COMPARISON OF MACHINE LEARNING MODELS AND THEIR PREDICTIVE POWER WITH REGARD TO CRYPTOCURRENCY PRICE MOVEMENTS

Savas Murt

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY
DEPARTMENT OF COGNITIVE SCIENCE & ARTIFICIAL INTELLIGENCE
SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
TILBURG UNIVERSITY

STUDENT NUMBER

1269645

THESIS COMMITTEE

Supervisor:
dr. Peter Hendrix

Second reader:
dr. Henry Brighton

LOCATION

Tilburg University
School of Humanities & Digital Sciences
Department of Cognitive Science &
Artificial Intelligence
Tilburg, The Netherlands

DATE

April 7, 2023

WORD COUNT

8794

Contents

COMPARISON OF MACHINE LEARNING MODELS AND THEIR PREDICTIVE POWER WITH REGARD TO CRYPTOCURRENCY PRICE MOVEMENTS

Savas Murt

Abstract

Cryptocurrencies are attracting more institutional investors due to their potential profitability, despite the need for a clear fundamental framework to anticipate their prices. Although they share specific characteristics with traditional asset classes, the extreme volatility and shortcomings present challenging issues in developing an accurate forecast method. This thesis addresses the lack of comparisons in the existing literature regarding machine learning algorithms in predicting cryptocurrency prices. This thesis aims to discover which machine learning models perform best in predicting cryptocurrency prices. Therefore, a comparison is made between Random Forest (RF), eXtreme Gradient Boosting (XGBoost), Multilayer Perceptron (MP), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM) models in the prediction of the daily close price of the 10 largest cryptocurrencies in terms of market capitalization, using a 3-fold TimeSeriesSplit cross-validation technique for the period from November 9, 2017, until November 9, 2022. This research also includes technical features such as the return of the selected cryptocurrencies and asset-based features like the volatility index, S&P 500, and NASDAQ returns. The findings demonstrate that the trained models perform substantially better than the baseline RF model. This study concludes that the LSTM model performs best, while the RNN model performed second to best, given their performance on the MAE and RMSE evaluation metrics. The MLP model placed third, followed by the XGBoost model, with the latter failing to outperform the baseline RF model on at least one occasion.

Data Source/Code/Ethics Statement

Work on this thesis did not involve collecting data from human participants or animals. The historical data for cryptocurrencies and other market-related data are obtained from Yahoo Finance. However, the author of this thesis acknowledges that they do not have any legal claim to this data. Furthermore, the code used in the thesis is not publicly available.

1. INTRODUCTION

1.1 Problem Statement

Investors have well-established frameworks for evaluating traditional asset classes such as equities, fixed income, foreign exchange, real estate, and commodities, whether it is fundamental or technical analysis. However, no clear framework exists to predict future prices of the relatively new asset-class cryptocurrencies, which increasingly gains the involvement of institutional investors (Huang et al., 2022). Technological improvements accompanied by investor interest in seeking new investment options have been significant factors in the birth of a wide range of cryptocurrencies. As a result, the overall market value of cryptocurrencies has reached astounding heights ever since. Because of this, the use of machine learning models for predicting cryptocurrency prices has become increasingly popular, primarily due to their potential profitability. However, as cryptocurrencies become more broadly recognized in the academic world with numerous published papers and in practice, it remains nascent for investors seeking validation of their investment thesis.

There is already a substantial devotion to price prediction using machine learning models for the traditional asset classes or Bitcoin, the most popular cryptocurrency introduced by Satoshi Nakamoto (2009). Although they share specific characteristics with more conventional asset classes, cryptocurrency price movements are characterized by extreme volatility, and establishing an accurate cryptocurrency prediction model is challenging. In 2022, Fang et al. outlined the state-of-the-art by covering 146 research papers on various areas of cryptocurrencies and gave a thorough assessment of the field. The prevalent machine learning techniques in this discipline include Random Forest (RF), XGBoost, Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) (Fang et al., 2022). In the literature review section of this thesis, the most recent work, and common approaches for machine learning-based methods on cryptocurrency prices were further identified and analyzed. Some studies examine and compare several machine learning models, like Jaquart, Dann, and Weinhardt (2021), but the previous studies are mainly focused on Bitcoin or other prominent cryptocurrencies like Ethereum. This is a great chance to stand back, evaluate the current level of research in this area, and identify research gaps that may benefit from further study. As a result, we identified that no study extensively compares different machine learning methods across various cryptocurrencies.

1.2 Research Goal and Relevance

As previously mentioned, the earlier studies were mainly focused on Bitcoin or other prominent cryptocurrencies, and even if there are studies that analyze smaller cryptocurrencies, it is futile to compare them. This is because prior research utilizes different periods, objectives, input variables, model parameters, and assessment metrics. Therefore, comparing these prediction models across papers is impractical. The problem statement that this research seeks to address is the lack of comparisons in the existing literature regarding machine learning models in predicting cryptocurrency prices. Therefore, in this thesis, a comparison will be made of five machine learning models and their predictive power regarding cryptocurrency price movements. This will be done by combining the domains of data science and cryptocurrency to get a deeper understanding of machine learning techniques and this relatively new asset class.

Depending on the situation and particular applications, this research is relevant from a scientific and societal standpoint. The scientific relevance is two-fold, first the understanding of market dynamics because cryptocurrencies are traded on open markets, and their prices change quickly depending on a variety of variables. Machine learning algorithms can assist academics in gaining insight into the underlying market dynamics and test different economic theories and models by evaluating enormous volumes of historical pricing data and detecting patterns and connections. In addition, machine learning algorithms may be trained on extensive historical cryptocurrency price data to create predictive models that can anticipate future prices with varied degrees of accuracy. These models may be examined using actual data and improved over time, potentially resulting in more accurate financial market forecasting techniques. Moreover, this research is important from a societal standpoint because cryptocurrencies have become a well-liked asset class in the broader society. Some people spend substantial sums of money in the expectation of making money off price changes. However, as stated before, cryptocurrencies are a risky investment for both people and institutions because of their extreme volatility, uncertainties, and no clear framework for their predictability. Correct cryptocurrency price forecasts can aid traders and investors in making more educated choices about whether to purchase, sell, or hold certain assets, potentially improving the performance of their investments. Furthermore, the market behavior of cryptocurrencies can have wider regulatory and policy ramifications, such as influencing tax collection, national security, or financial stability. Regulators can analyze cryptocurrency prices with machine learning algorithms to monitor and understand the impact of these assets on the larger economy and make well-informed choices about how to regulate or control them.

1.3 Research Questions and Strategy

This study will investigate different machine learning models together with historical cryptocurrency and asset-based data to predict the closing price of the selected cryptocurrencies and therefore aims to answer the following research question:

"*How well can the price of selected cryptocurrencies be predicted with machine learning methods?"*.

Two related sub-questions are created to address the main research question.

Sub-question 1: "*Which of a set of selected machine learning algorithms performs well in the prediction of cryptocurrency prices?"*

This sub-question will be answered by comparing five machine learning algorithms, including Random Forest, XGBoost, Multilayer Perceptron, Recurrent Neural Network, and Long Short-Term Memory, further elaborated in the methodology section. Moreover, both technical and asset-based features are used in the models to predict the close price, as mentioned in Jaquart, Dann, and Weinhardt (2021) and further outlined in the experimental setup section. Finally, two standard evaluation metrics are used to assess each model's performance, further outlined in the evaluation metrics section. In this paper, the machine learning algorithm is considered to perform well if it achieves a better MAE and RMSE score than the baseline model.

Sub-question 2: "*How does the predictive power of different machine learning algorithms compare across cryptocurrencies?*"

The second sub-question is answered by comparing the models, and their predictive power is then compared with the selected cryptocurrencies. The cryptocurrencies that are compared are Bitcoin (BTC), Ethereum (ETH), Binance Coin (BNB), Ripple (XRP), Dogecoin (DOGE), Cardano (ADA), Polygon (MATIC), Polkadot (DOT), Litecoin (LTC), Solana (SOL). Since different cryptocurrencies are used, there is no true apples-to-apples comparison for the MAE and RMSE scores. Therefore, an overview of the MAE and RMSE scores across periods of different models and coins was highlighted to see whether a model performed the best for one particular cryptocurrency and whether the predictive power was lacking (compared to other models) for the other cryptocurrency. This will give insight into the predictive power of different machine-learning algorithms across cryptocurrencies.

1.4 Findings

The findings demonstrate that the trained models perform substantially better than the baseline RF model. Furthermore, from this research, the LSTM model performs the best, while the RNN model performed second to best, given their performance on the MAE and RMSE evaluation metrics. Finally, the MLP model placed third, followed by the XGBoost model, with the latter model failing to outperform the baseline RF model on at least one occasion.

2. LITERATURE REVIEW

Predicting the future price using machine learning techniques is well documented for traditional asset classes.[1] However, the studies about machine learning techniques for cryptocurrencies mainly focused on Bitcoin (Fang et al., 2022). While the traditional markets have been around for decades, the cryptocurrency market is relatively less mature and limited in scope for most coins. Fang et al. (2022) came up with a broad survey of earlier studies on the cryptocurrency market focusing on the price prediction. Their paper presents an overview of the current state-of-the-art machine-learning techniques used to predict the price of cryptocurrencies and helps us identify the research gap this thesis aims to fill. Their research covers 146 papers from the cryptocurrency space, and most research findings are predominantly focused on basic regression, time-series methods, and decision trees, including Random Forest (RF) and XGBoost. Their study also states that RNN and LSTM prevail among the NN algorithms.

Jaquart, Dann, and Weinhardt (2021) compared six different machine learning methods (these include models like LSTM, RNN, and RF) for short-term predictability (one-minute to an hour) in Bitcoin prices. In their study, they made a categorization of the features, which they separated into four groups. These groups include technical-based features related to the history of a specific coin (e.g., returns or volume). Asset-based features, which are comprised of traditional asset classes (e.g., return of indexes like S&P 500

[1] The search result on Google Scholar for the price prediction using machine learning for stocks yielded more than 100 thousand results, while similar search results for cryptocurrencies were less than 20 thousand.

and commodities like gold), and blockchain-based features indicate specific features related to the coin's blockchain network (e.g., transactions and number of coins in time) and sentiment-based features which are associated with the sentiment (e.g., Twitter and Google searches). According to their research, the performance for predictability of BTC prices increases for longer horizons, and the results of the RNN and LSTM proved to be well-suited for Bitcoin price prediction.

Unlike the previous paper, the study by Mudassir et al. (2020) presents machine-learning regression models for both short- and medium-term changes in the price of Bitcoin. While most of the previous papers focused on short-term prediction (one-day and less), the authors explored the prediction of Bitcoin prices using a horizon of one to ninety-days. Their findings show that the provided models performed much better for short-term than longer-term horizons. Their results contradict and dispute the previously mentioned study, where the performance in predicting BTC prices increases for longer horizons.

Fleischer et al. (2022) compared the LSTM model against the ARIMA in predicting the future closing prices of several cryptocurrencies by only using the past closing price as an input feature. The authors used the RMSE score as a comparison, and the LSTM RSME results were as follows: Bitcoin (1,334.755), Dogecoin (0.007), and Ethereum (117.655). One of those studies that expand upon the latter study is Hansun et al. (2022), which also included the MAE scores of the following trading pairs BTC-USD (MAE; 1,617.75 and RMSE; 2,518.02), ETH-USD (MAE; 103.18 and RMSE; 150.09), ADA-USD (MAE; 0.13 and RMSE; 0.19) and BNB-USD (MAE; 18.08 and RMSE; 27.62). Moreover, Ammer and Aldhyani (2022) also used a multivariate LSTM model to predict closing prices. These studies included the open, high, low, close, and volume as features, replacing any missing values used in those features with the most recent available data. Moreover, Mohta et al. (2022) used machine learning techniques like RNN and LSTM to predict both short- and long-term close prices of Ethereum. Their research result showed that the error metrics (RMSE and MAE) become larger if the prediction duration increases. Just like the research of Mudassir et al. (2020), these results contradict and dispute the study of Jaquart, Dann, and Weinhardt (2021), where the performance in predicting the cryptocurrency price increases for longer horizons.

Chen (2022) and Tandon et al. (2019) compared NN models like LSTM and RNN with the Random Forest model. The RF model served as an excellent baseline model to assess whether the MAE and RMSE scores of the NN models performed better in Bitcoin price prediction.

Many alternative cross-validation techniques are proposed in the prediction of cryptocurrency prices. Oyewola et al. (2022) analyzed a "hybrid walk-forward ensemble optimization technique and applied it to predict the daily prices of fifteen cryptocurrencies" (p. 2). Since standard cross-validation techniques like k-fold or leave-one-out are not suited for time series, the authors suggested an improved version of the walk-forward cross-validation. Barnwal et al. (2019) researched the Bitcoin price direction using several different technical indicators, models, and two cross-validation methods (walk forward expanding window and purged cross-validation). They concluded that the latter leads to better accuracy. Erfanian et al. (2022) compared different machine learning models and investigated the importance of several indicators (asset- and blockchain-based) for Bitcoin price prediction. They also included a 10-fold/period rolling basis cross-validation method. The evaluation metric scores (R-squared and Root Mean Squared Error) are averaged across each period and used as their final evaluation metric. Their results yielded better performance when increasing the amounts of folds/periods.

In contrast, Cocco et al. (2021) applied cross-validation on expanding basis by using three folds/periods instead of 10. The best model was determined by calculating the Mean Absolute Percentage Error (MAPE) of each period and by taking the averaging across all the periods. As cited by Kuhn & Johnson (2019), the k-fold cross-validation folds are usually either 5 or 10. However, there are no clear guidelines on choosing the cross-validation method and the number of folds/periods. The studies above have contradicting results on predictability when increasing or decreasing the number of folds. However, if the number of folds in cross-validation increases, so is the computational requirement. Therefore, choosing fewer periods is better from a practical point of view.

Bouri et al. (2017) and Su et al. (2022) researched the correlation between Bitcoin and the fear index, commonly known as the volatility index (VIX), which is measured using the implied volatility across S&P 500 index. Their studies show that the Bitcoin price and the VIX index have an inverse relationship. Wang et al. (2022) and Nguyen (2022) show that Bitcoin prices positively correlate to traditional risky assets like stocks and conclude that the correlation increases under extreme shocks and high uncertainty. However, this contradicts and disputes the findings of Shahzad et al. (2019) and Al-Yahyaee et al. (2019), where the results suggest that Bitcoin can behave as a haven during uncertain times and that it could provide diversification benefits to traditional assets like stocks.

The studies above only used a few techniques, models, and cryptocurrencies. Therefore, a clear comparison between various machine-learning approaches across various cryptocurrencies has not yet been made. This research expands on the previous research and links several machine learning models and techniques with various cryptocurrencies to get more insight into the predictive power regarding cryptocurrency price movements. Moreover, a commonality shared by most of these papers is the use of the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) to compare the performance of the projected models, further outlined in section 4. Taking every paper into consideration, including Fang et al. (2022) broad survey results, the RF, XGBoost, MLP, RNN, and LSTM models will be compared for the set of selected cryptocurrencies mentioned in Table 1 and include cryptocurrencies that are less researched, like Solana and Polkadot to fill the gap with earlier studies.

