exist, the overall trend displays an upward trajectory. The closing prices exhibit a mix of rises and falls, influenced by various economic and market factors that shaped investor sentiment and market conditions during those years. The decline in 2020 coincided with the onset of the global COVID-19 pandemic, followed by an upwards trend that indicates a recovery from the companies.

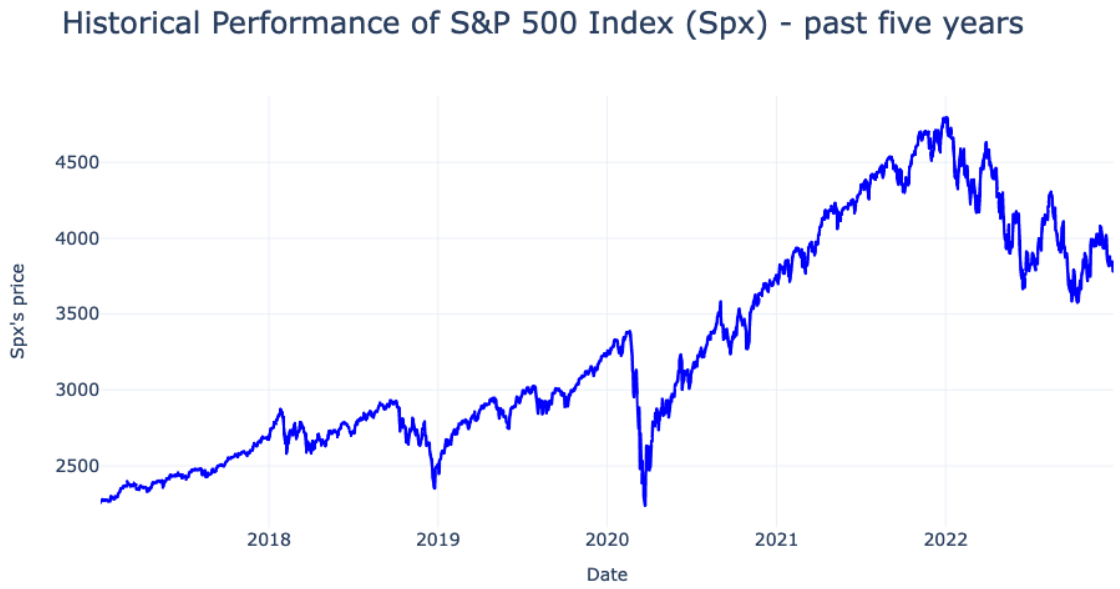


Figure 3.2 - S&P 500 Index (Spx) over the past five years

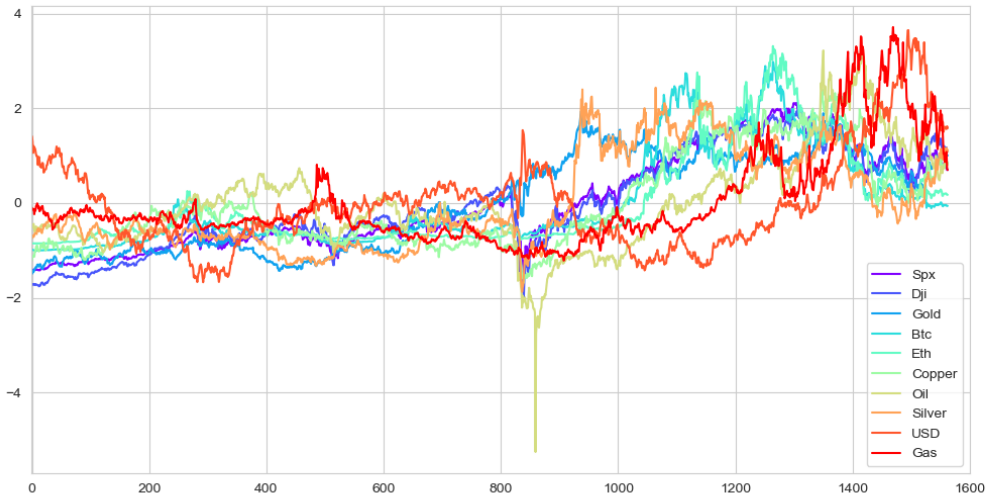


Figure 3.3 - Line Plot Showing the Time Series Evolution of 10 Variables

Figure 3.3 shows the time series line plot of the feature variables. Stochastic trends and volatility are patterns that can be observed immediately.

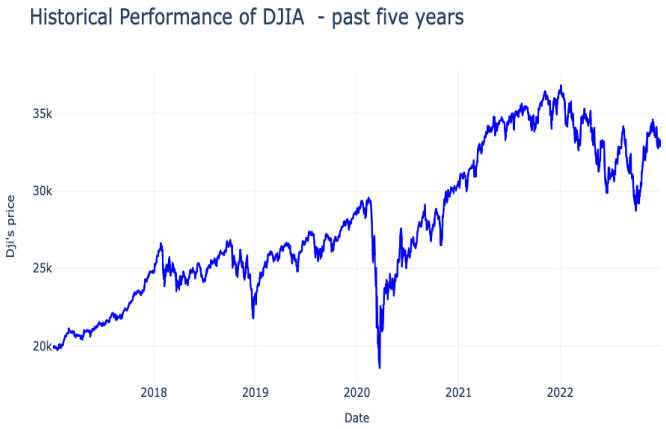


Figure 3.4 - Dow Jones Industrial Average's index over the past 5 years

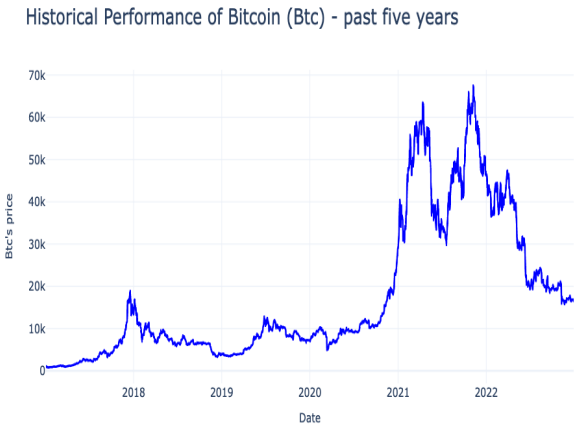


Figure 3.5 - Bitcoin's index over the past 5 years

From 2017 onward, an examination of the Dji, Figure 3.4, reveals a discernible positive long-term trend, marked by periodic corrective movements and the attainment of significant milestones, and also follows the same trend as of the Spx depicted in Figure 3.2. This trajectory reflects the overall growth of the stock market; however, it is also characterized by volatility, frequently influenced by economic events and news. Conversely, Bitcoin prices during the same period have demonstrated remarkable exponential growth, punctuated by recurrent cycles of bullish phases marked by rapid price appreciation and bearish phases

featuring corrections or declines. These two indexes significantly impact the Spx's, as we will see further in Chapter 5, and are also strongly correlated to the Spx, as seen in Figure 3.6.

As we embark on forecasting the S&P 500's price in our work, the correlation matrix plays a pivotal role in capturing the intricate relationships between the target and exogenous variables. Notably, we observe a strong positive correlation between the S&P 500 index and the Dow Jones Industrial Average, indicating a close relationship between these two market indicators. Additionally, there is a somewhat strong positive correlation between the S&P 500 and Bitcoin, suggesting a potential influence of the cryptocurrency market on the stock market dynamics.

Thorough analysis reveals weak correlations between the S&P 500 index and Gas and USD, as seen in Figure 3.6. While correlation provides valuable insights, it is important to note that it does not imply causality. In order to develop an accurate and effective predictive model, we have deliberately decided to exclude these two variables from the dataset. This approach lets us focus on the most influential and statistically significant factors, ensuring the model's accuracy and effectiveness.

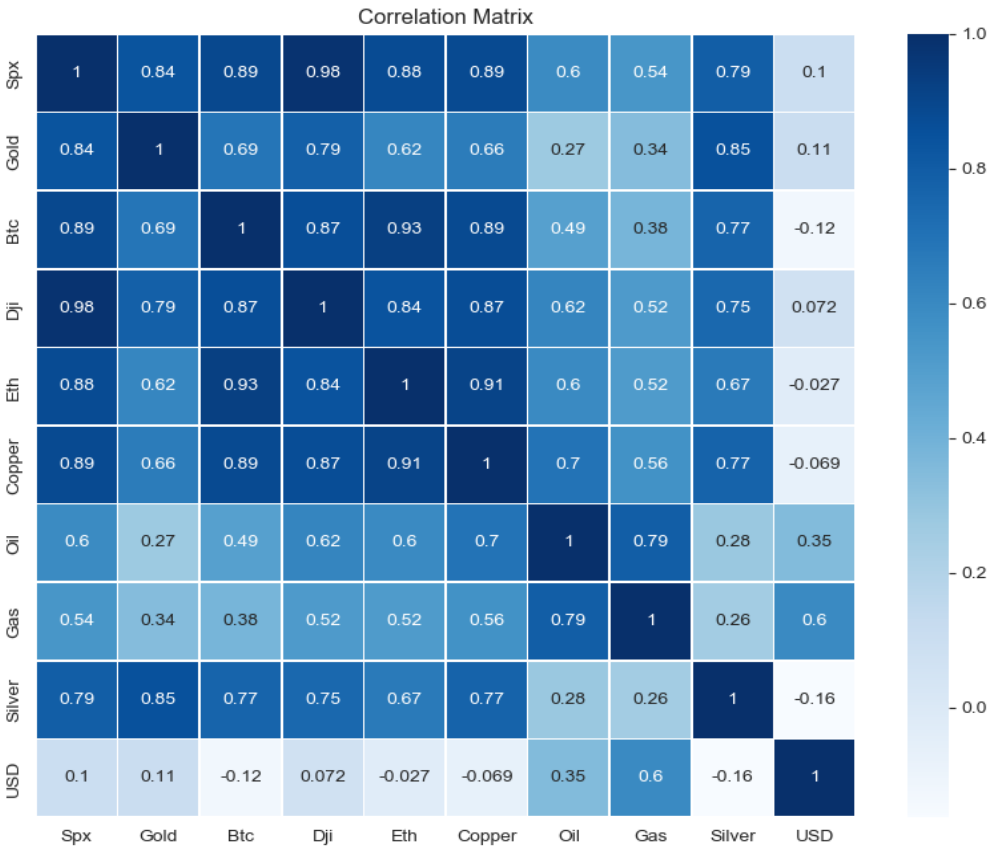


Figure 3.6 - Multivariate Analysis: Correlation Matrix of Key Features

During the data collection and preprocessing phase, we obtained the necessary datasets, merged them, handled missing values, and applied normalization techniques. A correlation analysis revealed strong positive correlations between Spx and Dji and medium-strong positive correlation between Spx and Bitcoin. Knowing these two correlations (Spx-Dji and Spx-Btc) will come in hand when analyzing the results in Chapter 5. The processed data is now ready for model training and evaluation, offering insights into potential market dynamics and their impact on the S&P 500.