3. METHODOLOGY

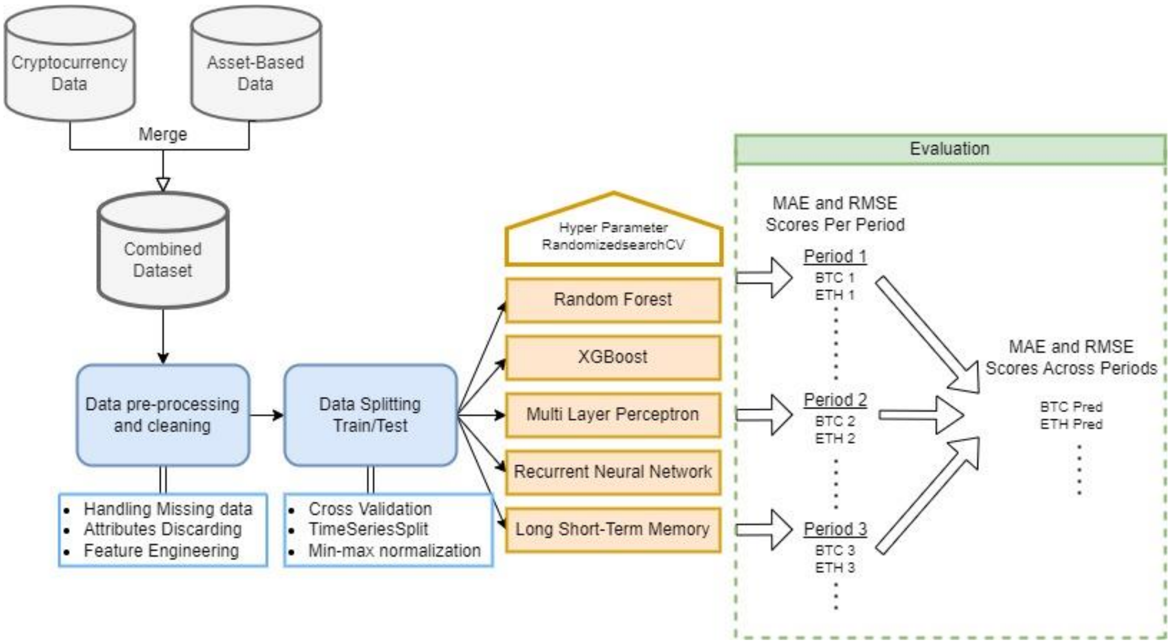
This section included the data science flow chart, machine learning models and the reason why we chose these models. The data science flow chart is shown first to provide an understanding of the data science pipeline and the steps taken.

3.1 Data Science Flow Chart

The models and actions taken in the data science pipeline are depicted in Figure 1. These are further outlined in the section 4 (Experimental Setup).

FIGURE 1

DATA SCIENCE FLOW CHART



3.2 Random Forest

Random forest (RF) uses an ensemble technique by constructing decision trees that can be applied to regression and classification tasks (Kumar, 2022). *Ensemble techniques* are methods that improve the machine learning model prediction. Specifically, RF uses bagging and serves as an extension to this technique, as Breiman (2001) proposed. In the RF model, the trees run parallel to each other, whereby the predictions of the trees are combined, and the average of all the trees is used as the final RF model prediction. To compare the performance of different models with each other, the RF will be used as the baseline model. The RF model is used as the baseline because it is the simplest model among the ones being compared. Despite its simplicity RF model is a well-liked option for price prediction. Moreover, previous studies like Chen (2022) and Tandon et al. (2019) also used the RF model as their baseline.

3.3 Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) uses decision trees like RF, but the difference is that XGBoost uses boosting as the ensemble technique. XGBoost and RF are solid and well-known in machine learning algorithms, but they vary in a few ways that make XGBoost the preferable option. Moreover, the studies that compared the XGBoost model proved better results on the evaluation metrics than the RF model. For the XGBoost model, the trees run sequentially to each other because of the boosting ensemble technique, which makes it suitable to handle data effectively since it is built to be scalable, as opposed to the RF model. Moreover, XGBoost uses second-order derivatives to optimize the loss function, allowing it to converge more quickly and prevent overfitting, as stated more effectively in Chen & Guestrin (2016).

3.4 Multilayer Perceptron

Multilayer Perceptron (MLP) is a form of artificial neural network (ANN) that interconnects a group of nodes in a feed-forward direction, the so-called Feedforward neural network (FNN). So, each neuron (perceptron) in the first layer (input layer) has a direct connection to the second layer (hidden layer) and, after that, the third layer (output layer), which represents the results. Overall, MLP is more complex than the models above and an effective method for predicting cryptocurrency prices. This is because it can capture intricate correlations between input and output layers, extract important features from data, and deal with noise and scale to accommodate vast volumes of data. In addition, since cryptocurrency prices fluctuate a lot, occasionally seeing sharp jumps or drops. Outliers may be handled by MLP models without having a substantial impact on the model's performance.

3.5 Recurrent Neural Network

Recurrent Neural Network (RNN), just like the MLP model, is a form of ANN that is also interconnected with a group of nodes. However, the difference is that RNN nodes are not connected feedforward (one way only) but can go both directions (recurrent). This also allows the RNN model to have an internal memory of the input. RNNs are a particular kind of NN that function well for sequential data, including time series data. RNNs contain a feedback loop that enables them to keep track of prior inputs, which is crucial for problems involving sequence prediction. Therefore, RNNs are a viable option for forecasting time series data, such as cryptocurrency prices, and studies like Jaquart, Dann, and Weinhardt (2021) and Chen (2022) proved promising results.

3.6 Long Short-Term Memory

Long Short-Term Memory (LSTM) is a form of RNN, and just like RNN, it also maintains an internal memory of the input. However, the difference is that LSTM maintains that memory longer. An LSTM model can hold this information for an extended period because it uses input, output, and forget gates. These gates can, in turn, control the flow of information. Since LSTM can manage long-term dependencies, it is better than typical RNN. In addition, this makes it possible for LSTM to successfully model the intricate interactions between input and output variables, which is crucial for correctly forecasting the price of cryptocurrencies. Like the RNN model, LSTM showed promising results, although the latter model had better predictions in previous studies like Mohta et al. (2022) and Ammer and Aldhyani (2022).

4. EXPERIMENTAL SETUP

4.1 Cryptocurrency data

The 10 most valuable cryptocurrencies, measured by market capitalization, were chosen. The data was collected as a CSV file from Yahoo Finance's Application Programming Interface (API).[2] The following cryptocurrencies are excluded from the analysis: Tether (USDT), USD Coin (USDC), Binance USD (BUSD), and Dai (DAI), even though these cryptocurrencies are in the top ten in terms of market capitalization. These stablecoins are excluded because their values are pegged to the US dollar, and it would not be

[2] Based on the market capitalization data from https://coinmarketcap.com/ as of November 9, 2022.

informative to analyze them in this research. The cryptocurrencies that will be analyzed in this research and their category are shown in Table 1. As previously stated in the literature review, unlike Bitcoin, less researched cryptocurrencies like Solana or Polkadot, among others, are included. The start and end date and the number of daily observations differ for each cryptocurrency since some projects were launched at a later stage. Moreover, the maximum data for the selected cryptocurrencies will be set at no more than five years due to the fact of extreme volatility in the emergence phase of the cryptocurrency market when crypto was less mature. Additionally, to overcome the discrepancies between cryptocurrencies that were launched at a later stage.

TABLE 1

CRYPTOCURRENCY DATA

Cryptocurrencies	Ticker	Category	Trading Pair	Start Date	End Date	No. of Daily Data
Bitcoin	BTC	Digital currency	BTC-USD	9-Nov-2017	9-Nov-2022	1,827
Ethereum	ETH	Blockchain network	ETH-USD	9-Nov-2017	9-Nov-2022	1,827
Binance Coin	BNB	Native coin of Binance exchange and ecosystem	BNB-USD	9-Nov-2017	9-Nov-2022	1,827
Ripple	XRP	Digital currency	XRP-USD	9-Nov-2017	9-Nov-2022	1,827
Dogecoin	DOGE	Meme coin	DOGE-USD	9-Nov-2017	9-Nov-2022	1,827
Cardano	ADA	Blockchain network	ADA-USD	9-Nov-2017	9-Nov-2022	1,827
Polygon	MATIC	Layer-2 scaling solution	MATIC-USD	28-Apr-2019	9-Nov-2022	1,292
Polkadot	DOT	Blockchain network	DOT-USD	20-Aug-2020	9-Nov-2022	812
Litecoin	LTC	Digital currency	LTC-USD	9-Nov-2017	9-Nov-2022	1,827
Solana	SOL	Blockchain network	SOL-USD	10-Apr-2020	9-Nov-2022	944

4.2 Asset-based data

Technical features like the return of the cryptocurrencies and some of the asset-based features mentioned in Jaquart, Dann, and Weinhardt (2021) are also included in this research. The asset-based features used in this analysis are the daily returns for the S&P 500 and the NASDAQ, which resemble and correlate with cryptocurrencies, as further shown in the exploratory data analysis section. Moreover, the VIX returns will be included as they have an inverse correlation to Bitcoin, according to Bouri et al. (2017) and Su et al. (2022). This was further explored in the exploratory data analysis to see the relation with another cryptocurrency. The daily returns are then calculated by subtracting the daily opening prices from the closing price. The information on the asset-based dataset with their category is shown in Table 2. The trading pair column is removed as the tickers are not tradable and are only used to show the price movements. Furthermore, the asset-based data has fewer observations than the cryptocurrency dataset, even if the start- and end dates are the same. The reason for this is that traditional assets are traded on the stock market, which is not open on certain days (e.g., holidays like Independence Day or the weekends) and periods (trading hours) depending on where it is exchanged, unlike cryptocurrencies that are traded twenty-four hours seven days a week.

TABLE 2

ASSET-BASED DATA

Asset-Based	Ticker	Category	Start Date	End Date	No. of Daily Data
S&P 500	^GSPC	Stock market index	9-Nov-2017	9-Nov-2022	1,258
NASDAQ	^IXIC	Stock market index	9-Nov-2017	9-Nov-2022	1,258
VIX	^VIX	Volatility measure	9-Nov-2017	9-Nov-2022	1,258

Both cryptocurrencies and asset-based data are denominated in $USD (United States Dollar). The data type with their description is shown in Table 3 and is the same for cryptocurrency and asset-based datasets.

TABLE 3

DATA TYPE AND DESCRIPTION

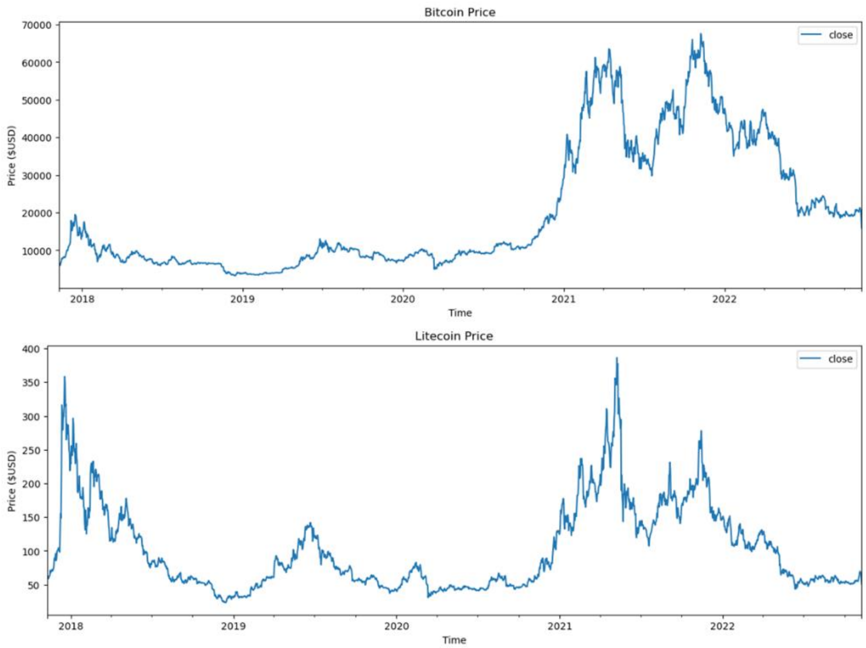
Variable	Data Type	Description
Date	object	Date of the corresponding data
Open	float64	The price of the first trade of the day
High	float64	The price of the highest trade of the day
Low	float64	The price of the lowest trade of the day
Close	float64	The price of the last trade of the day
Adj. Close	float64	The close price after adjustments
Volume	int64	The total volume of the trading day

4.3 Exploratory Data Analysis

In this section, due to the word limit imposed (explained in section 6.2) and for brevity, the returns of two cryptocurrencies (Bitcoin and Litecoin) and returns of the asset-based were outlined from the selected cryptocurrencies to perform an initial investigation. Figure 2 depicts Bitcoin and Litecoin prices over the same time span from November 9, 2017, until November 9, 2022. It is immediately apparent that Bitcoin price has grown significantly over the years. For both cryptocurrencies, a significant price surge is seen in and around the first and second quarters of 2021. Furthermore, unlike Bitcoin, it is apparent that Litecoin reached a similar price height of around $350 at the end of 2017.

FIGURE 2

BITCOIN AND LITECOIN PRICE



Moreover, Figure 3 illustrates the correlation heatmap of Bitcoin and Litecoin, and Figure 4 illustrates the pairwise scatterplot. From the correlation heatmap, we see that the returns of Bitcoin and Litecoin are positively correlated with Nasdaq and S&P500 returns, and a negative correlation exists with the VIX index. Additionally, we see that Litecoin has a lower correlation with the above indices, given the lower numbers. Furthermore, the pairwise scatterplot shows the graphed visual distribution of the Bitcoin and Litecoin data with the indices. Whereby the closer the data points approximate a straight line, the higher the connection between these factors. A good example can be seen between the strong positive association of the S&P500 with the Nasdaq, given the direction of the values.

FIGURE 3

CORRELATION HEATMAP OF BITCOIN (LEFT) AND LITECOIN (RIGHT)

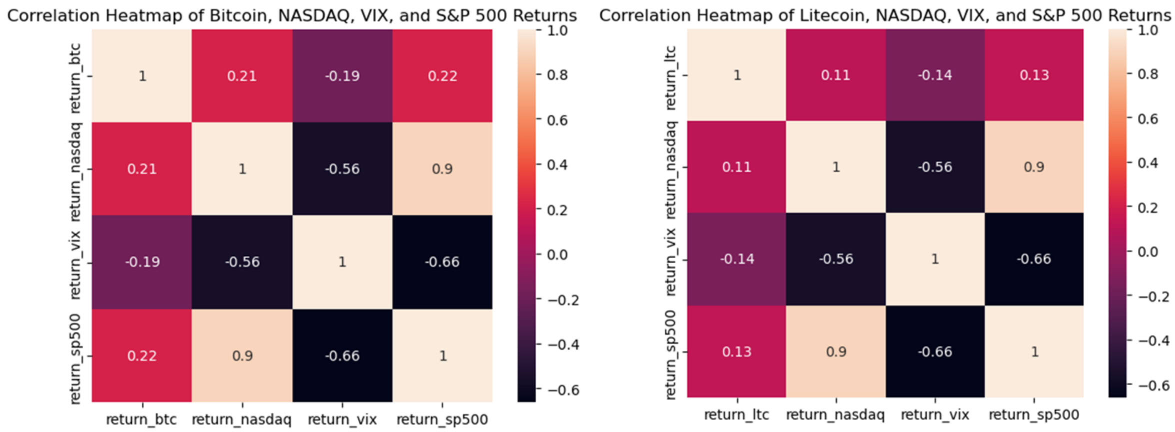
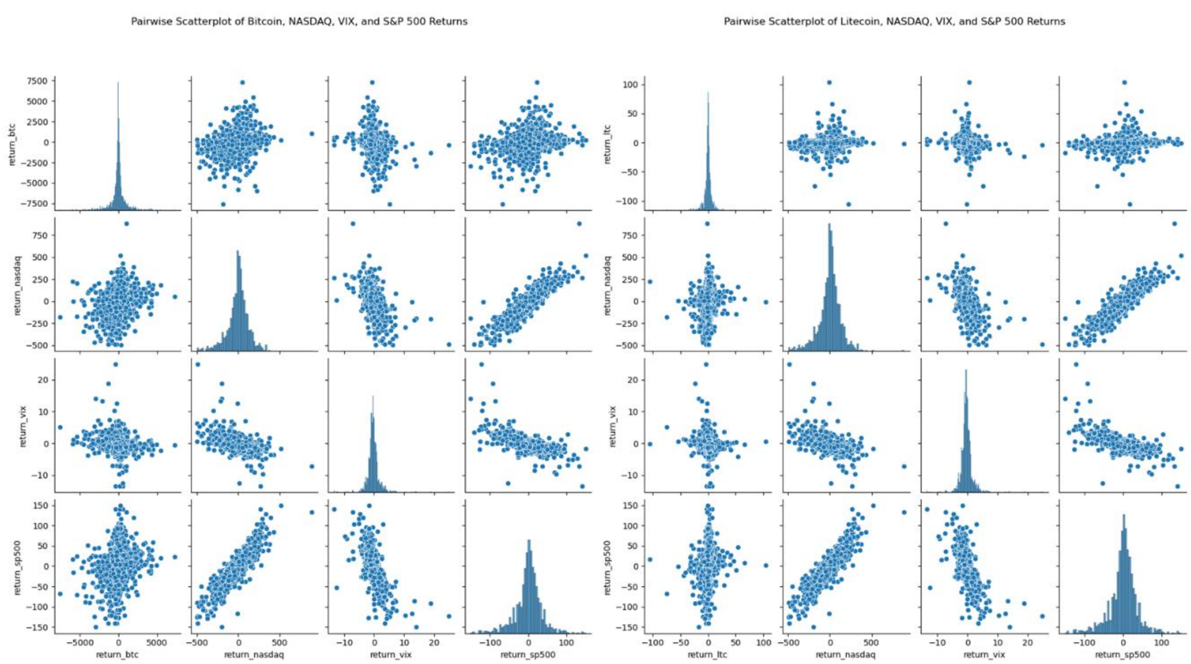


FIGURE 4

PAIRWISE SCATTERPLOT OF BITCOIN (LEFT) AND LITECOIN (RIGHT)



4.4 Preprocessing

After merging the cryptocurrency and asset-based data, the initial stage is to check for missing values. As mentioned, the asset-based data had several missing values (NAs). This can be filled with several methods, like using the mean. In this study, the gaps are filled by using a simple imputation in which the most recent available observation is taken, which is also used by Hansun et al. (2022) and Ammer and Aldhyani (2022). This approach is suitable because asset-based data are not subject to rapid fluctuations in short periods, especially for indices. Since the date variable is an object type, it is converted to DateTime format and set as the index. After, the 'Adj. Close' variable, which represents the price after paying off dividends are removed as the selected coins do not pay dividends (the values are in this case the same as the Close variable). The daily returns are calculated by subtracting the previous day's closing price from the current day's closing price and then dividing the result by the previous day's closing price. Moreover, just like in the studies by Jaquart, Dann, and Weinhardt (2021), Hansun et al. (2022), the values are rescaled (normalized) with min-max normalization, further elaborated in section 4.9.

4.5 Model Parameters

Whereas grid search examines every conceivable combination of hyperparameters to identify the optimal parameters within a model, it is computationally expensive. Therefore, in this research, the decision was made to opt for *RandomizedSearchCV* to find the parameters. With random search, a random combination of hyperparameters from the grid is chosen to find the parameters instead of trying out every combination, leading to faster computation. In addition, the number of iterations for the RF and XGBoost models is set at a default of 100, as is in line with common practice. Table 4 shows the values tested as input derived from earlier studies and a common practice for the RF and XGBoost model hyperparameters.

TABLE 4

MODEL PARAMETERS FOR RF AND XGBOOST MODELS

Model	Hyperparameter	Input values	Parameter explanation
Random Forest	n_estimators	100, 250, 500, 1000, 2000	Unit of trees in RF
	min_samples_split	2, 5, 10	Min. sample required to split node
	min_samples_leaf	1, 2, 4	Min. sample required in leaf node
	max_depth	10, 20, 50, 100	Max. level of splits in each tree
XGBoost	n_estimators	100, 250, 500, 1000, 2000	Unit of trees in XGBoost
	max_depth	3, 6, 9, 12	Max. dept of each tree
	learning_rate	0.01, 0.03, 0.05, 0.1	The shrinkage at every step

However, implementing *RandomizedsearchCV* with *TimeSeriesSplit* to identify the optimal parameters for the NN models is not practical. This is because deep learning techniques sometimes call for large datasets, which can result in models that require training for hours or days (Brownlee, 2020). To keep run times reasonable, different combinations are tested and showcased in appendix D. Table 5 shows the input values that are tested for the MLP, RNN, and LSTM models, which were derived from the literature and common practice.

TABLE 5

MODEL PARAMETERS FOR MLP, RNN AND LSTM MODELS

Model	Hyperparameter	Input values	Parameter Explanation
MLP	hidden_layer_sizes	50, 100, 200, 500	Number of nodes in each layer
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	The learning rate of MLP
RNN	epochs	30, 60, 120, 150	Number of times dataset is passed in training
	batch_size	4, 8, 16, 32, 64, 128	Training samples amount
	learning_rate	0.005, 0.001, 0.0005, 0.0001	The learning rate of RNN
LSTM	epochs	30, 60, 120, 150	Number of times dataset is passed in training
	batch_size	4, 8, 16, 32, 64, 128	Training samples amount
	learning_rate	0.005, 0.001, 0.0005, 0.0001	The learning rate of LSTM

4.6 Baseline Model

To compare the performance of different models, the RF will be used as the baseline model. As mentioned in the methodology section, the RF model is used as the baseline because it is the simplest model among the ones being compared, and previous studies like Chen (2022) and Tandon et al. (2019) also used the RF model as their baseline. This research compares the MAE and RMSE scores of comparative models (XGBoost, MLP, RNN, and LSTM) to determine whether it outperforms the baseline. The machine learning algorithm is considered to perform well if it achieves a better performance in terms of MAE and RMSE scores than the baseline model. Throughout this paper, the RF model and baseline model are used interchangeably.

4.7 Features and Target for Models without Memory Unit

Since they do not explicitly describe the temporal connections in the data, models like RF and XGBoost are examples of models without an explicit memory unit, as described in section 3. Given that the models' designs and presumptions about the nature of the data differ, it might be difficult to compare models that are built to handle time-series data with models that are not. To overcome this, we opted for the 3-fold *Timeseriessplit* and a sliding window approach with window size of 7, further elaborated in section 4.11. As standard RF and XGBoost models do not have an explicit memory function, this approach can be seen as a way of creating a memory-like effect by giving the models with a sequence of past values to learn from. Furthermore, using this approach, we don't need to create separate features for each lagged value. Instead, we rearrange the input features into a time-series format by sliding a window over the data and we would be able to use the same input features for all models. Moreover, the features that are used for the models with and without memory are the daily: 1) Open, 2) Close, 3) High, 4) Low, 5) Volume, 6) Return for each selected cryptocurrency, 7) Nasdaq return, 8) S&P 500 return, 9) VIX return. However, instead of using a three-dimensional tensor like the models with memory unit, we flatten the data for each window into a one-dimensional vector where each input and output shape for these models is a two-dimensional tensor. With the target being the 'Close' price, representing the predicted closing price of the selected cryptocurrency for the next day, and since we are using a sliding window approach with a step size of one, the models can capture some of the temporal relationships between adjacent windows. Unlike the studies by Jaquart, Dann, and Weinhardt (2021) for predicting short-term (1-min to 60-min) cryptocurrency prices, in this research, the next day close price will be forecasted, just like the study of Mudassir et al. (2020), which showcased promising results.

4.8 Features and Target for Models with Memory Unit

On the other hand, memory units are all present in the RNN and LSTM models. Although lagged copies of the input variables can be included as extra inputs to the model, previous models lack a clear method for adding historical data in the sense that they do not explicitly contain the memory function. While having an explicit memory structure that enables them to recognize temporal correlations in the input, RNNs and LSTMs are primarily created to represent sequential data as mentioned in the methodology section. Furthermore, the MLP model may not explicitly model temporal dependencies, it can still capture some level of temporal information by using aforementioned approach, and since MLP belongs to NN category, in this research the MLP model is listed in the model with memory unit. Moreover, for the models with memory unit we create a similar DataFrame with columns for open, high, low, close, volume, and returns (selected cryptocurrencies, VIX, Nasdaq, and S&P500). We then train these memory-based models utilizing a sliding window approach to incorporate temporal dependencies. This generates the input shape of three-dimensional tensor that is then used to train and evaluate the memory-based models and the output shape is a two-dimensional tensor representing the predicted closing price of the selected cryptocurrency for the next day. Moreover, in all cases for the RNN and LSTM models, the number of layers is set at three, and the parameter of units is arranged as [32, 64, 128] due to common practice. The study of Hansun et al. (2022) suggested that a straightforward three-layer structure, particularly for this type of prediction, basic architecture can obtain performance outcomes that are equivalent to those of deeper and more complicated ones. In addition, as mentioned above, for the NN models, as common practice, the data is reshaped into a 3D array before being fed into the model. For each model, the error pattern visualization is showcased in the results section and in Appendix B and C. Furthermore, 'Adam' optimization is used in all the NN models with 'ReLu' as activation, just like Chen (2022).

4.9 Data Normalization

The most prevailing approach in the literature review is the Min-Max scaler as a normalizing method. This feature scaling technique rescales the range for the features between [0,1], so that the features are measured on the same scale, and improve the machine learning performance.

EQUATION 1

MIN-MAX NORMALIZATION

$$Normalized_{(X)} = \frac{X - Min_{(X)}}{Max_{(X)} - Min_{(X)}}$$

4.10 Evaluation Metrics

The prediction ability of the models will be assessed using two common error measures for regression models in machine-learning: the Mean-Absolute-Error (MAE) and Root-Mean-Squared-Error (RMSE). In addition, most studies mentioned in the literature review used these error metrics. The MAE score is calculated by taking the average absolute difference between the forecasted- and actual value across the dataset. For the RMSE, the difference between the forecasted- and actual-value are squared. Afterward the average is taken across the dataset. For both error metrics, lower values represent a better model performance.

EQUATION 2

ERROR METRICS

Error Metrics	Formula
Mean Absolute Error (MAE)	$MAE = \frac{1}{n}\sum_{i=1}^{n} \lvert y_i - y \rvert$
Root Mean Squared Error (RMSE)	$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - y)^2}$

Where y_i is the predicted value, and y is the actual value

Since the data is split into several periods using *TimeSeriesSplit* cross-validation, this will lead to a separate MAE and RMSE score for each period. Just like the studies of Erfanian et al. (2022) and Cocco et al. (2021), the separate MAE and RMSE scores are then averaged across all periods to be used as the final evaluation metric.

4.11 Cross-validation

While a standard k-fold cross-validation technique (e.g., hold-out, leave-one-out, or stratified k-folds) is an excellent way to split the data and avoid overfitting, it creates a problem for time-series data as future observations should not be used (trained on) to make predictions of the past. This is because time-series data are not independent and evenly distributed. However, the data is dependent, and using a standard k-fold cross-validation will result in data spillover (peeking into the future). The main idea is that each test set period must come later than the previous period. There are several ways to overcome this issue, but the two commonly used methods in the literature review are the rolling window (sometimes mentioned as the sliding window) and expanding window method (sometimes mentioned as the walk forward).[3] For example, Cocco et al. (2021) and Erfanian et al. (2022) used the expanding window approach, while Oyewola et al. (2022) and Barnwal et al. (2019 used the walk forward approach. The rolling window has a fixed size, while the expanding window includes new data along the periods. This research used a 3-fold

[3] The name of the methods for splitting time-series data using cross-validation is sometimes used interchangeably in the previous research papers, even though there is a slight difference between the techniques.

TimeSeriesSplit with 70-30% train-test data split as common practice, and a sliding window approach with a window size of 7, which gives a good balance of training data and model complexity. Furthermore, in appendix A, the visualization for each train/test period using *TimeSeriesSplit* and cryptocurrencies are showcased.

4.12 Algorithms and Software

All processing and execution in this thesis are done in Python 3.7.13 (Anaconda Navigator 2.3.2 and Jupyter Notebook 6.4.12). The following packages and libraries are utilized: Pandas (1.3.5), NumPy (1.18.5), Pyplot, Seaborn (0.12.0), Tqdm (4.64.1), Scikit-learn, TensorFlow (1.15.0), Keras (2.1.6). To access both the cryptocurrency and asset-based data, YahooFinancials API is used.

5. RESULTS

This section will outline the model performance in terms of MAE and RMSE scores for each cryptocurrency and period separately. Finally, a general overview of the average MAE and RMSE results across periods of all cryptocurrencies and models will be outlined.

5.1 Bitcoin

The MAE and RMSE results per period for the models of the Bitcoin (BTC) dataset are shown in Table 6, and the accompanying error pattern visualization graphs for period 3 are shown in Figure 5. The RF model resulted in average MAE and RMSE scores of 3,619.93 and 6,547.88, respectively. The XGBoost model performed slightly better, with an average MAE of 3,556.44 and an RMSE score of 6,506.36 across periods. The best-performing model is the RNN model. with an average MAE and RMSE score of 1,846.83 and 2,515, with the LSTM model as runner-up having a slightly higher MAE and RMSE score of 2,003.99 and 2,710.97 respectively. While the MLP model had an inferior performance on average across all periods compared to the other two NN models, it still outperformed RF and XGBoost models by a considerable margin with an average MAE and RMSE of 2,127.92 and 3,535.66, respectively. Moreover, the MLP model had a far superior performance in predicting period 3, as showcased in the error pattern visualization. The smaller distance between the actual and predicted price lines indicates a better performance. Furthermore, as shown in Figure 5, RNN and LSTM models were outperformed by the RF and XGBoost models in predicting the price of BTC for period 3.