CHAPTER 4

Modeling

To advance our investigation, we implemented two Machine Learning algorithms and compared their performances based on various metrics. The following subsections provide detailed insights into each model's architecture, methodologies, and parameter optimizations. The first model, LSTM, is based on recurrent neural networks (RNN), while the second one, XGBoost, follows a different approach with gradient boosting. For both models, we use seven exogenous variables (Gold, Dji, Btc, Eth, Oil, Silver and Copper) to achieve our first task – to predict the Spx's index. We then explain which of these seven variables have more impact on the target variable. Through a rigorous evaluation of their performance using selected metrics (MAE, MSE, RMSE, MAPE), we draw valuable conclusions on their suitability for forecasting the S&P 500 index.

4.1. LSTM

In contrast to its ancestor, RNN, which faced challenges in effectively modeling sequential data, such as the inability to capture long-term dependencies and the difficulty in parallelizing the sequential calculation approach (Wang *et al.*, 2019), LSTM emerges as a specialized type of RNN that overcomes these limitations. LSTM introduces additional capabilities for capturing and memorizing data sequences. The key feature of LSTM lies in its memory cells or system modules, where information flows are recorded and stored (Siami-Namini *et al*., 2018). This unique architecture allows LSTM to capture long-term dependencies, effectively modeling complex temporal patterns in sequential data (Goodfellow et al., 2016).

4.1.1 Base Model development

4.1.1.1 LSTM Model architecture

In the base model development phase, the architecture of the LSTM model was designed to establish a starting point for further improvement. The base model aimed to capture temporal dependencies and extract meaningful information from the sequential data.

The LSTM model architecture consisted of two layers. The first layer had 32 neurons, while the second had 16 neurons. The choice of the number of neurons in each layer was based on experimentation and striking a balance between model complexity and computational efficiency.

In order to introduce non-linearity and capture complex patterns in the data, the Tanh (Hyperbolic Tangent) and ReLU (Rectified Linear Unit) activation functions were utilized in the first and second LSTM layers, respectively. ReLU is a simple and efficient activation

function that outputs the input value if it is positive and zero otherwise, while the Tanh function is a mathematical function that maps the input values to a range between -1 and 1. The parameters utilized on each layer of the base model can be found in Table 4.1.

The base model was also optimized using the Adam optimizer, a popular and efficient optimization algorithm. Adam combines the benefits of AdaGrad[2] and RMSProp[3], adapting the learning rate for each parameter individually. The optimizer's choice aids in efficiently updating the weights and biases of the neural network during the training process.

	Units	Activation Function	Dropout Rate
1st Layer	64	Tanh	0.2
2nd Layer	32	ReLU	0.2

Table 4.1 - LSTM base model's layers architecture

4.1.1.2 Input Shape and Temporal Dependencies

The input shape was crucial in capturing the temporal dependencies inherent in the sequential data. This study considers a three-dimensional input tensor with the following format: (samples, time steps, features).

Namely, the input shape was (1250, 5, 7), meaning there were 1250 samples, each consisting of 5-time steps, and each time step contained 7 features. This structure allowed the LSTM model to process the sequential data meaningfully and leverage the inherent order and temporal information present in the time series.

By structuring the input data this way, the LSTM model could capture the relationships between past observations and make informed predictions based on the patterns learned from the historical data.

4.1.1.3 Base Model evaluation

The base model's performance was evaluated using various metrics to assess its effectiveness in predicting the price of the S&P 500's index. Standard evaluation metrics were considered, including MSE (Equation 1), MAE (Equation 2), RMSE (Equation 3) and MAPE (Equation 4). In what follows are the formulas for each of the metrics:

[2] https://optimization.cbe.cornell.edu/index.php?title=AdaGrad
[3] https://keras.io/api/optimizers/rmsprop/

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \quad (1)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right| \quad (2)$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \quad (3)$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|y_i - \hat{y}_i\right|}{y_i} \quad (4)$$

Where, $\hat{y}_i$ is the predicted value, y_i is the actual value, and n is the number of observations in the dataset.

The base model's predictions were compared against the ground truth values to calculate the MSE, MAE, RMSE and MAPE. These metrics provided insights into the performance of the model's predictions. Additionally, several plots were used to analyze the model's performance visually.

While the base model served as a starting point, it is important to acknowledge its limitations and identify space for improvement. The evaluation results helped identify shortcomings and better understand the model's strengths and weaknesses. Valuable insights were obtained through the evaluation of the base model, laying the groundwork for the subsequent steps of optimizing the model and improving its performance through hyperparameter tuning, as discussed in the next section. The calculated metrics for the base model's performance can be found in Table 4.2.

4.1.1.4 Hyperparameter tuning

Hyperparameter tuning plays a major role in achieving the optimal performance of a model, as the default hyperparameters may not always result in the highest accuracy or best predictive capabilities (Schratz *et al*., 2019). Fine-tuning the hyperparameters becomes crucial to ensure the model is effectively adapted to the specific problem. However, hyperparameter tuning can be complex and time-consuming, mainly when dealing with models that consider multiple parameters (Liu *et al.*, 2019). It involves iteratively exploring different sets of hyperparameters to find the optimal configuration that maximizes the model's performance. Various techniques, such as grid search, random search, and more advanced approaches like Bayesian optimization, can be employed to explore the hyperparameter space and identify the

best parameter combination. These techniques help overcome the challenges of hyperparameter tuning and enable the model to be fine-tuned for optimal results.

In what follows, we briefly explore a commonly employed hyperparameter tuning technique, the Random Search algorithm.

4.1.1.5 Random Search optimization

The random search optimization algorithm randomly samples from the hyperparameter space to find the optimal value for the parameters of a model (Nevendra *et al.,* 2022). This technique offers a more efficient alternative than algorithms such as grid search (which evaluates all possible combinations for the parameters, making it more time and resource costly). Instead, it explores a subset of combinations iteratively, which can often yield comparable results to grid search with significantly fewer evaluations. For each randomly sampled hyperparameter configuration, the model's performance was evaluated using the same metrics used for the base model, and all the scores can be found in Table 4.2. Also, loss graphs were plotted to visualize each randomly sampled configuration's convergence and training progress, as shown in Figure 4.1. Figures 4.2 to 4.5 depict the result of four experiments conducted over varying numbers of past observations to find the better fitting window for future forecasting; refer to Section 4.1.1.7 for more details on this. The training-validation loss graphs suggest a slight overfit of the model, addressed in the final version of the model by introducing two regulators, L1 and L2[4] , which is a common approach to address overfitting in machine learning models, particularly in the context of linear regression and neural networks.

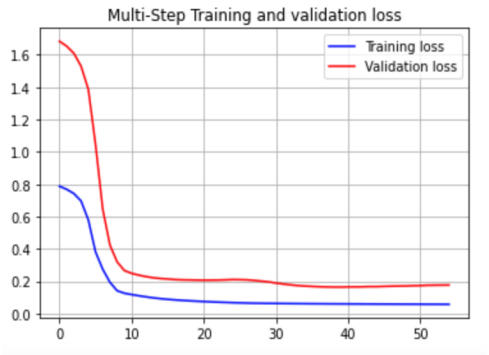


Figure 4.1: Random search loss graph for 5 past observations

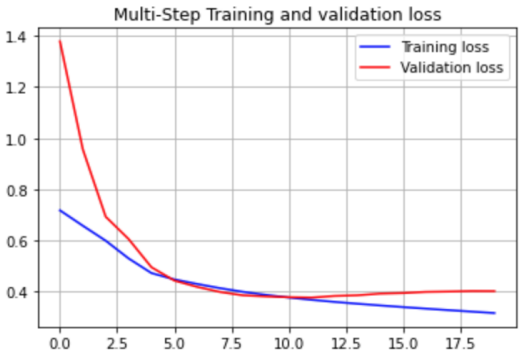


Figure 4.2: Random search loss graph for 10 past observations

[4] The L1 regularization approach is the Lasso Regression, while the L2 model is the Ridge Regression. The main distinction between these two is the penalty term. Ridge regression penalizes the loss function with the "squared magnitude" of the coefficient. For more information on the L1 and L2 please refer to https://towardsdatascience.com/l1-and-l2-regularization-methods-ce25e7fc831c

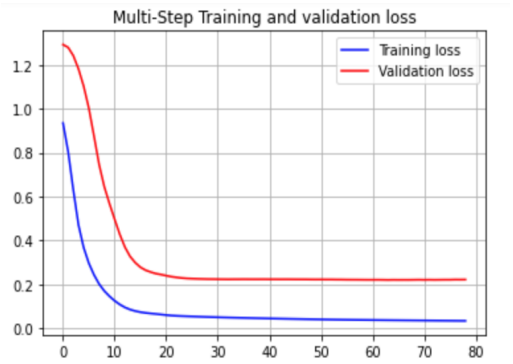


Figure 4.3: Random search loss graph for 22 past observations

Figure 4.4: Random search loss graph for 60 past observations

Upon initial examination, an intriguing pattern emerges when comparing the four graphs. The loss function depicted in Figure 4.2 stands out as it suggests a more pronounced relationship between the training and validation loss than the other graphs. However, it is important to highlight that this graph uniquely showcases an overfitting scenario where the model excessively tailors its performance to the training data. Conversely, the observation for 22 lags, as depicted in Figure 4.3, falls significantly shorter in terms of accurately approximating the relationship between the training and validation loss compared to the graph in Figure 4.1, which showcases a better relationship between the train and validation when compared to the other three. It is worth mentioning that Figure 4.1, the experiment in which we introduce 5 lags, offers the most compelling visualization while indicating the minimum loss, showcasing the strongest alignment with the metrics presented in Table 4.2. It is also worth mentioning that the graph in Figure 4.2 could show better results with further parameter tuning.

4.1.1.6 Hyperparameters tuned

During the hyperparameter tuning phase, several hyperparameters were considered for optimization. These hyperparameters include:

- Learning Rate: The learning rate determines the step size at which the model adapts its internal parameters during training.
- Number of Layers: The number of LSTM layers in the model can impact its capacity to capture complex patterns and represent the data.
- Number of Neurons: The number of neurons in each LSTM layer influences the model's complexity and ability to learn intricate features.
- Dropout Rate: Dropout is a regularization technique that randomly drops a fraction of the neurons during training, reducing overfitting.

- Batch Size: The batch size determines the number of samples processed before updating the model's internal parameters.
- Activation function: This hyperparameter determines the activation function used in the neural network. It is represented as a range from 0 to 7, where different values correspond to other activation functions like sigmoid, tanh, ReLU, etc.
- Optimizer: This hyperparameter specifies the optimization algorithm that updates the neural network weights during training.
- Normalization: This hyperparameter determines whether input data should be normalized before being fed into the neural network.

4.1.1.7 Performance Evaluation Metrics

In order to find the right time window for the observations that the model should consider before making predictions, four experiments were conducted consisting of 5, 10, 20, and 60 lags of historical observations. During these experiments, we also considered the presence and absence of cryptocurrencies in the dataset to see how the model behaves when exposed to a specific number of past observations when accounting for crypto and when not.

Accessing the values employed by the metrics allows us to quantify the model's predictive accuracy and to compare and select the best-performing hyperparameter configuration. The hyperparameters suggested by the random search[5] algorithm using a window with 5 past observations while having crypto currency in the mix are applied to the LSTM model to obtain the best predictions since these performed better. The Bayesian optimization[6] results can be found in Appendix B. In Table 4.2, the Bayesian technique consistently ranks second in every evaluation during this optimization phase, except for two specific cases, with one of them being the case where the model is trained without the cryptocurrencies with 22 lags and the other one with the same dataset being fed to the model but with 10 lags.

It is worth mentioning that even though the model performed better in some cases when fed data without the cryptocurrencies, we proceeded to use the data with the cryptos as the metrics difference was not too significant. We do this intending to observe later how the

[5] *Python was utilized for the entire hyperparameter tuning and modeling process. The* `tensorflow` *library was employed, along with the* `sklearn.model_selection.TimeSeriesSplit` *and* `sklearn.model_selection.RandomizedSearchCV` *modules for time series cross-validation and randomized hyperparameter search, respectively. The* `tensorflow.keras.wrappers.scikit_learn.KerasRegressor` *wrapper was used to make the model compatible with scikit-learn's estimator interface.*

[6] https://towardsdatascience.com/bayesian-optimization-concept-explained-in-layman-terms-1d2bcdeaf12f

cryptos influence our models, if it does at all. The results are presented in Table 4.3, which shows the hyperparameter configuration that yielded the highest predictive accuracy.

Model with crypto features					Model without crypto features				
	MAE	**MSE**	**RMSE**	**MAPE(%)**		**MAE**	**MSE**	**RMSE**	**MAPE(%)**
Performance for past 60 days					**Performance for past 60 days**				
base model metrics	0.26	0. 10	0.32	17.80	base model metrics	0.43	0.24	0.49	27.32
Bayesian model metrics	0.30	0.14	0.38	22.35	Bayesian model metrics	0.65	0.50	0.71	42.49
Random search CV model metrics	0.36	0.20	0.44	23.07	Random search CV model metrics	0.75	0.79	0.89	49.83
Performance for past 22 days					**Performance for past 22 days**				
Base model metrics	0.38	0.20	0.44	21.99	Base model metrics	0.31	0.14	0.37	18.31
Bayesian model metrics	0.26	0.10	0.31	16.80	Bayesian model metrics	0.17	0.04	0.20	10.39
Random search CV model metrics	0.22	0.07	0.26	13.95	Random search CV model metrics	0.25	0.10	0.32	14.87
Performance for past 10 days					**Performance for past 10 days**				
Base model metrics	0.30	0.12	0.35	17.34	Base model metrics	0.30	0.13	0.37	17.71
Bayesian model metrics	0.30	0.12	0.34	17.48	Bayesian model metrics	0.24	0.07	0.27	14.05
Random search CV model metrics	0.40	0.26	0.51	23.71	Random search CV model metrics	0.18	0.05	0.22	11.09
Performance for past 5 days					**Performance for past 5 days**				
Base model metrics	0.25	0.08	0.29	14.70	Base model metrics	0.30	0.13	0.36	17.47
Bayesian model metrics	0.24	0.08	0.29	15.54	Bayesian model metrics	0.26	0.09	0.30	15.65
Random search CV model metrics	0.18	0.04	0.21	10.61	Random search CV model metrics	0.19	0.05	0.23	12.55

Table 4.2: Model's performance (normalized values)

Epochs	Lstm units 1	Lstm units 2	Dropout rate	Learning rate	steps per epoch	Optimizer	Activation
100	64	42	false	0.0001	265	Nadam	Gelu

Table 4.3: Model's best hyperparameters with Random Search optimization

The table of best hyperparameters suggested by the random search algorithm provides valuable insights into the optimal configuration for the model. These hyperparameters have been identified as the most effective choices for maximizing performance based on the search conducted. These optimal hyperparameters offer a starting point for further fine-tuning and optimization to enhance the model's overall performance and, with time, can even be more tuned.

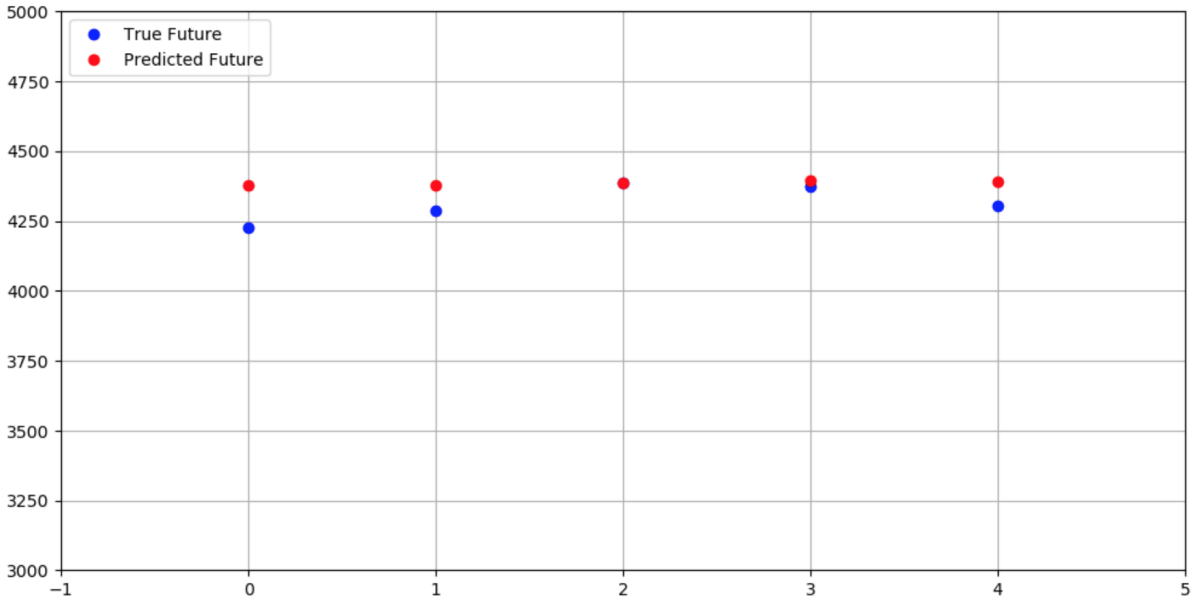


Figure 4.5 - Comparative Analysis of Predicted vs. True Values for S&P 500, Random Search Optimized Hyperparameters with 5 lags

In Figure 4.5, the model with the hyperparameters suggested by the random search, with the data in its original scale, resulted in a strong approximation between the predicted and true values. The optimized model accurately captures the underlying patterns of the regression task and highlights the model's ability to generalize well to unseen data while minimizing overfitting, making it a robust solution for our regression problem.