TABLE 6

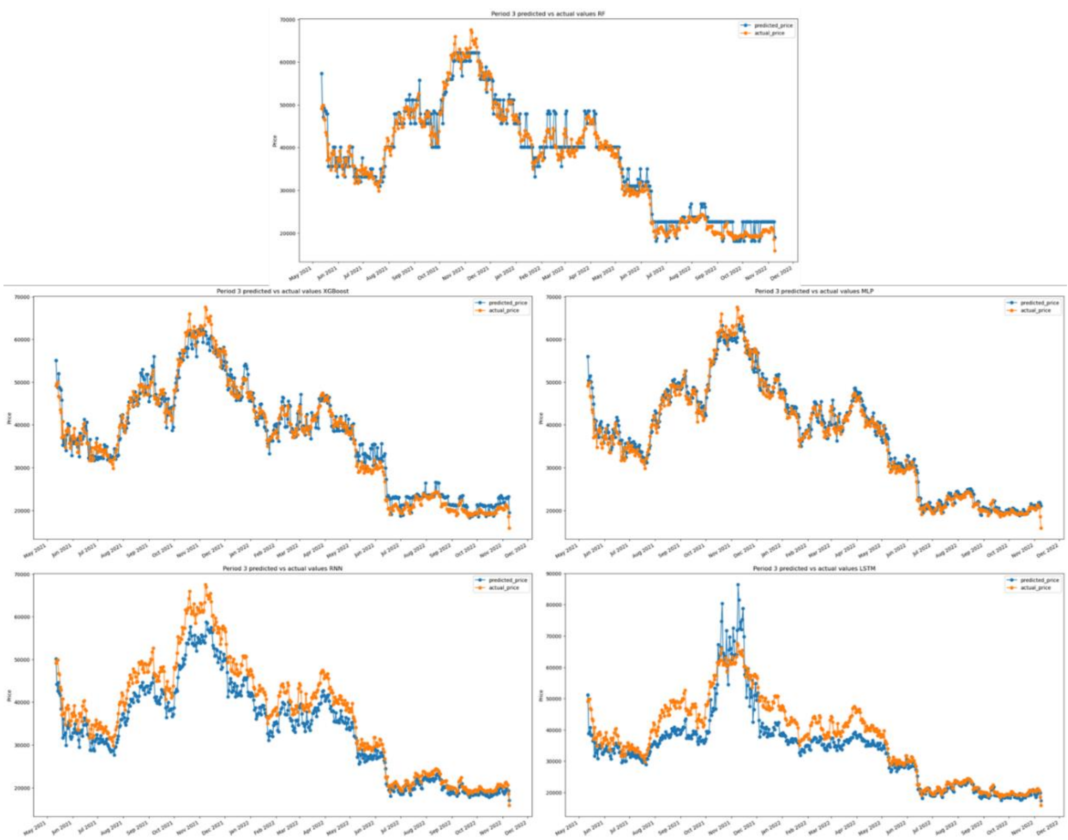
BITCOIN MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	1,021.3402	1,058.4958	1,322.0843	718.2236	1,277.1019
Period 2	7,822.6211	7,689.6479	3,566.7063	1,177.1306	637.2353
Period 3	2,015.8153	1,921.1776	1,494.9556	3,645.1483	4,097.6457

Average	3,619.9256	3,556.4405	2,127.9155	1,846.8342	2,003.9944
RMSE					
Period 1	1,399.1677	1,496.5066	1,601.2831	973.4368	1,691.6154
Period 2	15,714.3719	15,577.5591	7,018.4809	2,226.4023	1,112.9666
Period 3	2,530.0973	2,445.0232	1,987.2083	4,345.1585	5,328.3298
Average	6,547.8790	6,506.3630	3,535.6575	2,514.9992	2,710.9707

Figure 5

Bitcoin Error Pattern Visualization for Period 3 (From top to bottom and left to right RF, XGBOOST, MLP, RNN and LSTM)



5.2 Ethereum

The MAE and RMSE results per period for the models of the Ethereum (ETH) dataset are shown in Table 7, and the accompanying error pattern visualization graphs for period 3 are shown in Figure 6. The RF model resulted in average MAE and RMSE scores of 188.93 and 317.16, respectively. The XGBoost model performed slightly better, with an average MAE and RMSE score across periods of 187.88 and 316.12, respectively. The MLP model had a lower average MAE of 115.27 and an average RMSE score across periods of 160.72. The best-performing model is the LSTM model, given the lowest average MAE score of

94.04 and RMSE score across periods of 126.73. The RNN model had the second-lowest average MAE and RMSE scores across periods of 99.66 and 136.25, respectively. The MLP and LSTM models had the best performance in predicting the ETH price for period 3, as seen in the error pattern visualization, where the actual and predicted lines follow each other closely.

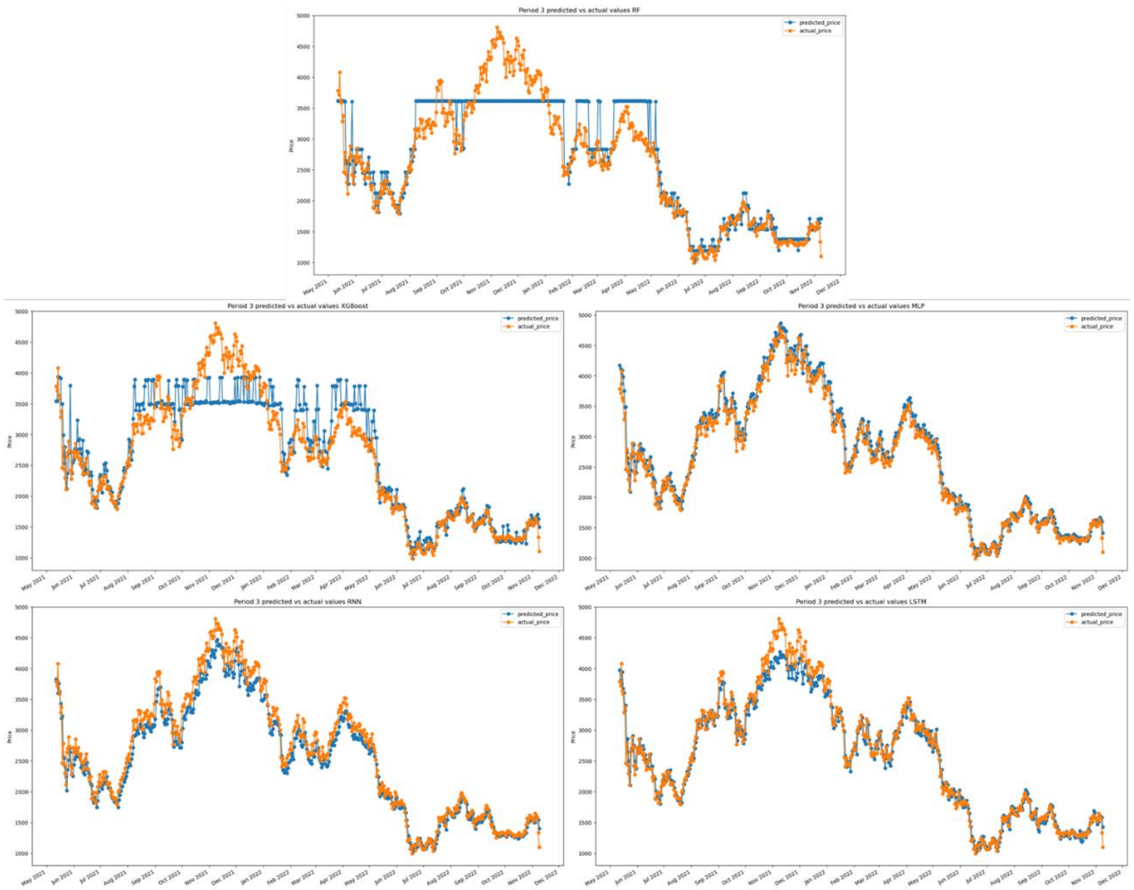
TABLE 7

ETHEREUM MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	125.7137	137.8595	161.2175	106.6618	125.5759
Period 2	166.4938	151.8153	61.6038	43.5030	34.4451
Period 3	274.5883	273.9661	122.9863	148.8092	122.0854
Average	188.9320	187.8804	115.2692	99.6581	94.0355
RMSE					
Period 1	142.7886	154.8428	171.7173	120.3267	140.1135
Period 2	426.0585	405.1560	141.9610	93.0554	67.8574
Period 3	382.6364	388.3472	168.4945	195.3555	172.2089
Average	317.1612	316.1154	160.7243	136.2459	126.7266

FIGURE 6

ETHEREUM ERROR PATTERN VISUALIZATION FOR PERIOD 3 (RF, XGBOOST, MLP, RNN AND LSTM)



5.3 Binance Coin

The MAE and RMSE results per period for the models of the Binance Coin (BNB) dataset are shown in Table 8, and the accompanying error pattern visualization graphs for period 3 are shown in Figure 7. The RF model resulted in average MAE and RMSE scores of 31.26 and 66.06, respectively. The XGBoost performed better with an average MAE score across periods of 30.97 and an average RMSE score across periods of 65.47. While the MLP model had the lowest MAE score (20.11), the average RMSE score (41.16) across periods was higher than the other NN models. However, the MLP model had a far superior performance in predicting period 3, also showcased in the error pattern visualization. The RNN model had the lowest RMSE score of 33.88 and an average MAE of 24.95. The LSTM model had the worst performance among the NN models with an average MAE and RMSE score across periods of 26.50 and 41.17, respectively, but still had a far better performance than the RF and XGBoost models in predicting the BNB price.

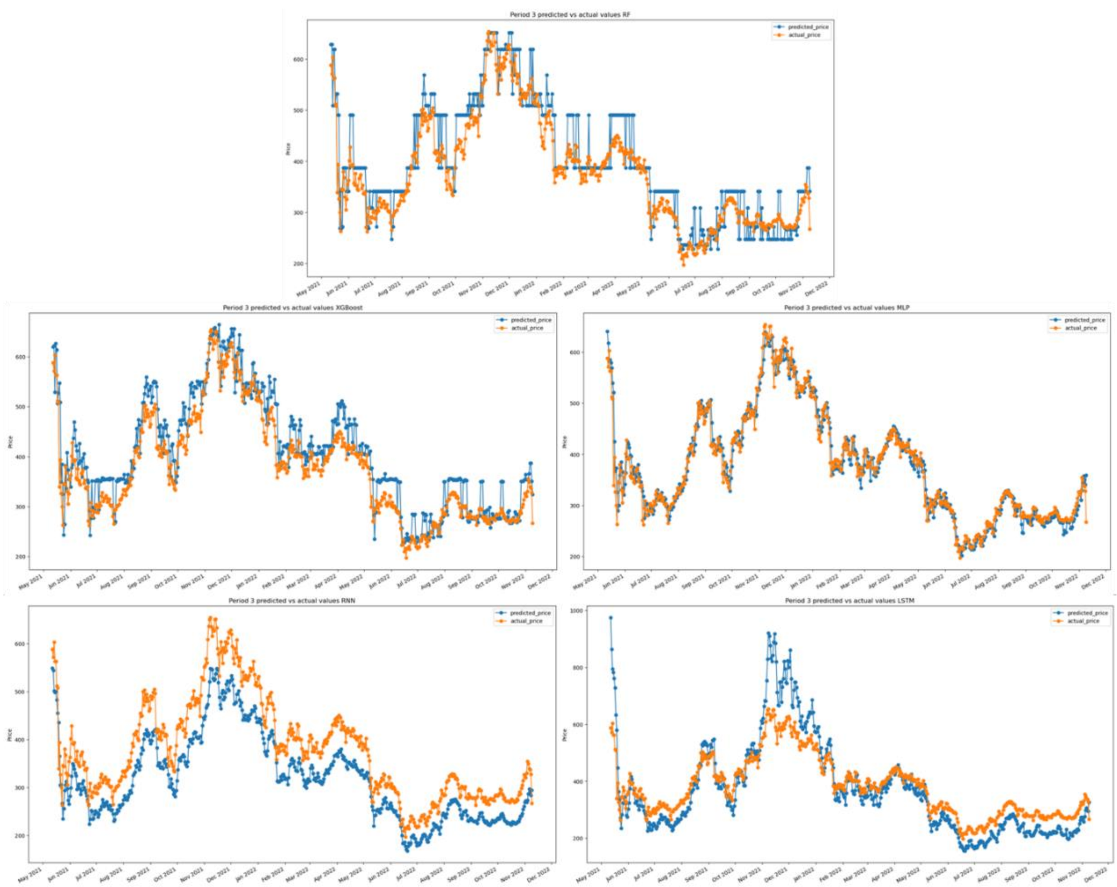
TABLE 8

BINANCE COIN MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	3.2730	2.5866	7.9500	3.2789	8.2884
Period 2	55.6892	55.5107	36.6681	11.5018	14.9256
Period 3	34.8293	34.8206	15.7346	60.0765	56.2947
Average	31.2639	30.9727	20.1176	24.9525	26.5030
RMSE					
Period 1	5.4440	4.4613	10.7069	5.5800	11.2422
Period 2	148.8639	148.7983	90.5595	31.7298	37.4617
Period 3	43.8851	43.1536	22.2409	64.3551	74.8280
Average	66.0644	65.4711	41.1692	33.8883	41.1773

FIGURE 7

BINANCE COIN ERROR PATTERN VISUALIZATION FOR PERIOD 3 (RF, XGBOOST, MLP, RNN AND LSTM)



5.4 Ripple

The MAE and RMSE results per period for the models of the Ripple (XRP) dataset are shown in Table 9, and the accompanying error pattern visualization graphs for period 3 are shown in Figure 8. The RF model resulted in average MAE and RMSE scores of 0.0485 and 0.0767, respectively. The XGBoost model performed better, with an average score of 0.045 and 0.0722, respectively. The best-performing model is the LSTM model, with an average MAE and RMSE score of 0.0302 and 0.0517, respectively. While the RNN model had a lower average MAE score of 0.0337 compared to the MLP model (0.0362), it had a slightly higher average RMSE score across periods. Moreover, it can be seen in Figure 8 that all the NN models had a similar performance in predicting the price of XRP for period 3, where the predicted price line closely resembles the actual price for each model.