In evaluating our LSTM model's performance over the original values, we computed several metrics that provided insights into the model's predictive capabilities. Our model's MAE is 224.18, indicating an average absolute deviation of this magnitude between the

predicted and actual values. Additionally, the MSE is calculated at 74662.67, reflecting the average squared deviation between predictions and ground truth values. To provide a more interpretable measure of prediction accuracy, the RMSE is determined to be 273.24. In the mean, the predicted value differs by 273$ from the actual value. Lastly, the MAPE is evaluated to be 4.91%, which quantifies the average percentage difference between predicted and actual values. These metrics collectively serve as essential indicators of our LSTM model's performance and will be valuable for assessing its reliability in forecasting tasks, as discussed in subsequent sections of this thesis.

In conclusion, the application of the random search algorithm significantly improved our model's performance and its ability to approximate the target values accurately. Table 4.2 also contains the results (in which the random search algorithm stands out from the others) of all the experiments performed in searching for the optimal model that best fits our data. The same model will be used in the explainability chapter based on the lag that yielded the best results, as shown in Table 4.2. Notably, the RMSE values for the 5-lag experiment with cryptocurrencies among the exogenous variables for the random search algorithm were the lowest among all four lags tested, accounting for 0.21, indicating the model's strong predictive capability in capturing short-term dependencies. When experimenting with the other three lags(10, 22, 60) the model shows more prominent results when working with 22 lags, with an RMSE of 0.41, and the falling last with the worse RSME is the 10-lag experiment accounting for 0.51. As seen in Figure 4.5, when predicting 5 future values (5 lags), the alignment between the predicted and true values instills confidence in the model's reliability, making it a promising choice for our specific regression task.

4.2 XGBoost

The extreme gradient boost (XGBoost) algorithm is a powerful implementation of gradient boosting that has gained widespread popularity in machine learning (Zhang *et al*., 2021). As an ensemble method, XGBoost combines the predictions of multiple weak classifiers, typically decision trees, to create a more robust and accurate classifier.

One of the key advantages of XGBoost is its ability to address some of the limitations of individual decision tree classifiers. By combining the features of multiple trees, XGBoost can overcome weaknesses such as handling missing values more effectively, mitigating the risk of overfitting, and optimizing calculations in parallel, significantly reducing the model's training and prediction time (Luo *et al*., 2021).

The versatility and effectiveness of XGBoost have been demonstrated across a wide range of domains and applications. Its performance has made it popular for various machine

learning tasks, including classification, regression, and ranking problems (Chen *et al*. 2016). Furthermore, XGBoost has been successfully applied in areas such as image recognition, natural language processing, and financial forecasting.

4.2.1 Model architecture

In order to implement the XGBoost model to predict the price of the S&P 500 using the exogenous variables, it is essential to define the key parameters that govern its behavior and performance. The selection of these parameters significantly impacts the model's performance, complexity, and generalization capabilities. This step outlines the most important XGBoost parameters and justifies our choices based on relevant literature or empirical evidence.

The XGBoostRegressor is designed for regression tasks and employs the sum of squared errors (SSE) as the loss function. By minimizing the squared error, the model aims to reduce the average squared difference between predicted and target values.

As previously stated, the XGBoostRegressor architecture consists of an ensemble of decision trees, where each tree is sequentially added to the ensemble to correct the errors made by the previous trees. The number of trees in the ensemble, known as *n_estimators*, determines the model's complexity and capacity. In our study, the optimal number of trees was determined to be 64, through hyperparameter tuning.

Additionally, the *max_depth* hyperparameter controls the maximum depth of each decision tree in the ensemble. A higher *max_depth* allows the trees to capture more complex relationships in the data, but it can also lead to overfitting if not regularized properly. Through hyperparameter optimization and evaluation, we found that a *max_depth* of 4 balanced capturing important patterns and avoiding overfitting.

In summary, the XGBoostRegressor, with its ensemble of decision trees and the squared error loss function, offers a robust framework for regression tasks, and the parameters for this model were not explored in as much depth as those of the LSTM since a simple set of parameters performed well, after the hyperparameters optimization process.

4.2.2 Hyperparameters tuning

To optimize the performance of the XGBoostRegressor model, we conducted hyperparameter tuning using GridSearchCV. GridSearchCV is the process of choosing models and hyperparameters by testing and evaluating each combination separately (Wu *et al.,* 2022) and systematically exploring a predefined grid of hyperparameter values to identify the optimal combination.

After running the optimizer, we identified the optimal hyperparameter values for our regression task: n_estimators = 64, max_depth = 4, and learning_rate = 0.2. The fine-tuned selection aimed to optimize the performance of XGBoostRegressor, considering both accuracy and generalization capabilities.

4.2.3 Performance evaluation metrics

The XGBoost model's performance was evaluated using various metrics, the same metrics used to assess the LSTM model. The evaluation showed that the model predictions closely align with the actual values of the time series data. The MAE of 144.43 indicates an average difference of approximately 144.43 units between the predicted and actual values. The MSE of 31727.29 represents the average squared difference between the predicted and actual values, with a higher weight on larger errors. The RMSE of 190.96 provides an interpretable measure of the average magnitude of these errors. The MAPE of 4.16% represents the average percentage difference between the predicted and actual values. These evaluation metrics indicate the XGBoost model's effectiveness in capturing the time series' underlying patterns. Furthermore, a visual representation in Figure 4.6 illustrates the model's performance in forecasting the target variable, showing a close alignment between the predicted and actual values. The model demonstrated accurate predictions for the target variable with the close alignment between predicted and actual values and the low values of MAE, MSE, and RMSE, and MAPE. These results highlight the XGBoost model's potential as a reliable and accurate forecasting tool for the research problem.

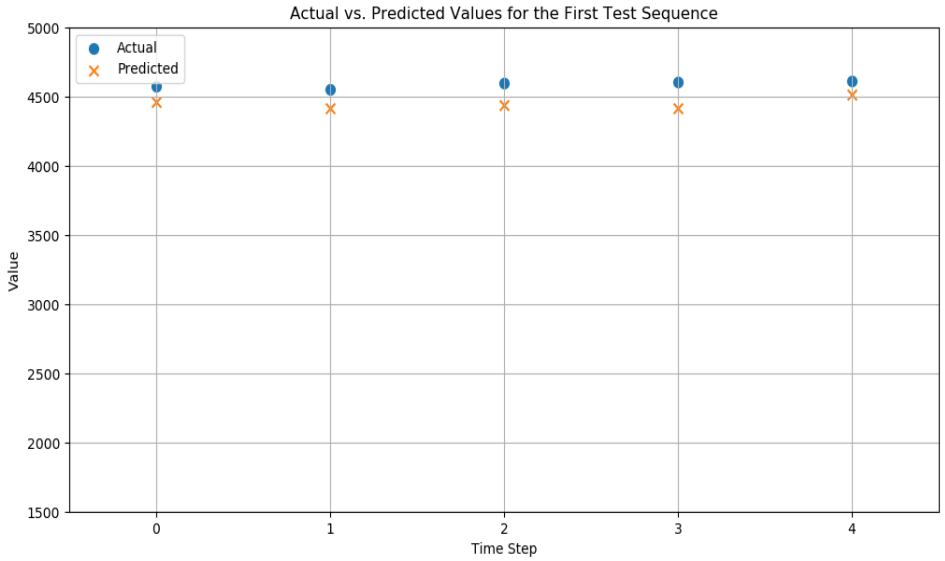


Figure 4.6: XGboost predicted vs actual values

The XGBoost performance graph in Figure 4.6 shows that the model's predictions are fairly accurate, closely following the true values. Although they are not extremely close, this indicates a balanced level of generalization, avoiding overfitting. Overall, Figure 4.6 shows that the XGBoost model has learned meaningful patterns from the data and can make reasonably accurate predictions on new, unseen instances.

4.3 LSTM vs XGBoost

The two models were trained to forecast five future values of the S&P 500's price using seven features: Dji, Gold, Btc, Eth, Copper, Oil, and Silver. The performance of the LSTM and XGBoost models was evaluated based on RMSE and MAPE metrics.

The LSTM model achieved an RMSE of 705.94 and a MAPE of 11.94%. These results indicate that the LSTM model exhibited a higher error level in its predictions than the XGBoost model. The relatively larger RMSE and MAPE values suggest that the LSTM model had difficulty accurately forecasting the S&P 500 using the given features.

On the other hand, the XGBoost model demonstrated more promising performance, with an RMSE of 190.96 and a MAPE of 4.16% as seen in Table 4.4. These lower values show that the XGBoost model yielded more accurate predictions than the LSTM model, providing a better approximation of the actual S&P 500 values using the given feature set.

Model	RMSE	MAPE(%)
LSTM	705.94	11.91
XGBoost	190.96	4.16

Table 4.4: LSTM and XGBoost metrics for 5-day out-of-sample forecast

Based on the shown in Table 4.4, it can be concluded that the XGBoost model outperformed the LSTM model in terms of forecasting the S&P 500 using the selected features. The XGBoost model exhibited lower prediction errors, indicating its potential suitability for this specific forecasting task. However, further analysis and evaluation may be required to explore other factors contributing to the performance differences between the two models, such as the model architectures, hyperparameters, and dataset characteristics.

The investigation revealed that removing the cryptocurrencies from the list of exogenous variables, in both models, had a positive impact on each of the model's performance, particularly concerning the LSTM model (Table 4.2). When the cryptocurrencies were excluded, the model demonstrated better performance in 33.3% of all experiments conducted

in the LSTM model. On the other hand, keeping the crypto variables led to a modest improvement in both model's performance, making their inclusion a feasible option. Overall, the XGBoost model outperformed the LSTM one with a difference of approximately 71.19% and 74.72% on the MAPE and RMSE metrics, respectively. Moreover, results from Table 4.2 suggest that carefully considering cryptocurrency variables can contribute to optimizing the model's predictive capabilities. However, as we will see in Chapter 5, cryptocurrencies may not always have the desired impact on a model's performance.

CHAPTER 5

Explaining the predictions of the LSTM and XGBoost models

5.1 Applying LIME to the LSTM Model

The LSTM model, with its sequential nature, presents unique challenges and opportunities when it comes to applying explainability methods. In this section, we delve into applying LIME[7] to interpret the predictions and behaviors of the LSTM model.

Following the literature, due to its ability to explain individual predictions in a local context, LIME is applied to make local explanations of the LSTM model used for forecasting the S&P 500's price to gain insights into the relationships between the exogenous variables and the model's predictions.

The LIME algorithm, initially designed for tabular data, is adapted for LSTM models in our research by using the module *lime_tabular* from the *lime* library itself and then the *RecurrentTabularExplainer* which was designed explicitly for recurrent models like LSTM, taking into account the temporal nature of the data.

Applying LIME to the LSTM model, we gained valuable insights into the significance of exogenous variables in predicting the S&P 500's price for specific instances. By analyzing 10 instances carefully selected based on the model's confidence, we identified the most influential variables through the LIME analysis (refer to Table 5.1). These findings shed light on the factors driving the LSTM model's predictions and provide a deeper understanding of the impact of various variables on the S&P 500's price in specific market scenarios.

Dji appeared 5 times as a positive variable in all 10 instances, indicating its positive influence on the S&P 500's price predictions. The positive relationship between Dji and the S&P 500 is consistent with historical trends, where changes in the Dji have often been mirrored in the S&P 500 (Liu *et al*., 2016).

On the negative side, BTC (Bitcoin) appeared in the top 3 negative instances with 21 negative appearances in all 10 cases, suggesting its consistent negative influence on the S&P 500's price prediction. Similar negative impacts were also observed for Dji and Eth, which also came up in the top 3 negative instances with 15 and 10 negative appearances, respectively, in all 10 cases.

[7] https://homes.cs.washington.edu/~marcotcr/blog/lime/

Furthermore, the analysis revealed the importance of Gold, which appeared in the top 4, being the most positive variable among the others.

The updated analysis of 10 confident instances using LSTM-LIME highlights the importance of Bitcoin, Ethereum, Dji and Gold prices as the most influential variables in the LSTM model's predictions for the S&P 500's price. The presence of Dji as both a positive and negative variable underscores its context-dependent relationship with the S&P 500.

Variables	Occurrences	Occurrences (Positive)	Occurrences (Negative)	Weighted impact[8]
Btc	24	3	21	-18
Dji	10	5	15	-10
Eth	15	5	10	-5
Gold	8	6	2	4

Table 5.1 - Top Influential Variables in S&P 500 Price Prediction for the LSTM model according to LIME

LIME analysis of 10 of the most confident instances reveals essential insights into the factors driving the LSTM model's predictions for the S&P 500's price. This analysis shed light on the crucial role played by Dji, BTC, Eth and Gold prices in predicting the S&P 500's price within the LSTM model. The results highlight BTC as a significant driver, with a weighted impact of -18, signifying its substantial influence on the target variable. Meanwhile, Dji emerges as the secondary driver, appearing in all ten instances with a weighted impact of -10, indicating its significant impact, especially in negative contexts.

The highly negative impact of BTC on the S&P 500's index is also in line with the findings of Erdas and Caglar(2018), which suggested how changes in cryptocurrency affect the investor's decision regarding the S&P 500.

The frequent appearances of the variable Dji in the LIME-LSTM analysis suggest its significant importance in the LSTM model's performance and interpretability. With Dji appearing in all 10 instances, it becomes evident that Dji holds substantial predictive power and influences the model's decision-making process. Its repeated occurrence in LIME explanations indicates that it plays a crucial role in understanding the model's behavior for specific instances. Consequently, changes in Dji (also in BTC) are likely to have a significant

[8] To calculate the weighted impact, we assign a weight of +1 to the positive influence and -1 to the negative influence and then sum the products. The following formula applies:
w = (num_pos_occurrence * +1) + (num_neg_occurrence * -1)

impact on the model's predictions or explanations, and its relevance may vary depending on the context or data instances being analyzed. Additionally, the presence of Btc and Eth as influential variables reflects the changing landscape of financial markets, where the interactions between traditional and cryptocurrency markets are becoming increasingly relevant in predicting stock prices.

In order to assess the real impact of these variables, we ran the LSTM model using only these top three features (Dji, Gold and Eth). The results were remarkable, but not unexpected. The RMSE dropped significantly from 705.94 to 273.24, substantially improving the model's predictive accuracy. Additionally, the MAPE also saw a considerable reduction, decreasing from 11.94% to 4.94%. These compelling metrics demonstrate the critical role played by the Dji, Gold, and Eth variables in enhancing the LSTM model's performance and overall predictive power.

It is worth mentioning that even though bitcoin had a substantial appearance in the instances where the model felt more confident, its highly negative weighted and negative impact overall also emerged once used among the other three variables and proved to penalize the model's performance as the metrics only went up and through experimentations when left aside, the model yielded better results.

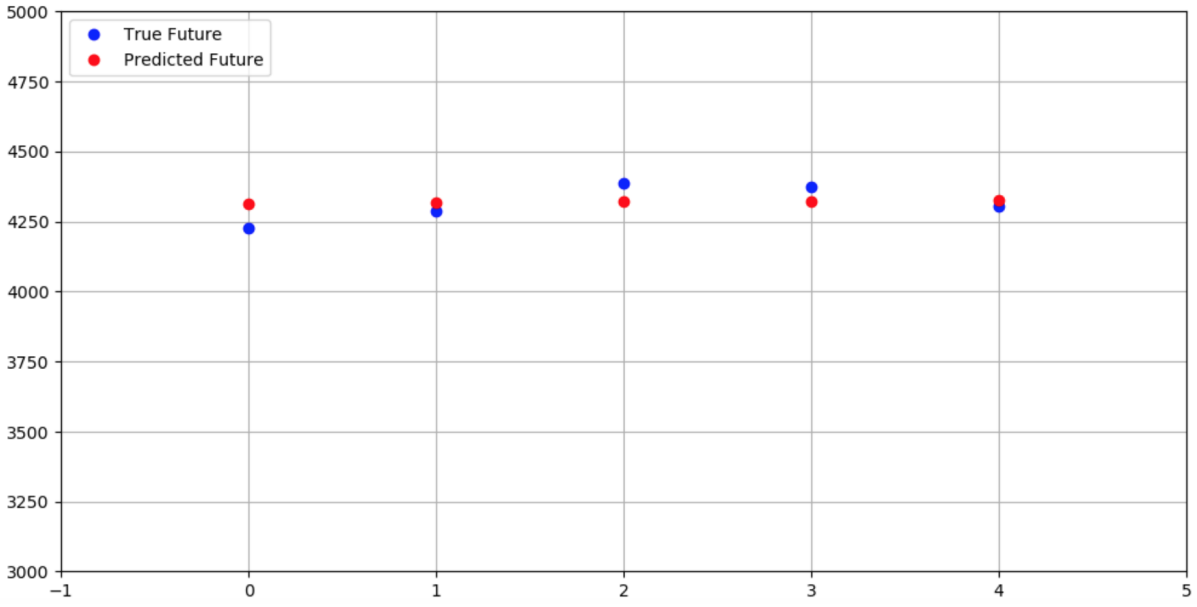


Figure 5.1 - Prediction Graph: Comparative Analysis of Predicted vs. True Values Using Top 3 Variables with 5 lags on the LSTM Model

The LSTM performance graph, depicted in Figure 5.1, demonstrates significantly improved predictions compared to the initial version (illustrated in Figure 4.5). The model shows high precision by closely aligning its predicted values with the true values, indicating

robust generalization and avoiding overfitting. The considerable reduction in RMSE and MAPE further validates the LSTM model's effectiveness for predictive analysis on unseen instances.

However, it is important to acknowledge that these conclusions are based on local interpretations for specific instances and should be further validated and generalized through additional analysis and exploration of more instances.

5.2 Applying SHAP and LIME to the XGBoost Model

The SHAP analysis was successfully conducted on the XGBoost model, providing valuable insights into the importance of the exogenous variables in predicting the S&P 500's price. The SHAP values quantify the contribution of each variable to the model's predictions, allowing us to determine their relative importance.

According to SHAP analysis, the variables' importance in predicting the S&P 500's price ranked as follows: Dji (Dow Jones Industrial Average) had the highest influence, followed by Silver, with its dynamics related to industrial demand, inflation expectations, and global economic conditions, which also had a substantial impact. Gold, known as a safe-haven asset (Kumar and Padakandla, 2022), appears in the top three, providing insights into investor risk appetite. Oil, Copper, and Eth were also significant predictors, representing oil price fluctuations, global economic conditions, and the relevance of the cryptocurrency market. Btc (Bitcoin) had the least importance, suggesting a lower impact on predictions. These findings enhance our understanding of factors driving the S&P 500's forecast of the XGBoost model. Figure 5.2 illustrates the SHAP results.