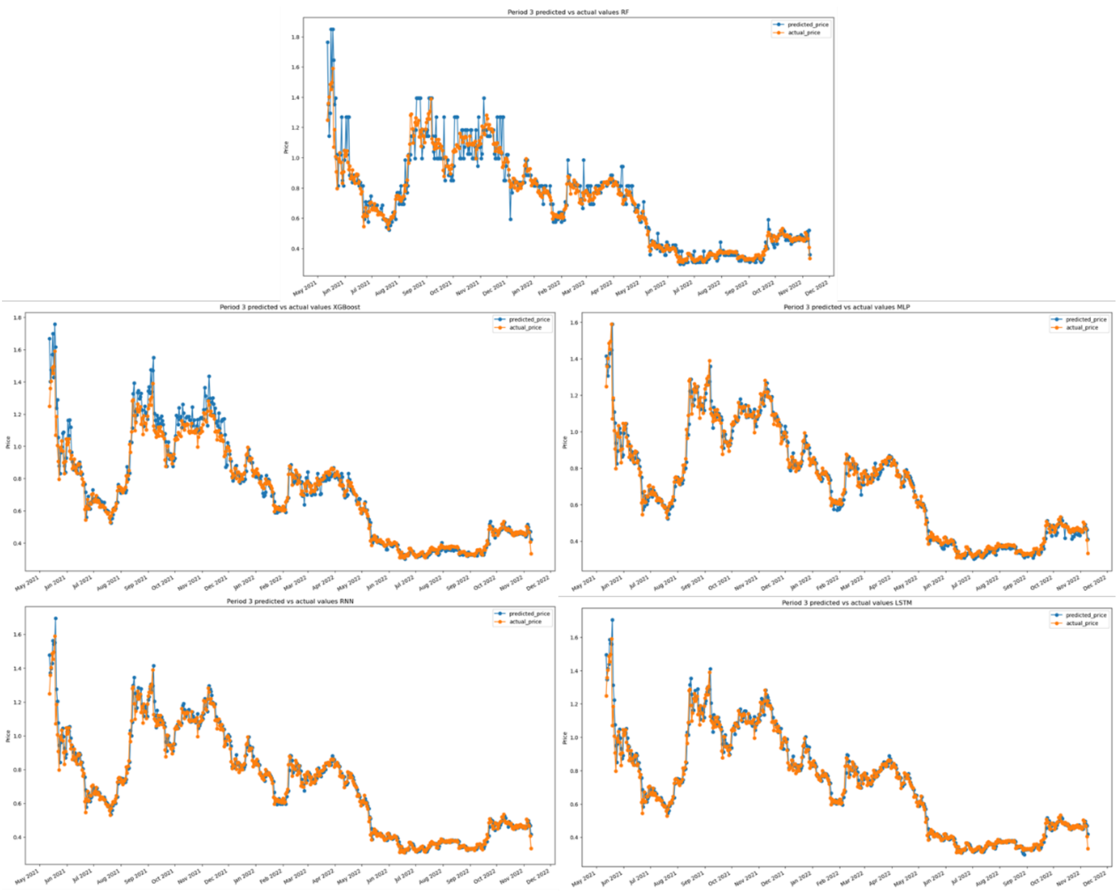
TABLE 9

RIPPLE MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	0.0602	0.0623	0.0267	0.0276	0.0225
Period 2	0.0315	0.0314	0.0501	0.0442	0.0382
Period 3	0.0537	0.0412	0.0316	0.0290	0.0297
Average	0.0485	0.0450	0.0362	0.0337	0.0302
RMSE					
Period 1	0.0766	0.0907	0.0360	0.0397	0.0316
Period 2	0.0637	0.0574	0.0892	0.0913	0.0702
Period 3	0.0897	0.0684	0.0504	0.0522	0.0532
Average	0.0767	0.0722	0.0586	0.0611	0.0517

FIGURE 8

RIPPLE ERROR PATTERN VISUALIZATION FOR PERIOD 3 (RF, XGBOOST, MLP, RNN AND LSTM)



5.5 Dogecoin

The MAE and RMSE results per period for the models of the Dogecoin (DOGE) dataset are shown in Table 10, and the accompanying error pattern visualization graphs for period 3 are shown in Figure 9. The RF model resulted in average MAE and RMSE scores of 0.0243 and 0.0538, respectively. The RF model outperformed the XGBoost model with average MAE and RMSE scores across periods of 0.0353 and 0.0654, respectively. The best-performing model is the LSTM model, with an average MAE score across periods of 0.0141 and an RMSE score across periods of 0.0335. The MLP model had the second-lowest average MAE and RMSE scores across periods of 0.0142 and 0.0347, respectively. Despite falling short compared to the LSTM models in terms of the average score across periods, the MLP model had a far better performance predicting period 3, as showcased in the error pattern visualization graph.

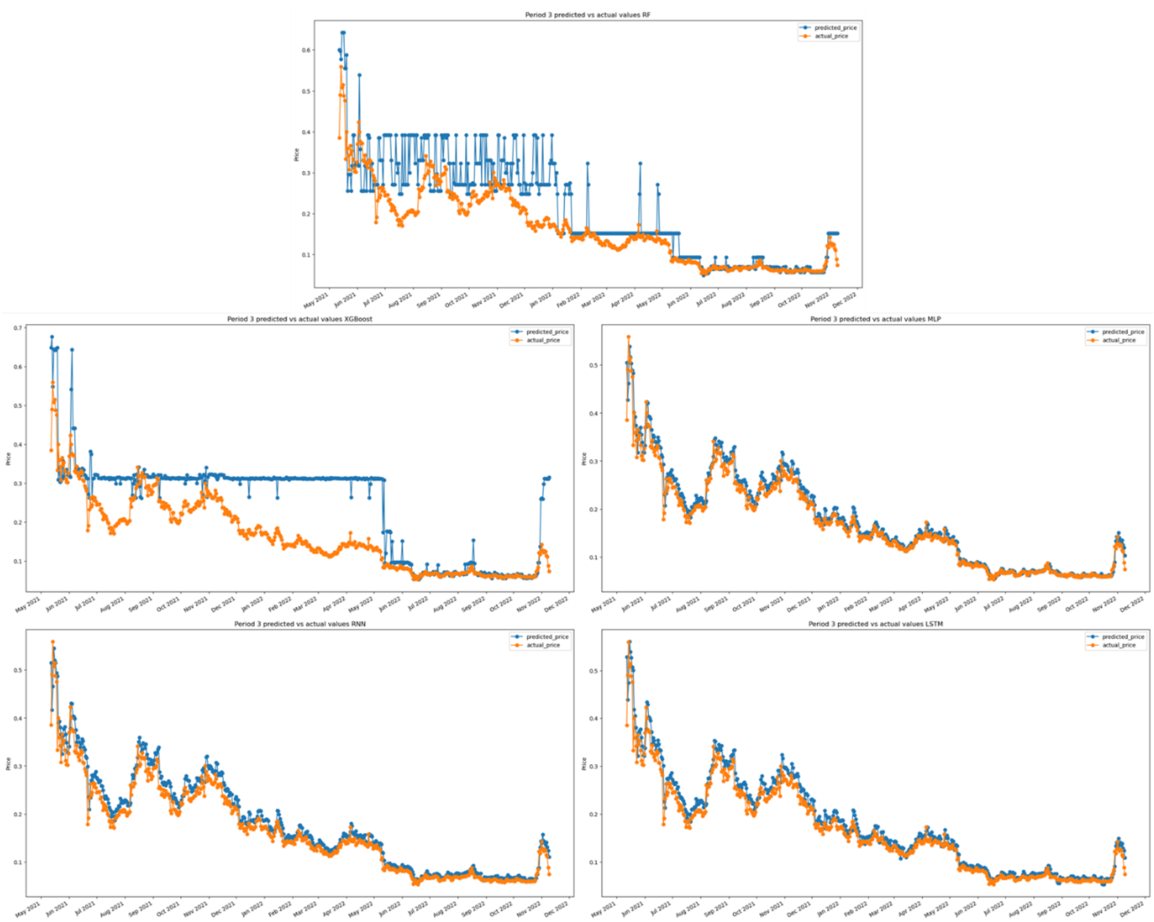
TABLE 10

DOGECOIN MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	0.0002	0.0003	0.0067	0.0051	0.0025
Period 2	0.0240	0.0238	0.0236	0.0258	0.0240
Period 3	0.0485	0.0817	0.0122	0.0167	0.0155
Average	0.0243	0.0353	0.0142	0.0159	0.0141
RMSE					
Period 1	0.0003	0.0003	0.0071	0.0055	0.0031
Period 2	0.0883	0.0881	0.0776	0.0866	0.0751
Period 3	0.0728	0.1078	0.0192	0.0230	0.0220
Average	0.0538	0.0654	0.0347	0.0384	0.0335

FIGURE 9

DOGECOIN ERROR PATTERN VISUALIZATION FOR PERIOD 3 (RF, XGBOOST, MLP, RNN AND LSTM)



5.6 Cardano

The MAE and RMSE results per period for the models of the Cardano (ADA) dataset are shown in Table 11. The accompanying error pattern visualization graphs for period 3 are shown in Figure 10. The RF model resulted in average MAE and RMSE scores of 0.0882 and 0.1591, respectively. The XGBoost model performed better with average MAE and RMSE scores across periods of 0.0752 and 0.1394, respectively. The LSTM model had the lowest average MAE score across periods (0.0303) compared to the MLP (0.0457) and RNN model (0.0310); however, the LSTM RMSE score (0.0501) was slightly higher than the RNN model (0.0493). Furthermore, as seen in the error pattern visualization, the RNN and LSTM model best predicted the ADA price for period 3.

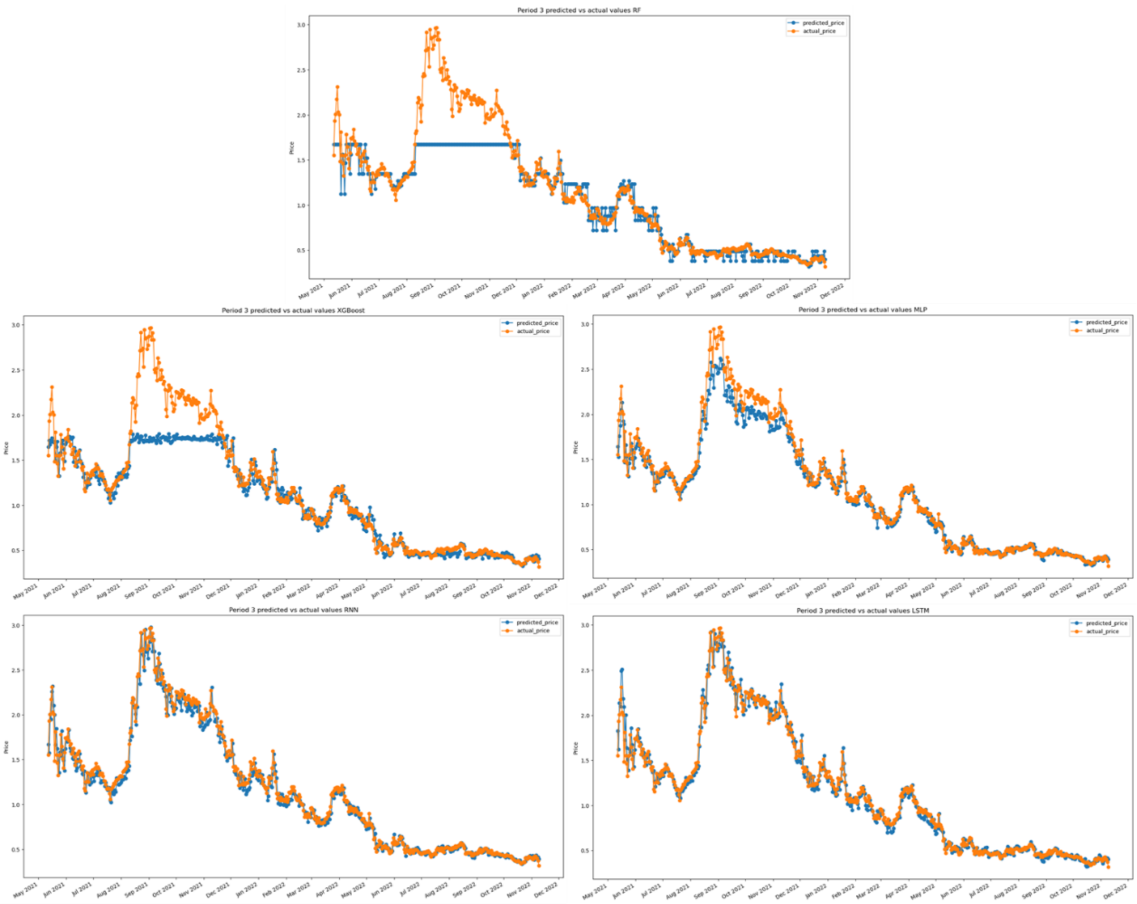
TABLE 11

CARDANO MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	0.0429	0.0351	0.0184	0.0114	0.0042
Period 2	0.0506	0.0412	0.0428	0.0204	0.0276
Period 3	0.1709	0.1493	0.0758	0.0609	0.0591
Average	0.0882	0.0752	0.0457	0.0310	0.0303
RMSE					
Period 1	0.0477	0.0397	0.0200	0.0179	0.0059
Period 2	0.1268	0.1043	0.0769	0.0393	0.0557
Period 3	0.3027	0.2742	0.1220	0.0907	0.0886
Average	0.1591	0.1394	0.0730	0.0493	0.0501

FIGURE 10

CARDANO ERROR PATTERN VISUALIZATION FOR PERIOD 3 (RF, XGBOOST, MLP, RNN AND LSTM)



5.7 Polygon

The MAE and RMSE results per period for the models of the Polygon (MATIC) dataset are shown in Table 12, and the accompanying error pattern visualization graphs for period 3 are shown in Figure 11. The RF model resulted in average MAE and RMSE scores of 0.28 and 0.40, respectively. The XGBoost model performed slightly better, with an average MAE across periods of 0.27 and an average RMSE score across periods of 0.39. The best-performing model is the RNN model, with an average MAE and RMSE score of 0.24 and 0.33, respectively. The LSTM model had marginally lower MAE (0.25) and RMSE (0.35) scores compared to the MLP model (0.26 and 0.36, respectively). For period 3 however, the price predictions for the MLP model were the closest to the actual value, also seen in the error pattern visualization in Figure 11.

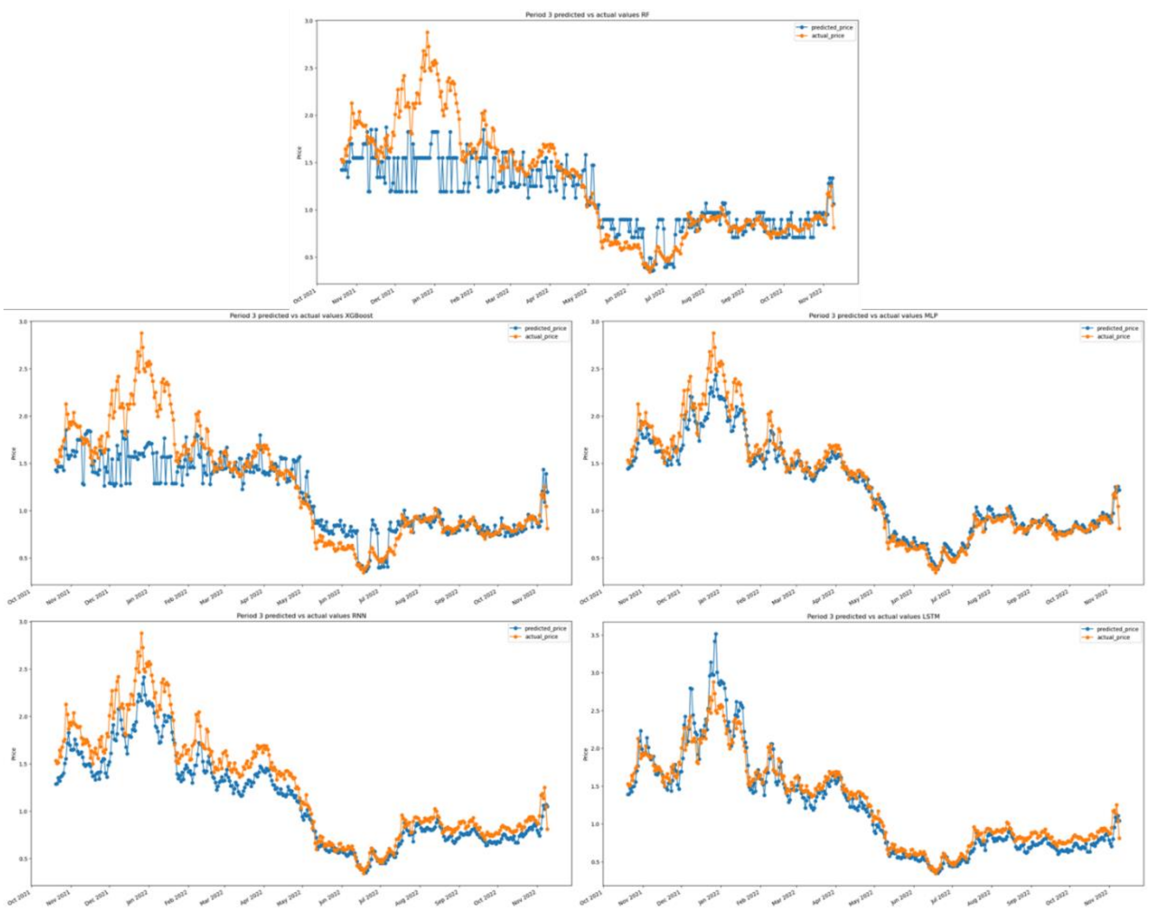
TABLE 12

POLYGON MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	0.0020	0.0015	0.1323	0.0054	0.0042
Period 2	0.6146	0.6143	0.5763	0.5550	0.6259
Period 3	0.2396	0.2062	0.0938	0.1631	0.1251
Average	0.2854	0.2741	0.2675	0.2412	0.2518
RMSE					
Period 1	0.0030	0.0025	0.1347	0.0070	0.0053
Period 2	0.8679	0.8678	0.8123	0.7800	0.8776
Period 3	0.3515	0.3223	0.1374	0.2060	0.1686
Average	0.4075	0.3976	0.3615	0.3310	0.3506

FIGURE 11

POLYGON ERROR PATTERN VISUALIZATION FOR PERIOD 3 (RF, XGBOOST, MLP, RNN AND LSTM)



5.8 Polkadot

The MAE and RMSE results per period for the models of the Polkadot (DOT) dataset are shown in Table 13, and the accompanying error pattern visualization graphs for period 3 are shown in Figure 12. The RF model resulted in average MAE and RMSE scores of 7.01 and 8.77, respectively. Like the previous results, the XGBoost model had lower average MAE and RMSE scores across periods (6.64 and 8.43, respectively). The best-performing model is the LSTM model, given the lowest average MAE score of 4.14 and RMSE score across periods of 5.28. The RNN model had the second-lowest average MAE and RMSE scores across periods of 4.58 and 5.83, respectively. Despite falling short compared to the other NN models in terms of the average score across periods, the MLP model had a far better performance predicting period 3, as showcased in the error pattern visualization.

TABLE 13

POLKADOT MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	16.8703	16.8500	14.2757	11.5782	10.4964
Period 2	2.8235	2.2489	1.5266	1.4475	1.3011
Period 3	1.3546	0.8306	0.5373	0.7304	0.6427
Average	7.0162	6.6432	5.4466	4.5854	4.1467
RMSE					
Period 1	20.9662	20.9532	17.7699	14.6346	13.2619
Period 2	3.6298	3.2914	2.0704	1.9326	1.8075
Period 3	1.7240	1.0568	0.7109	0.9227	0.7898
Average	8.7734	8.4339	6.8505	5.8300	5.2864

FIGURE 12

POLKADOT ERROR PATTERN VISUALIZATION FOR PERIOD 3 (RF, XGBOOST, MLP, RNN AND LSTM)

5.9 Litecoin

The MAE and RMSE results per period for the models of the Litecoin (LTC) dataset are shown in Table 14, and the accompanying error pattern visualization graphs for period 3 are shown in Figure 13. The RF model resulted in average MAE and RMSE scores of 9.77 and 16.09, respectively. The XGBoost model surprisingly had a higher average MAE score across periods of 11.23. However, the average RMSE score of the XGBoost model was 15.67, thus lower than the RF model. The MLP model had a lower average MAE score across periods of 8.64 than the RNN model at 8.67. However, the RMSE score was worse compared to the RNN model. The LSTM model had the best performance, with average MAE and RMSE scores across periods of 7.96 and 11.52, respectively. The LSTM and RNN model performed the best in predicting the price for period 3 as seen in Figure 13, with the latter model having a slightly better prediction.

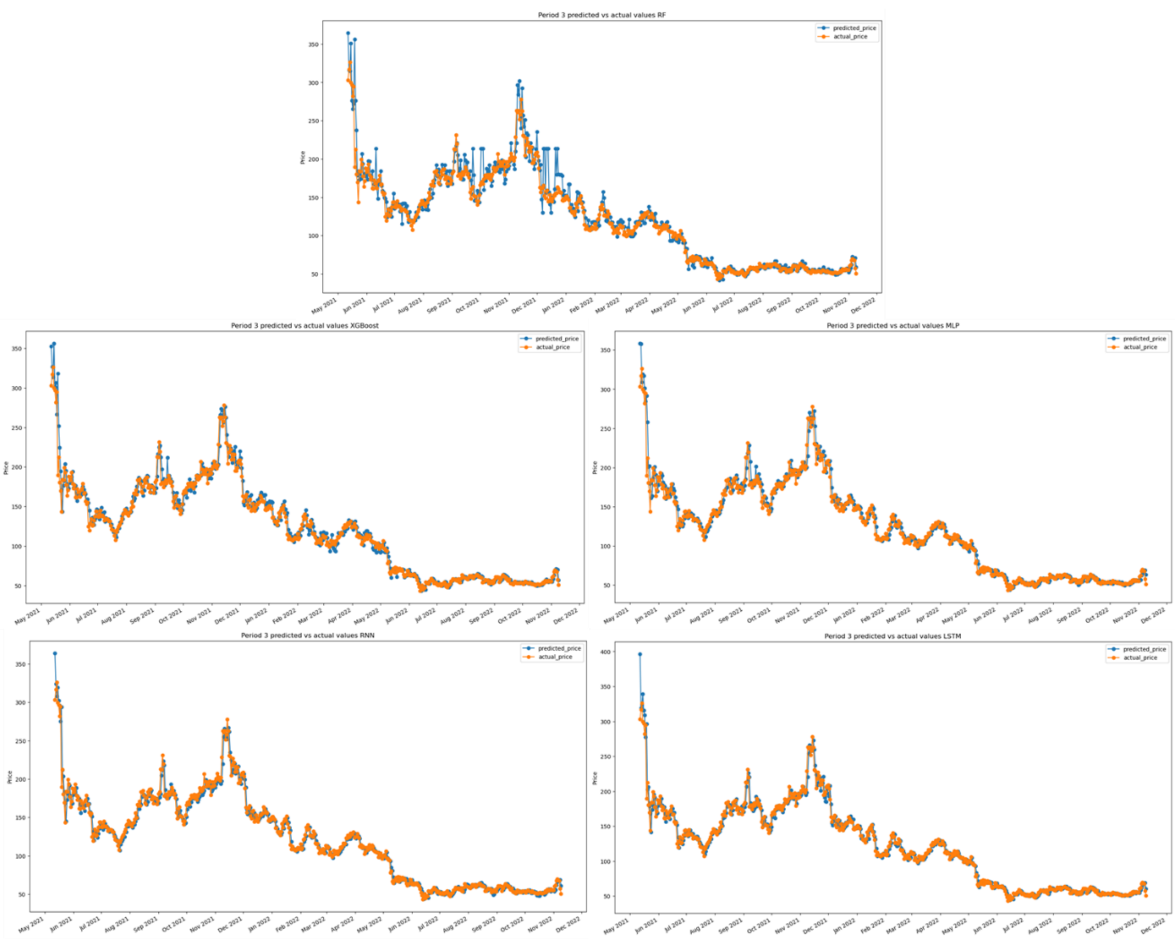
TABLE 14

LITECOIN MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	12.8455	21.6784	13.6878	15.9603	13.8639
Period 2	7.9645	6.1564	5.5826	4.9078	4.8090
Period 3	8.5103	5.8576	6.6621	5.1453	5.2291
Average	9.7735	11.2309	8.6442	8.6712	7.9674
RMSE					
Period 1	18.0986	25.7149	17.0314	18.8385	15.5734
Period 2	14.7308	10.8480	8.8358	8.6542	9.2163
Period 3	15.4500	10.4591	10.7666	9.0697	9.7863
Average	16.0932	15.6741	12.2113	12.1875	11.5254

FIGURE 13

LITECOIN ERROR PATTERN VISUALIZATION FOR PERIOD 3 (RF, XGBOOST, MLP, RNN AND LSTM)



5.10 Solana

The MAE and RMSE results per period for the models of the Solana (SOL) dataset are shown in Table 15, and the accompanying error pattern visualization graphs for period 3 are shown in Figure 14. The RF model resulted in average MAE and RMSE scores of 29.88 and 40.57, respectively. The XGBoost model performed better, with an average MAE score across periods of 28.56 and an average RMSE across periods of 39.33. The best-performing model was the LSTM model, with average MAE and RMSE scores across periods of 10.37 and 13.98, respectively. The RNN model predicted the price of SOL better on average than the MLP model. It also had the best performance in predicting the price for period 3, also seen on the error pattern visualization where the actual and predicted lines follow each other closely.

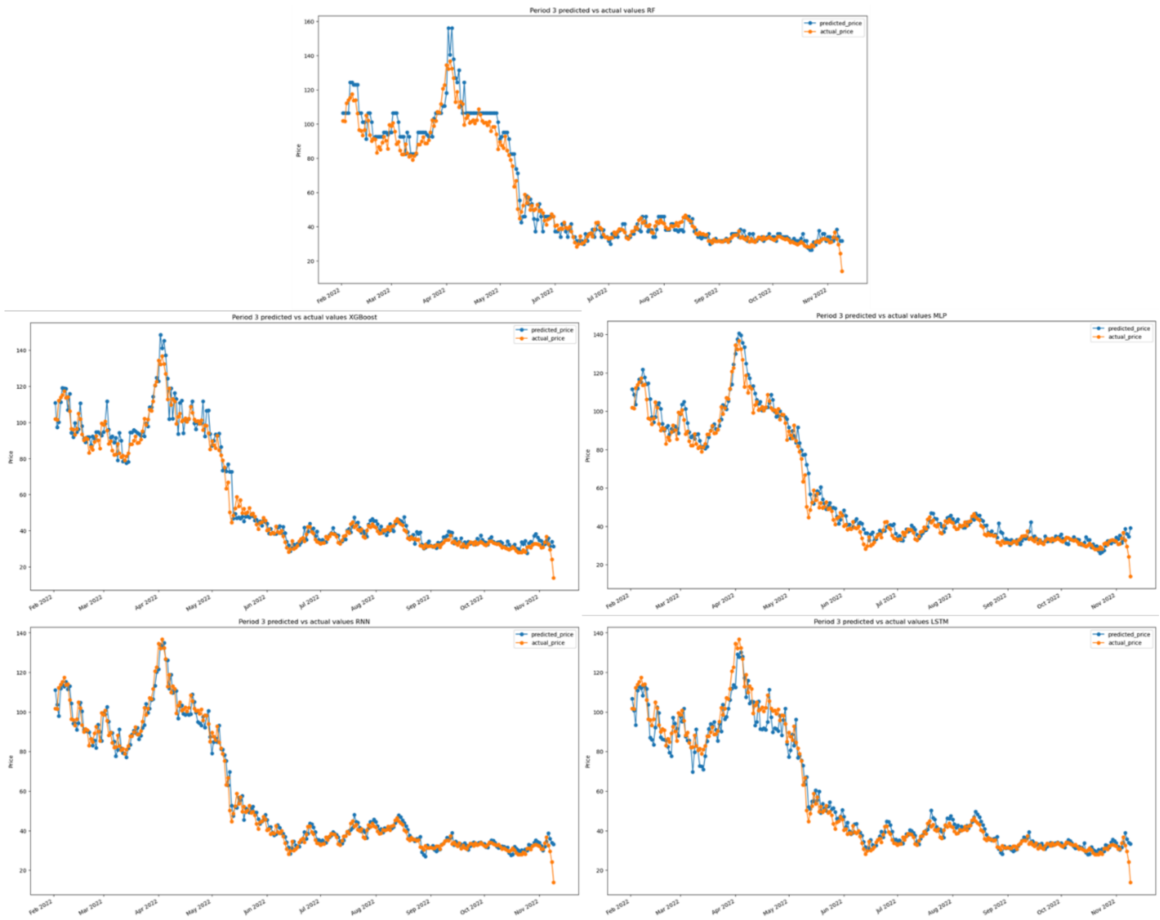
TABLE 15

SOLANA MAE AND RMSE RESULTS

Periods	RF	XGBoost	MLP	RNN	LSTM
MAE					
Period 1	5.6913	5.6673	10.0307	6.0891	6.1090
Period 2	79.5157	76.1985	28.7436	32.7164	21.1795
Period 3	4.4425	3.8164	3.6235	2.9963	3.8288
Average	29.8832	28.5608	14.1326	13.9340	10.3725
RMSE					
Period 1	9.8387	9.8240	13.5543	10.0764	10.0897
Period 2	105.5675	102.7142	39.5981	40.5201	26.5273
Period 3	6.3331	5.4535	5.2302	4.2882	5.3296
Average	40.5798	39.3306	19.4609	18.2949	13.9822

FIGURE 14

SOLANA ERROR PATTERN VISUALIZATION FOR PERIOD 3 (RF, XGBOOST, MLP, RNN AND LSTM)



5.11 Results overview

Table 16 shows the average MAE and RMSE scores across periods of different models and coins. The numbers highlighted in green are the ones that performed the best compared to other models, and the numbers highlighted in red are the ones that the baseline model has outperformed. In Table 16, it is demonstrated that the LSTM model had seven times the lowest MAE score and six times the lowest RMSE score across the analyzed cryptocurrencies. The RNN model had two times the lowest MAE and four times the lowest RMSE score across different cryptocurrencies, while the MLP model only outperformed every other model once for the BNB-USD pair in terms of the lowest MAE score. Moreover, the XGBoost model, on average, had a better performance than the baseline model in predicting close prices. However, it was outperformed three times for the DOGE-USD and LTC-USD pairs, which could be due to the chosen hyperparameters.