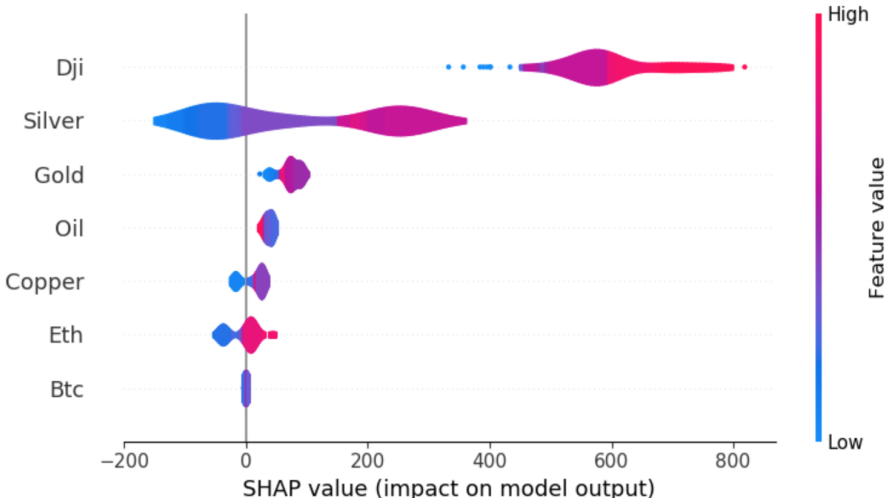


Figure 5.2: Variables importance according to SHAP

Applying LIME to explain local observations of the XGBoost model also provided valuable insights. The LIME analysis was conducted on the XGBoost model using multiple

instances, aiming to identify the relative importance of the exogenous variables in predicting the S&P 500's price for each specific instance. The analysis revealed the top influential variables for predicting the S&P 500's price across these instances, shedding light on the factors driving the model's predictions, the importance of certain variables, and their impact on the S&P 500's movements within the XGBoost model.

According to the LIME analysis, the top influential variables for predicting the S&P 500's price in this instances are as follows (refer to Table 5.2 for more details): the most influential variable was found to be Dji, appearing in all 10 instances analyzed with positive influence, indicating that changes in the Dow Jones index have a significant positive impact on the predicted S&P 500's price for the 10 cases investigated. Oil and Ethereum also appeared in all 10 instances, having a positive effect, signifying their consistent positive influence on the model's prediction.

Dji, oil, and Eth suggest that these variables are pivotal in driving the S&P 500's price movements within the XGBoost model. The Dow Jones index reflects overall market sentiment and investor confidence, while oil and Ethereum are likely influenced by broader economic conditions and cryptocurrency market growth, respectively.

Furthermore, silver appeared in all 10 instances as a negative variable, indicating its consistent negative impact on the S&P 500's price prediction for this specific instance. Similarly, Bitcoin appeared in 8 out of 10 cases with a negative influence and 2 out of 10 as positive, suggesting that changes in the Bitcoin market had a counteractive effect on the model's predictions for those 8 particular instances.

Variables	Occurrences	Occurrences (Positive)	Occurrences (Negative)	Weighted Impact
Dji	10	10	-	10
Oil	10	10	-	10
Eth	10	10	-	10
Btc	10	2	8	-6
Silver	10	-	10	-10

Table 5.2 – LIME Top Influential Variables in S&P 500's Price Prediction for the XGBoost model

After conducting the XGBoost-LIME analysis and identifying the top 3 positive influential variables, we proceeded to refine the model by running it with these variables. The

selected positive variables, namely Dji, Oil, and Eth, were incorporated into the XGBoost model. As a result of this enhancement, the model's performance notably improved, as evidenced by enhanced evaluation metrics, with the RMSE decreasing significantly from 190.96 to 155.47, indicating a remarkable reduction in prediction errors. Moreover, the mean absolute percentage error also experienced a notable improvement, declining from 4.16% to 2.98%. The predictions aligned more accurately with the actual S&P 500's price movements, demonstrating the significance of these variables in capturing underlying patterns and contributing to better predictive capabilities.

Oil prices are one of the variables that appeared in the top 3 positive instances in 10 out of 10 cases, indicating its significant impact on the model's prediction. This finding aligns with existing knowledge in the financial market, such as the work of Kilian and Park (2009), which suggests that the global crude oil market's demand and supply shocks are collectively responsible for 22% of the long-term volatility in U.S. real stock returns, as the price of oil has been recognized as a key driver of the S&P 500's movements. The influence of oil suggests that fluctuations in the oil market can have a substantial effect on the S&P 500's price.

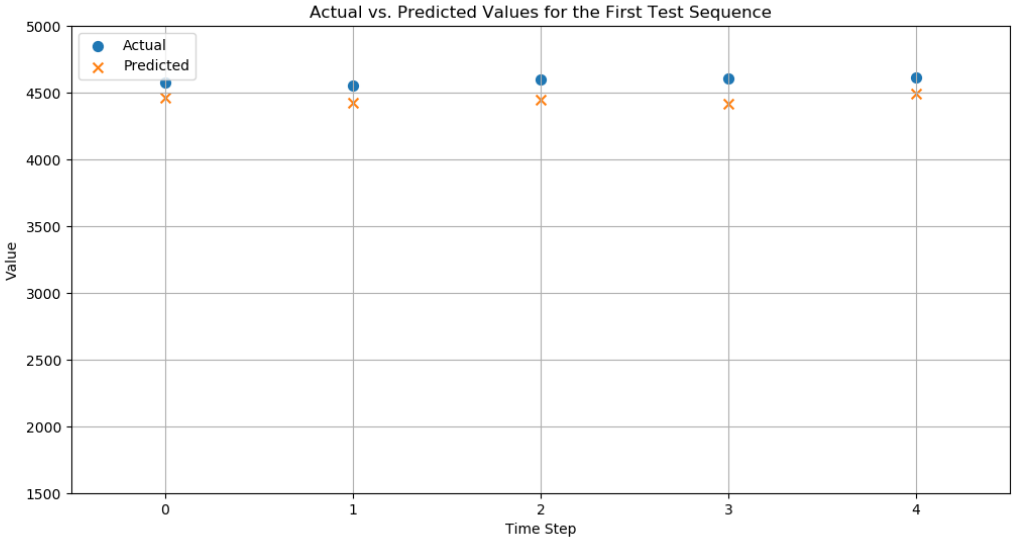


Figure 5.3 - Prediction Graph: Comparative Analysis of Predicted vs. True Values Using Top 3 Variables with 5 lags on the XGBoost Model

In Figure 5.3, we can observe a noteworthy improvement in the alignment between the predicted and actual values compared to previous results. The graph shows a more refined and accurate approximation, indicating that the model's predictions are now tracking the actual S&P 500's price movements more closely.

The successful integration of the top 3 positive influential variables into the XGBoost model showcases their importance in capturing relevant signals in the financial markets. This enhancement not only contributes to a more accurate prediction of the S&P 500's price but also highlights the potential benefits of utilizing key exogenous variables in bolstering the model's overall performance.

5.5. Comparison of LIME results

The LIME analysis was performed on both the LSTM and XGBoost models in order to compare the importance of exogenous variables in predicting the S&P 500's price. For the 10 most confident instances, the LSTM model identified BTC, Dji, Eth and Gold as the four most important variables. In contrast, the XGBoost model ranked Dji as the top variable, followed by Oil, and Eth. Interestingly, both models indicated negative importance for BTC, suggesting a counteractive effect on the predicted S&P 500's price.

Comparing the results, it is notable that Dji consistently emerged as an important variable in both models. This indicates the significance of the Dow Jones Industrial Average in predicting the S&P 500's price across different modeling techniques. However, there were variations in the rankings and inclusion of other variables. The LSTM model gave higher importance to Gold as the variable with the most positive impact (and the only one with positive weighted impact), while the XGBoost model emphasized Dji, Oil and Eth. Additionally, both models shared the observation of negative importance for BTC, implying a distinct behavior compared to other variables.

These disparities in variable importance highlight the differences in modeling approaches and algorithms employed by LSTM and XGBoost. Each model might capture unique aspects of the data and exhibit varying sensitivities to different exogenous variables. Therefore, the comparison of LIME results provides valuable insights into the strengths and limitations of each model and the factors they prioritize when predicting the S&P 500's price. The full LIME output can be found in Appendix C.

CHAPTER 6

Conclusion

In this work, we implemented two models, LSTM and XGBoost, to predict the S&P 500's price using a set of exogenous variables. Both models exhibited promising results in predicting the S&P 500's price given the set of exogenous variables, but their performance characteristics and interpretability differed due to their unique modeling approaches.

This thesis consists of two parts, with the first one being the model development and S&P 500's price prediction and the second part being the explanation of the prediction made in the first part.

In the model development phase, we aimed to find the optimal model by creating a base model with basic and default hyperparameters for both LSTM and XGBoost. This base model served as a starting point for further improvement during the hyperparameter tuning phase. By training these initial models, we gained valuable insights into their performance on the dataset and identified areas for enhancement. We used common evaluation metrics, such as MAE, MSE, MAPE and RMSE.

After extensive hyperparameter tuning and cross-validation, we fine-tuned the LSTM and XGBoost models to achieve the best possible performance. Careful consideration was given to balancing the models' complexity and avoiding overfitting to ensure generalization to unseen data. Through this iterative process, we arrived at the optimal models, which exhibited superior predictive power compared to the base models.

With the optimal models in place, we shifted our focus to the explainability phase, a critical step in building trust and understanding in the predictive models. The primary objective during this phase was to determine the variables that had the most significant impact on the models' predictions and to understand how a variable influenced a model and also how the variable that influenced one model also affected the other one.

Through the application of LIME and SHAP techniques, we gained valuable insights into the importance of exogenous variables for predicting the S&P 500's price in each model. Notably, Dow Jones Industrial Average consistently emerged as one of the most influential variables in both LSTM and XGBoost models, reaffirming its significance in capturing overall market sentiment and investor confidence.

However, intriguing differences arose in the variable importance rankings between the two models. For the LSTM model, Dji, Btc, Eth and Gold were identified as important variables, with Gold rising as the only variable with positive weighted impact in the model,

reflecting its presence on predicting the S&P 500's price within the context of sequential data analysis. In contrast, XGBoost didn't even consider Gold as an important variable in all the cases where it felt most confident about the prediction. This, along with all the other variable rankings made by the two models and the degree of difference between the variables' importance according to each model, goes to show that a variable considered to be the most influential in one model might have negative or even zero impact in the other model.

The XGBoost model further emphasized the importance of additional variables, including Dji, Oil, Eth, Btc and Silver, with the first three being positive in both weighted and regular impact underscoring their collective influence on predicting the S&P 500's price in this modeling approach.

Another interesting finding was that Bitcoin consistently appeared negatively important in both models, suggesting a counteractive effect on the predicted S&P 500's price. This observation implies that the behavior of the Bitcoin market has a distinct impact on stock market movements, irrespective of the modeling technique used.

While both LSTM and XGBoost models effectively demonstrated the ability to leverage exogenous variables, they exhibited varying sensitivities to different features. The LSTM model showcased its strength in capturing temporal dependencies, while the XGBoost model excelled in handling tabular data with non-linear relationships.

We acknowledge the limitations of these models, such as the computational complexity of LSTM and the sensitivity of XGBoost to feature engineering and hyperparameter tuning, which may present challenges in real-time applications. Additionally, while LIME and SHAP provided valuable local interpretability, further research is needed to enhance global interpretability and comprehend the models' behavior across the entire dataset.

In conclusion, our study sheds light on the predictive capabilities of LSTM and XGBoost models in the context of the S&P 500's price prediction. The comparison of LIME and SHAP results deepened our understanding of some commodities' importance in decision-making processes in each model. As we navigate the intricate landscape of financial markets, this research contributes to informed decision-making and inspires future advancements in predictive modeling and interpretability.

References

Agarwal, A., Bhatia, A., Malhi, A., Kaler, P., & Pannu, H. S. (2022, July). Machine Learning Based Explainable Financial Forecasting. In *2022 4th International Conference on Computer Communication and the Internet (ICCCI)* (pp. 34-38). IEEE.

Balagopalan, A., Zhang, H., Hamidieh, K., Hartvigsen, T., Rudzicz, F., & Ghassemi, M. (2022). The Road to Explainability is Paved with Bias: Measuring the Fairness of Explanations. *arXiv preprint arXiv:2205.03295*.

Dang, T., Van, H., Nguyen, H., Pham, V., & Hewett, R. (2020, July). Deepvix: Explaining long short-term memory network with high dimensional time series data. In *Proceedings of the 11th international conference on advances in information technology* (pp. 1-10).

Erdas, M. L., & Caglar, A. E. (2018). Analysis of the relationships between Bitcoin and exchange rate, commodities and global indexes by asymmetric causality test. *Eastern Journal of European Studies*, *9*(2).

Confalonieri, R., Coba, L., Wagner, B., & Besold, T. R. (2021). A historical perspective of explainable Artificial Intelligence. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *11*(1), e1391.

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785-794).

Çelik, T. B., İcan, Ö., & Bulut, E. (2023). Extending machine learning prediction capabilities by explainable A.I. in financial time series prediction. *Applied Soft Computing*, *132*, 109876.

Freeborough, W., & van Zyl, T. (2022). Investigating explainability methods in recurrent neural network architectures for financial time series data. *Applied Sciences*, *12*(3), 1427.

Frazier, P. I. (2018). A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811*.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Hoffman, R. R., Mueller, S. T., Klein, G., & Litman, J. (2018). Metrics for explainable A.I.: Challenges and prospects. *arXiv preprint arXiv:1812.04608*.

Kilian, L., & Park, C. (2009). The impact of oil price shocks on the U.S. stock market. *International economic review*, *50*(4), 1267-1287.

Kumar, A. S., & Padakandla, S. R. (2022). Testing the safe-haven properties of Gold and bitcoin in the backdrop of COVID-19: a wavelet quantile correlation approach. *Finance research letters*, *47*, 102707.

Liu, C., Wang, J., Xiao, D., & Liang, Q. (2016). Forecasting s&p 500 stock index using statistical learning models. *Open journal of statistics*, *6*(6), 1067-1075.

Liu, J., Tripathi, S., Kurup, U., & Shah, M. (2019, December). Auptimizer-an extensible, open-source framework for hyperparameter tuning. In *2019 IEEE International Conference on Big Data (Big Data)* (pp. 339-348). IEEE.

Luo, J., Zhang, Z., Fu, Y., & Rao, F. (2021). Time series prediction of COVID-19 transmission in America using LSTM and XGBoost algorithms. *Results in Physics*, *27*, 104462.

Machlev, R., Heistrene, L., Perl, M., Levy, K. Y., Belikov, J., Mannor, S., & Levron, Y. (2022). Explainable Artificial Intelligence (XAI) techniques for energy and power systems: Review, challenges and opportunities. *Energy and A.I.*, *9*, 100169.

Mangalathu, S., Hwang, S. H., & Jeon, J. S. (2020). Failure mode and effects analysis of R.C. members based on machine-learning-based SHapley Additive exPlanations (SHAP) approach. *Engineering Structures*, *219*, 110927.

Maree, C., Modal, J. E., & Omlin, C. W. (2020, December). Towards responsible A.I. for financial transactions. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)* (pp. 16-21). IEEE.

Misheva, B. H., Osterrieder, J., Hirsa, A., Kulkarni, O., & Lin, S. F. (2021). Explainable A.I. in credit risk management. *arXiv preprint arXiv:2103.00949*.

Nevendra, M., & Singh, P. (2022). Empirical investigation of hyperparameter optimization for software defect count prediction. *Expert Systems with Applications*, *191*, 116217.

Ohana, J. J., Ohana, S., Benhamou, E., Saltiel, D., & Guez, B. (2021). Explainable A.I. (XAI) models applied to the multi-agent environment of financial markets. In *Explainable and Transparent A.I. and Multi-Agent Systems: Third International Workshop, EXTRAAMAS 2021, Virtual Event, May 3–7, 2021, Revised Selected Papers 3* (pp. 189-207). Springer International Publishing.

Park, S., & Yang, J. S. (2022). Interpretable deep learning LSTM model for intelligent economic decision-making. *Knowledge-Based Systems*, *248*, 108907.

Rojat, T., Puget, R., Filliat, D., Del Ser, J., Gelin, R., & Díaz-Rodríguez, N. (2021). Explainable artificial intelligence (xai) on timeseries data: A survey. *arXiv preprint arXiv:2104.00950*.

Schratz, P., Muenchow, J., Iturritxa, E., Richter, J., & Brenning, A. (2019). Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data. *Ecological Modelling*, *406*, 109-120.

Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2018, December). A comparison of ARIMA and LSTM in forecasting time series. In *2018 17th IEEE international conference on machine learning and applications (ICMLA)* (pp. 1394-1401). IEEE.

Zhang, L., Bian, W., Qu, W., Tuo, L., & Wang, Y. (2021, April). Time series forecast of sales volume based on XGBoost. In *Journal of Physics: Conference Series* (Vol. 1873, No. 1, p. 012067). IOP Publishing.

Wang, Z., Ma, Y., Liu, Z., & Tang, J. (2019). R-transformer: Recurrent neural network enhanced transformer. *arXiv preprint arXiv:1907.05572*.

Wu, C., Xue, X., & Song, Y. (2022, May). Research on Cancer Diagnosis Method Based on LightGBM-Gridsearchcv. In *Proceedings of the 4th International Conference on Big Data Engineering* (pp. 122-126).

Appendix A

	Spx	Gold	Btc	Dji	Eth	Copper	Oil	Gas	Silver	USD
count	1563.000000	1563.000000	1563.000000	1563.000000	1563.000000	1563.000000	1563.000000	1563.000000	1563.000000	1563.000000
mean	3282.708394	1552.489443	17620.445425	27837.383826	992.669750	3.276962	62.417562	3.495278	19.450898	96.304727
std	716.958189	260.506051	16739.834593	4609.579161	1152.545468	0.651926	19.038635	1.656967	4.094211	4.863314
min	2237.400000	1163.200000	778.600000	18591.930000	9.610000	2.179000	-37.630000	1.482000	11.772000	88.170000
25%	2711.595000	1289.100000	6460.350000	24591.400000	194.605000	2.752500	50.825000	2.584500	16.402500	92.946000
50%	3005.470000	1531.400000	9683.200000	26680.870000	382.410000	3.060000	59.330000	2.925000	17.781000	95.864000
75%	3915.985000	1798.350000	23566.500000	32037.300000	1541.055000	3.694500	71.025000	3.807000	23.341500	98.110000
max	4796.560000	2058.400000	67526.000000	36799.650000	4808.380000	4.839500	123.700000	9.647000	29.418000	114.047000

Table A: Dataset's spectrum

Appendix B

Bayesian optimization

Bayesian optimization is a more advanced technique composed of two components: a Bayesian statistical model for modeling the objective function and an acquisition function for deciding where to sample next (Frazier, 2018). The model was then evaluated over the set of parameters suggested by the Bayesian algorithm using the same metrics as on the base model (MAE, MASE, RMSE, MAPE), which can be seen in Table 4.2. Similarly to the Random Search optimization process, as seen in Figures B1 to B4, loss graphs were plotted to visualize each evaluated configuration's convergence and training progress.