Table 16

Results Overview: Average MAE and RMSE Score Across Periods

Cryptocurrencies	RF	XGBoost	MLP	RNN	LSTM
MAE					
BTC-USD	3,619.9256	3,556.4405	2,127.9155	1,846.8342	2,003.9944
ETH-USD	188.9320	187.8804	115.2692	99.6581	94.0355
BNB-USD	31.2639	30.9727	20.1176	24.9525	26.5030
XRP-USD	0.0485	0.0450	0.0362	0.0337	0.0302
DOGE-USD	0.0243	0.0353	0.0142	0.0159	0.0141
ADA-USD	0.0882	0.0752	0.0457	0.0310	0.0303
MATIC-USD	0.2854	0.2741	0.2675	0.2412	0.2518
DOT-USD	7.0162	6.6432	5.4466	4.5854	4.1467
LTC-USD	9.7735	11.2309	8.6442	8.6712	7.9674
SOL-USD	29.8832	28.5608	14.1326	13.9340	10.3725
RMSE					
BTC-USD	6,547.879	6,506.363	3,535.6575	2,514.9992	2,710.9707
ETH-USD	317.1612	316.1154	160.7243	136.2459	126.7266
BNB-USD	66.0644	65.4711	41.1692	33.8883	41.1773
XRP-USD	0.0767	0.0722	0.0586	0.0611	0.0517
DOGE-USD	0.0538	0.0654	0.0347	0.0384	0.0335
ADA-USD	0.1591	0.1394	0.0730	0.0493	0.0501
MATIC-USD	0.4075	0.3976	0.3615	0.3310	0.3506
DOT-USD	8.7734	8.4339	6.8505	5.8300	5.2864
LTC-USD	16.0932	15.6741	12.2113	12.1875	11.5254
SOL-USD	40.5798	39.3306	19.4609	18.2949	13.9822

6. DISCUSSION

The problem statement that this research addressed is the lack of comparisons in the existing literature regarding machine learning models in predicting cryptocurrency prices. Therefore, in this thesis, a comparison was made of five machine learning models and their predictive power regarding 10 different cryptocurrencies. In particular, this thesis tried to answer the following research question:

"How well can the price of selected cryptocurrencies be predicted with machine learning methods?"

Two related sub-questions were created to address the main research question:

Sub-question 1: "*Which of a set of selected machine learning algorithms performs well in the prediction of cryptocurrency prices?"*

This sub-question will be answered by comparing the five machine learning algorithms across 10 cryptocurrencies to assess which model performed well in predicting the price of cryptocurrencies. This was done by assessing these models' average MAE and RMSE scores across 3 periods and against the baseline model. As was further outlined in the experimental setup section, different hyperparameters were used for each model and cryptocurrency. Moreover, both technical and asset-based features were used to predict the price. In this research, the machine learning algorithm is considered to perform well if it achieves a better MAE and RMSE score across periods than the baseline model, which was different for each cryptocurrency. The results show that, on average, all machine learning algorithms (XGBoost, MLP, RNN, and LSTM) compared to the baseline RF model performed well. The results are in line with previous studies like Mohta et al. (2022) and Tandon et al. (2019), where the comparative models outperformed the baseline model, which substantiates their hypothesis.

Sub-question 2: "*How does the predictive power of different machine learning algorithms compare across cryptocurrencies?"*

The second sub-question is answered by comparing the aforementioned models, and their predictive power is then compared with the selected cryptocurrencies mentioned in Table 1. We opted for the 10 largest cryptocurrencies in terms of market capitalization (as of November 9, 2022) and collected the data as a CSV file from Yahoo Finance (API). Stablecoins were excluded from this research, which is not informative to analyze as their values are pegged to the US dollar. Moreover, the start- and end dates and the number of daily observations differed for each cryptocurrency since some projects were founded and launched at a later stage. Since different cryptocurrencies were used, there is no true apples-to-apples comparison for the MAE and RMSE scores. Therefore, an overview of the MAE and RMSE scores across periods of different models and coins was highlighted in the results overview to see whether a model performed the best for one particular cryptocurrency and whether the predictive power was lacking (compared to other models) for the other cryptocurrency. This will give insight into the predictive power of different machine-learning algorithms across cryptocurrencies. The conclusion that follows is that there is not just one machine-learning algorithm that is certain to be the most accurate for all cryptocurrencies. Moreover, on average, the NN models outperformed the machine-learning models. However, it's crucial to remember that a specific NN model might not always be the ideal option for every cryptocurrency price prediction, as showcased in the result section. This is not surprising, as factors like the unique properties of the coin under consideration, prediction horizon, and specific features and inputs influence the effectiveness of the price prediction for a particular model. This was also the case for prior studies like Hansun et al. (2022) and Fleischer et al. (2022), among others. Although they used different techniques, the predictive nature of a machine learning algorithm differed for a particular cryptocurrency.

With the answers to the sub-questions, the main research question will be answered:

"*How well can the price of selected cryptocurrencies be predicted with machine learning methods?"*

The findings demonstrate that the trained models perform substantially better than the baseline RF model. From this research, the LSTM model performed the best while the RNN model placed second, given the number of times these algorithms had the lowest MAE and RMSE score across different cryptocurrencies. The MLP model placed third, followed by the XGBoost model, but the latter failed to outperform the baseline model on at least one occasion.

6.1 Scientific and Societal Impact

This study provides a knowledge base to comprehend various machine learning algorithms and their predictive power across cryptocurrencies. While the aforementioned studies only used a few techniques, models, and cryptocurrencies. Particularly, it helps to understand what machine learning algorithm performs best for a specific cryptocurrency. From the scientific point of view, this research can assist academics in gaining insight into the predictive nature of different machine learning algorithms to test economic theories and models. In addition, these models may be used to create predictive models that can anticipate future prices with varied degrees of accuracy and potentially result in more accurate financial market forecasting techniques. Moreover, this research is interesting from a societal standpoint because cryptocurrencies have become a well-liked asset class in the broader society. Some people spend substantial sums of money in the expectation of making money off price changes. Correct cryptocurrency price forecasts can aid traders and investors in making more educated choices about whether to purchase, sell, or hold certain assets, potentially improving the performance of their investments.

6.2 Limitations and Future Research

Due to processing power, a more thorough parameter-tuning process could not be followed for this research, especially when the number of iterations and epochs was increased. Moreover, this research makes no inferences on modifying the ideal machine-learning algorithm to obtain the best performance. Future studies may also include additional hyperparameters. Regarding the NN models, combining different layers instead of a fixed number of layers could be experimented upon to see whether performance increases since there was no experimentation regarding the structure in this research due to the imposed word count limit and due to a vast number of cryptocurrencies that were analyzed. In addition, the prediction with different time horizons might lead to better results, which was left out in this research due to the word limit. Finally, as the market environment and the underlying cryptocurrency data are subject to change, further research could be used to find the optimal performance.

7. CONCLUSION

Although institutional investors are becoming more interested in the emerging asset class of cryptocurrencies, it is still challenging to predict their future pricing because there is no established framework. In the existing literature, there is a lack of comparative analysis of machine learning models for cryptocurrency price prediction. To fill this gap in the literature, the research employs a 3-fold TimeSeriesSplit cross-validation technique and sliding window approach to compare the predictive abilities of five machine learning models across ten different cryptocurrencies. MAE and RMSE scores are employed to assess these models' performance. The results reveal that trained models perform significantly better than the baseline model, with NN models showing particularly impressive results and outperforming other comparable machine-learning models. The LSTM model exhibits the highest predictive power of the models analyzed, as well as the lowest MAE and RMSE scores across a variety of cryptocurrencies.

Thus, this thesis provides a valuable contribution to the field by comparing the price prediction of ten cryptocurrencies that have not been extensively researched. The findings highlight the disparities in predictive ability among different machine learning approaches for various cryptocurrencies, which can help in choosing the best algorithm for a given cryptocurrency and potentially result in more precise methods for forecasting the financial markets.

REFERENCES

Al-Yahyaee, K. H., Mensi, W., Al-Jarrah, I. M., Hamdi, A., & Kang, S. H. (2019). Volatility forecasting, downside risk, and diversification benefits of bitcoin and oil and international commodity markets: A comparative analysis with yellow metal. The North American Journal of Economics and Finance, 49, 104–120. https://doi.org/10.1016/j.najef.2019.04.001

Ammer, M. A., & Aldhyani, T. H. (2022). Deep learning algorithm to predict cryptocurrency fluctuation prices: Increasing investment awareness. *Electronics*, *11*(15), 2349. https://doi.org/10.3390/electronics11152349

Assaad, R. H., & Fayek, S. (2021). Predicting the price of crude oil and its fluctuations using computational econometrics: Deep Learning, LSTM, and Convolutional Neural Networks. *Econometric Research in Finance*, *6*(2), 119–137. https://doi.org/10.2478/erfin-2021-0006

Barnwal, A., Bharti, H. P., Ali, A., & Singh, V. (2019). Stacking with neural network for cryptocurrency investment. *2019 New York Scientific Data Summit (NYSDS)*. https://doi.org/10.1109/nysds.2019.8909804

Breiman, L. (n.d.). *Random forests* . Retrieved December 13, 2022, from https://link.springer.com/content/pdf/10.1023/A:1010933404324.pdf

Brownlee, J. (2020, August 27). *How to grid search deep learning models for time series forecasting*. MachineLearningMastery.com. Retrieved December 20, 2023, from https://machinelearningmastery.com/how-to-grid-search-deep-learning-models-for-time-series forecasting/

Chen, J. (2022). Analysis of bitcoin price prediction using machine learning. *Journal of Risk and Financial Management*, *16*(1), 51. https://doi.org/10.3390/jrfm16010051

Cocco, L., Tonelli, R., & Marchesi, M. (2021). Predictions of bitcoin prices through machine learning based frameworks. *PeerJ Computer Science*, *7*. https://doi.org/10.7717/peerj-cs.413

Erfanian, S., Zhou, Y., Razzaq, A., Abbas, A., Safeer, A. A., & Li, T. (2022). Predicting bitcoin (BTC) price in the context of economic theories: A machine learning approach. *Entropy*, *24*(10), 1487. https://doi.org/10.3390/e24101487

Fang, F., Ventre, C., Basios, M., Kanthan, L., Martinez-Rego, D., Wu, F., & Li, L. (2022). Cryptocurrency trading: A comprehensive survey. *Financial Innovation*, *8*(1). https://doi.org/10.1186/s40854 021-00321-6

Fleischer, J. P., von Laszewski, G., Theran, C., & Parra Bautista, Y. J. (2022). Time series analysis of cryptocurrency prices using long short-term memory. *Algorithms*, *15*(7), 230. https://doi.org/10.3390/a15070230

Hansun, S., Wicaksana, A., & Khaliq, A. Q. (2022). Multivariate cryptocurrency prediction: Comparative analysis of three recurrent neural networks approaches. *Journal of Big Data*, *9*(1). https://doi.org/10.1186/s40537-022-00601-7

Huang, X., Lin, J., & Wang, P. (2022). Are institutional investors marching into the crypto market? *Economics Letters*, *220*, 110856. https://doi.org/10.1016/j.econlet.2022.110856

Jaquart, P., Dann, D., & Weinhardt, C. (2021). Short-term bitcoin market prediction via machine learning. *The Journal of Finance and Data Science*, *7*, 45–66. https://doi.org/10.1016/j.jfds.2021.03.001

Kuhn, M., & Johnson, K. (2013). Applied predictive modeling. https://doi.org/10.1007/978-1-4614-6849 3

Kumar, A. (2022, November 6). Differences: Decision tree & random forest. Data Analytics. R etrieved January 1, 2023, from https://vitalflux.com/differences-between-decision-tree-random forest/

Mudassir, M., Bennbaia, S., Unal, D., & Hammoudeh, M. (2020). Time-series forecasting of bitcoin prices using high-dimensional features: A machine learning approach. *Neural Computing and Applications*. https://doi.org/10.1007/s00521-020-05129-6

Nakamoto, S. (2009). *A peer-to-peer electronic cash system*. Bitcoin. Retrieved December 20, 2022, from https://bitcoin.org/en/bitcoin-paper

Nikou, M., Mansourfar, G., & Bagherzadeh, J. (2021). Stock price prediction using deep learning algorithm and its comparison with machine learning algorithms. *Intelligent Systems in Accounting, Finance and Management*, *26*(4), 164–174. https://doi.org/10.1002/isaf.1459

Oyewola, D. O., Dada, E. G., & Ndunagu, J. N. (2022). A novel hybrid walk-forward ensemble optimization for time series cryptocurrency prediction. *Heliyon*, *8*(11). https://doi.org/10.1016/j.heliyon.2022.e11862

S, M., Mohta, M., & Rangaswamy, S. (2022). Ethereum Price prediction using Machine Learning Techniques – A Comparative Study. *International Journal of Engineering Applied Sciences and Technology*, *7*(2), 137–142. https://doi.org/10.33564/ijeast.2022.v07i02.018

Sen, J., & Mehtab, S. (2021). Design and analysis of robust deep learning models for stock price prediction. *Artificial Intelligence*. https://doi.org/10.5772/intechopen.99982

Shahzad, S. J., Bouri, E., Roubaud, D., Kristoufek, L., & Lucey, B. (2019). Is bitcoin a better safe-haven investment than gold and commodities? International Review of Financial Analysis, 63, 322–330. https://doi.org/10.1016/j.irfa.2019.01.002

Soni, P., Tewari, Y., & Krishnan, D. (2022). Machine learning approaches in stock price prediction: A systematic review. *Journal of Physics: Conference Series*, *2161*(1), 012065. https://doi.org/10.1088/1742-6596/2161/1/012065

Tandon, S., Tripathi, S., Saraswat, P., & Dabas, C. (2019). Bitcoin price forecasting using LSTM and 10-fold cross validation. *2019 International Conference on Signal Processing and Communication (ICSC)*. https://doi.org/10.1109/icsc45622.2019.8938251

Wang, H., Wang, J., Cao, L., Li, Y., Sun, Q., & Wang, J. (2021). A stock closing price prediction model based on CNN-BISLSTM. *Complexity*, *2021*, 1–12. https://doi.org/10.1155/2021/5360828

Wang, P., Liu, X., & Wu, S. (2022). Dynamic linkage between Bitcoin and traditional financial assets: A comparative analysis of different time frequencies. *Entropy*, *24*(11), 1565. https://doi.org/10.3390/e24111565

APPENDIX A: Cryptocurrency TimeSeriesSplit

In Appendix A the *TimeSeriesSplit* periods for each cryptocurrency are showcased (starting with period 1 from top to bottom).