Figure B.1: Bayesian loss graph for 5 past observations

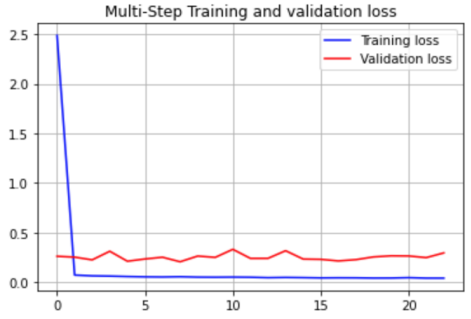


Figure B.2: Bayesian loss graph for 10 past observations

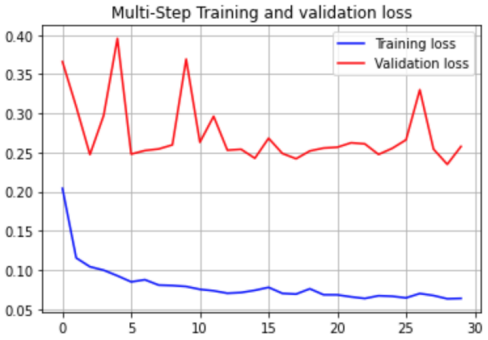


Figure B.3: Bayesian loss graph for 22 past observations

Figure B.4: Bayesian loss graph for 60 past observations

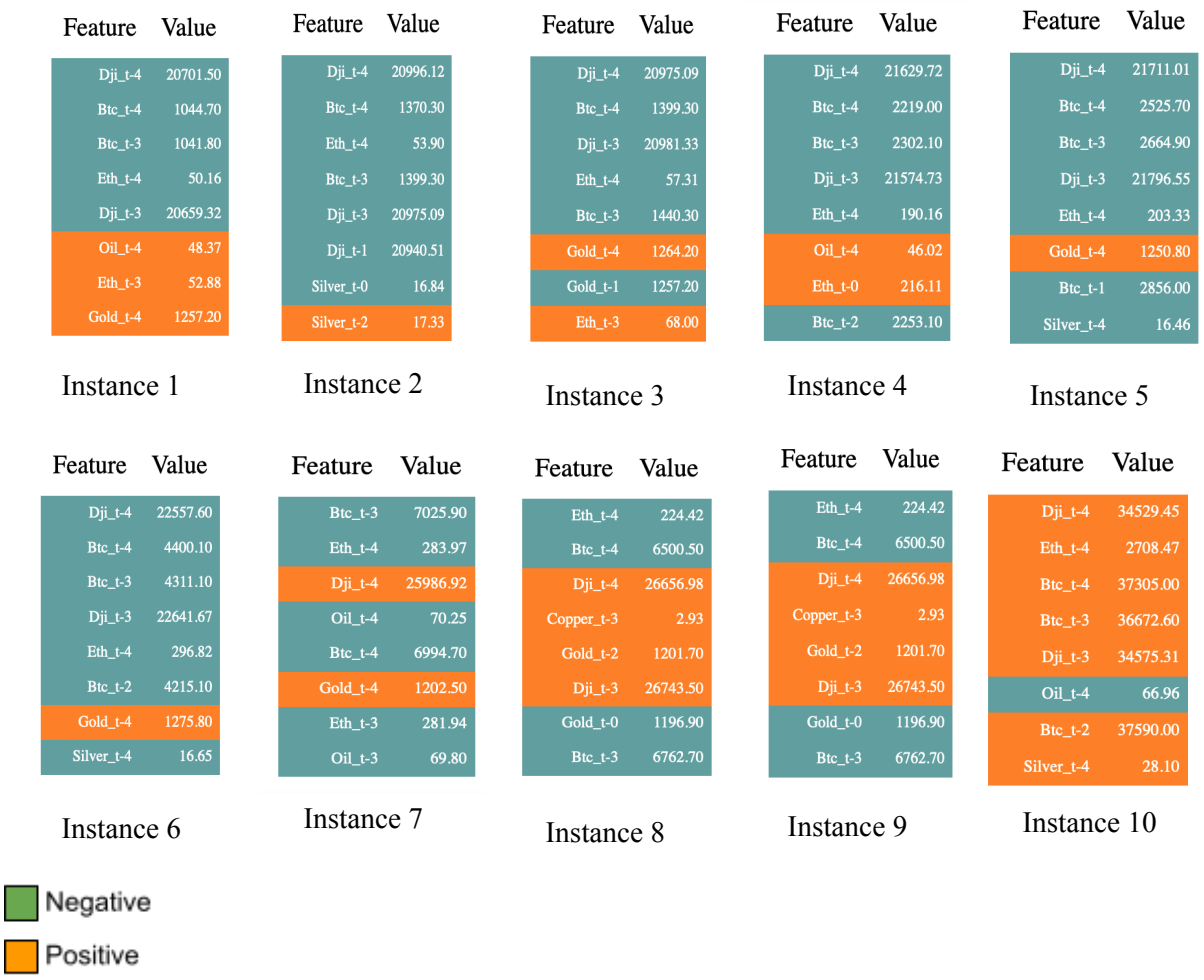
The four depicted graphs showcase the intriguing and unconventional behavior of the model during the training and validation process.

Importantly, these observed graph patterns align with the corresponding metrics, indicating a poor approximation of the validation set to the training set. This implies that the model struggles to generalize well beyond the training data, resulting in suboptimal performance on unseen examples.

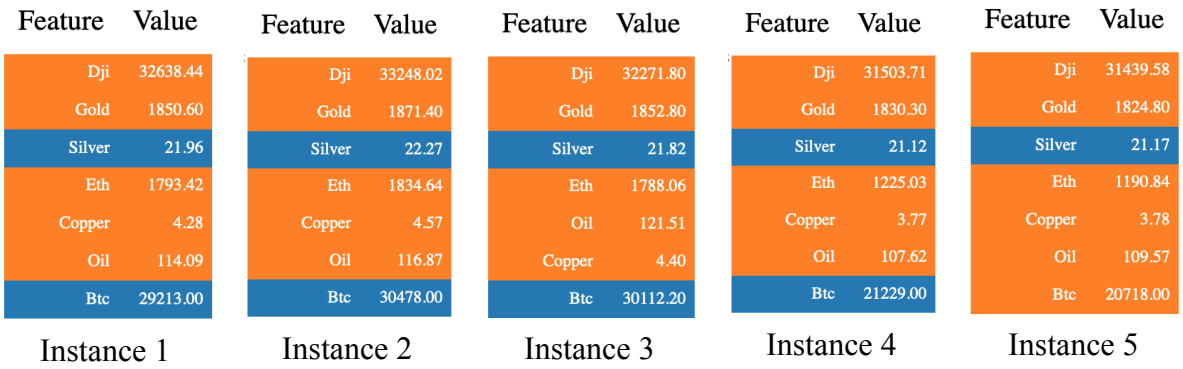
Further analysis and evaluation are necessary to identify the underlying causes of these unusual behaviors and develop potential strategies to enhance the model's performance and stability. One hypothesis of this abnormal performance is that the hyperparameters suggested by the Bayesian algorithm are not suited for this problem's domain, dataset characteristics, and/or model architecture.

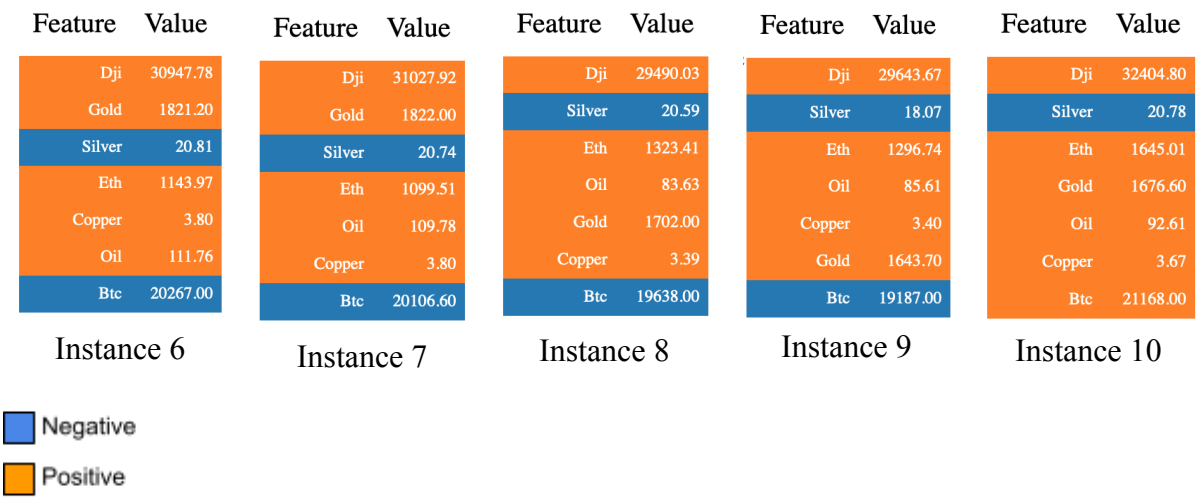
Appendix C

Lime feature importance for the LSTM Model



Lime feature importance for the XGBoost Model




Feature Value
Dji 30947.78
Gold 1821.20
Silver 20.81
Eth 1143.97
Copper 3.80
Oil 111.76
Btc 20267.00
Instance 6
Feature Value
Dji 31027.92
Gold 1822.00
Silver 20.74
Eth 1099.51
Oil 109.78
Copper 3.80
Btc 20106.60
Instance 7
Feature Value
Dji 29490.03
Silver 20.59
Eth 1323.41
Oil 83.63
Gold 1702.00
Copper 3.39
Btc 19638.00
Instance 8
Feature Value
Dji 29643.67
Silver 18.07
Eth 1296.74
Oil 85.61
Copper 3.40
Gold 1643.70
Btc 19187.00
Instance 9
Feature Value
Dji 32404.80
Silver 20.78
Eth 1645.01
Gold 1676.60
Oil 92.61
Copper 3.67
Btc 21168.00
Instance 10
Negative
Positive