FIGURE 15: *BITCOIN TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT*

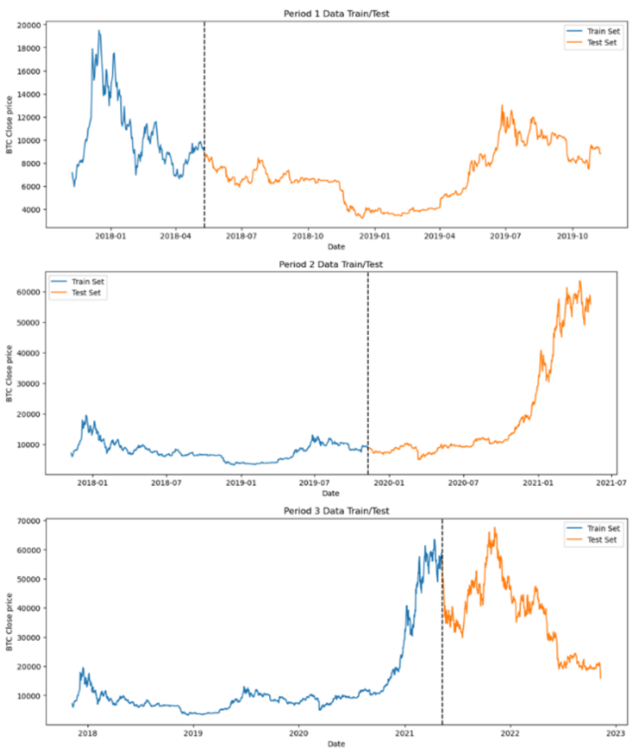


FIGURE 16: *ETHEREUM TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT*

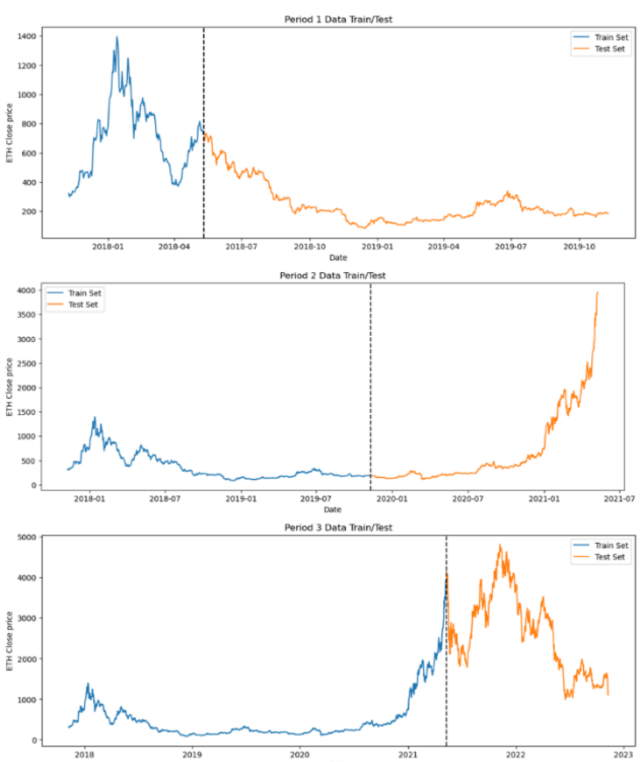


FIGURE 17: ***BINANCE COIN TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT***

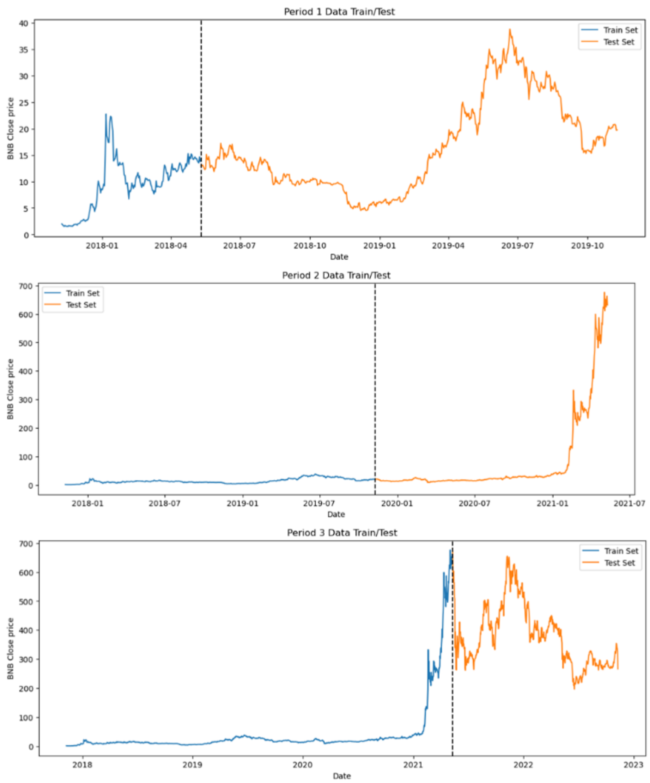


FIGURE 18: ***RIPPLE TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT***

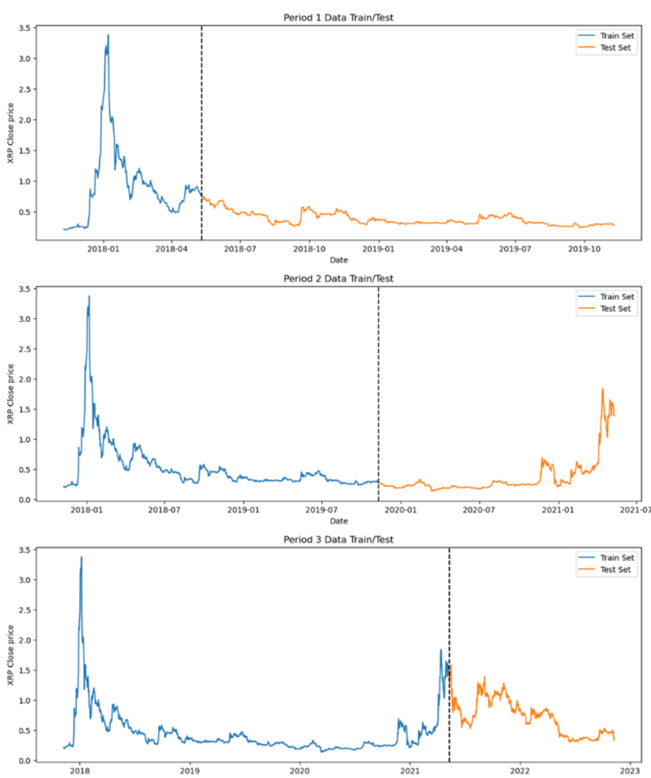


FIGURE 19: *DOGECOIN TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT*

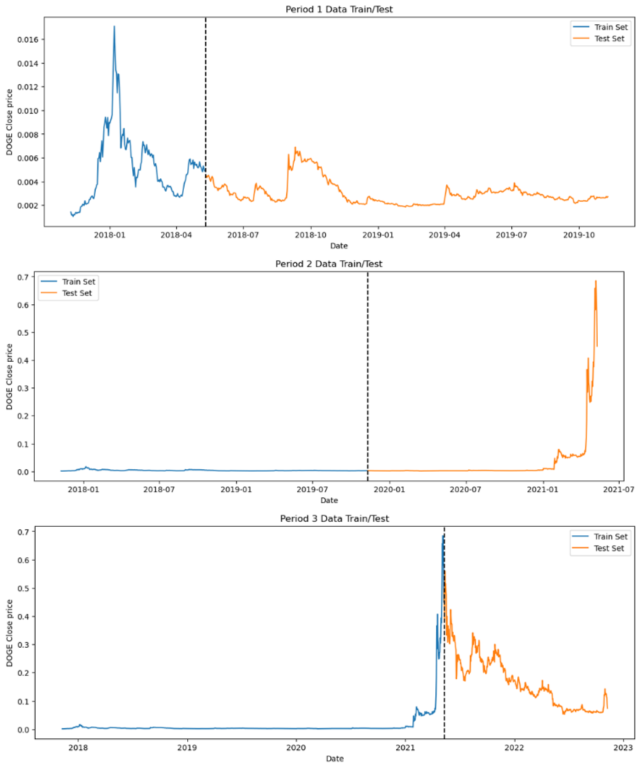


FIGURE 20: *CARDANO TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT*

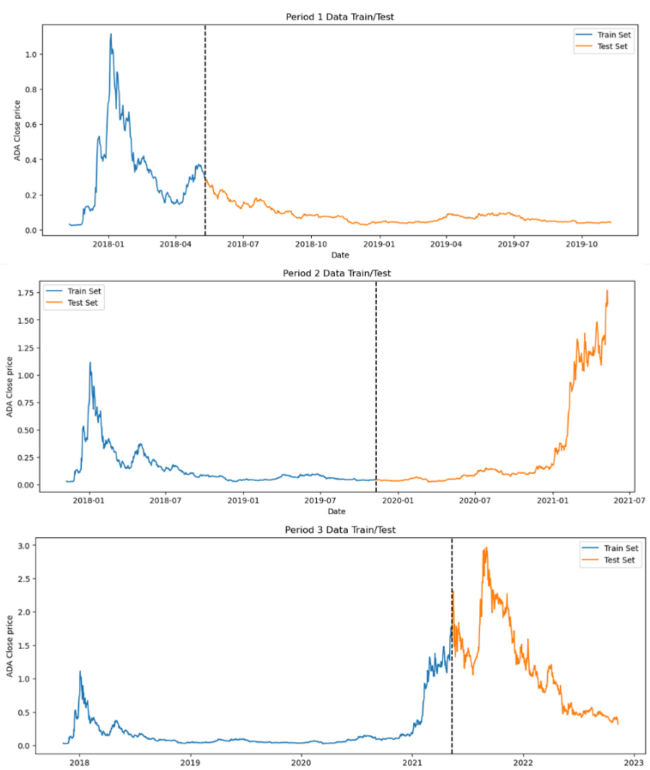


FIGURE 21: *POLYGON TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT*

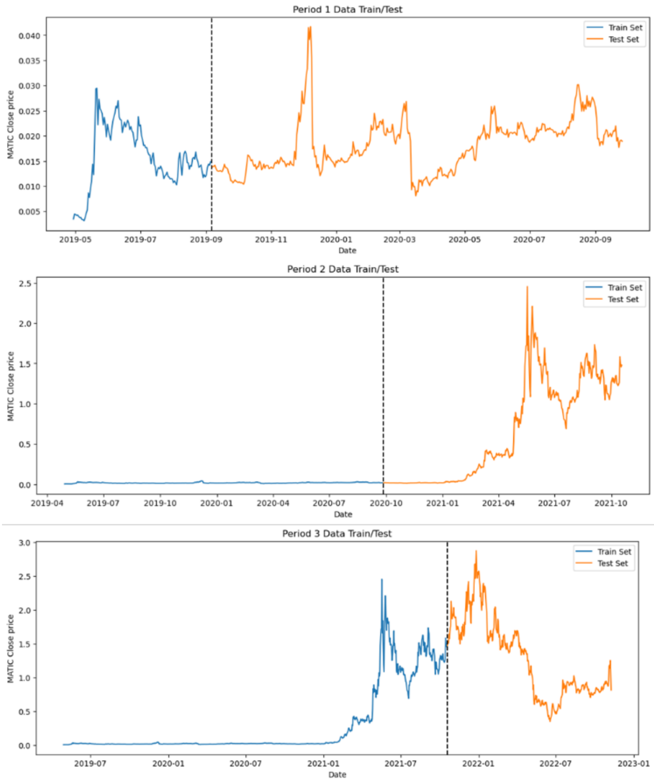


FIGURE 22: *POLKADOT TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT*

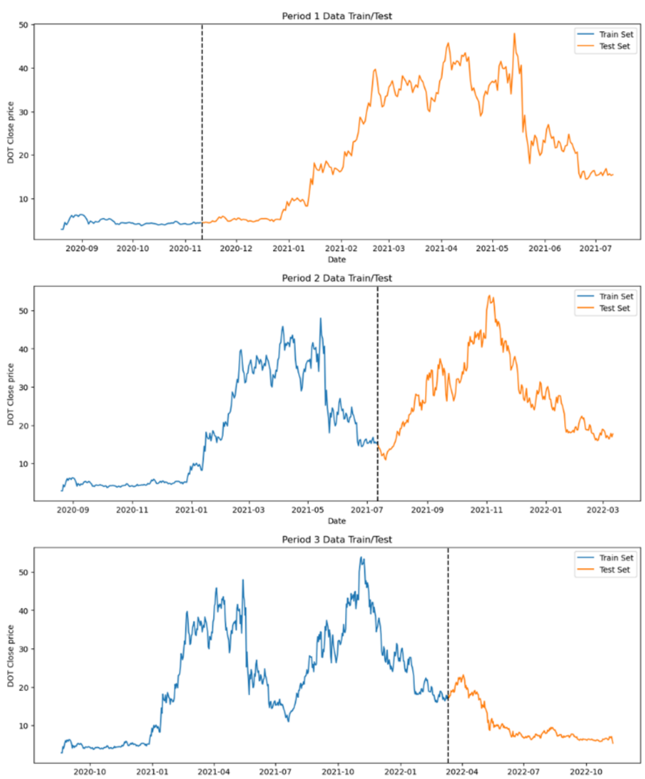


FIGURE 23: *LITECOIN TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT*

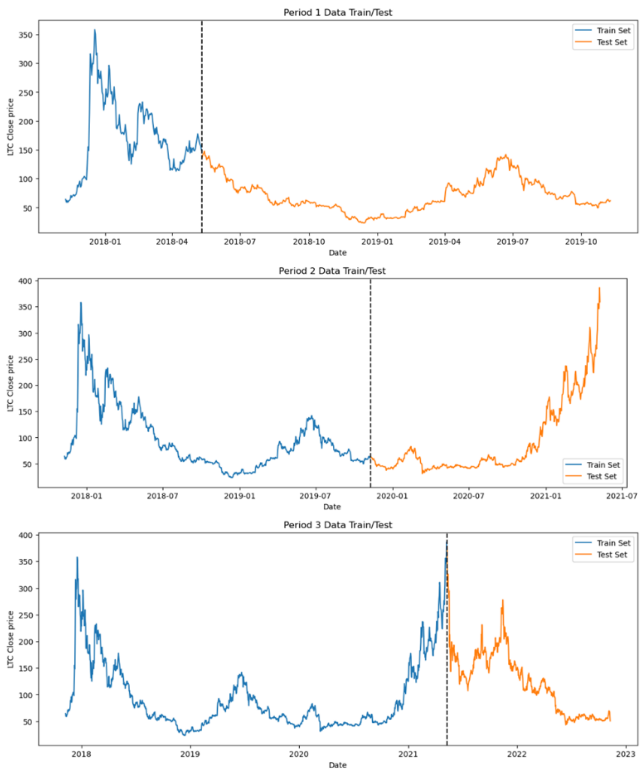
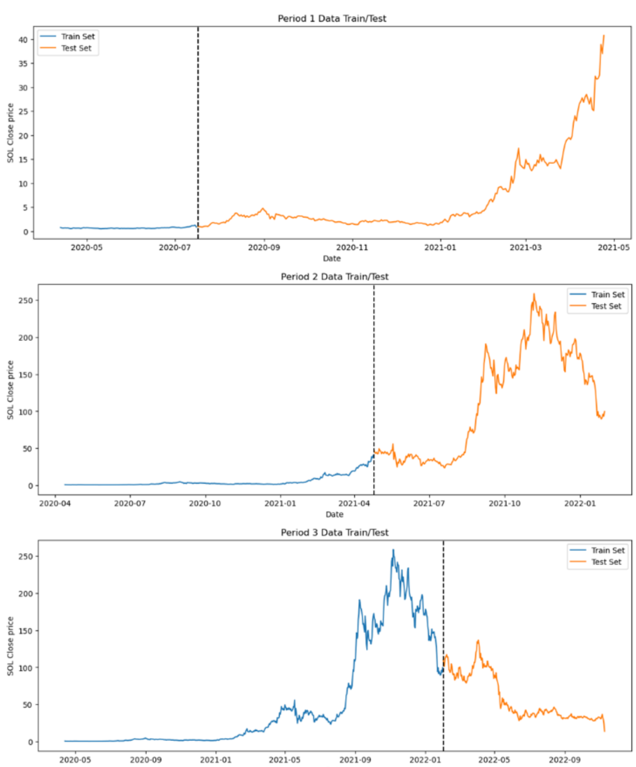


FIGURE 24: *SOLANA TRAIN/TEST PERIOD (1, 2 AND 3) SPLIT*



APPENDIX B: Error pattern visualization for period 1

This appendix includes the error pattern visualization (predicted vs. actual) for period 1. The figure on top is the RF model and from top to bottom left to right XGBoost, MLP, RNN and LSTM is shown.

FIGURE 25: *BITCOIN ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN AND LSTM)*

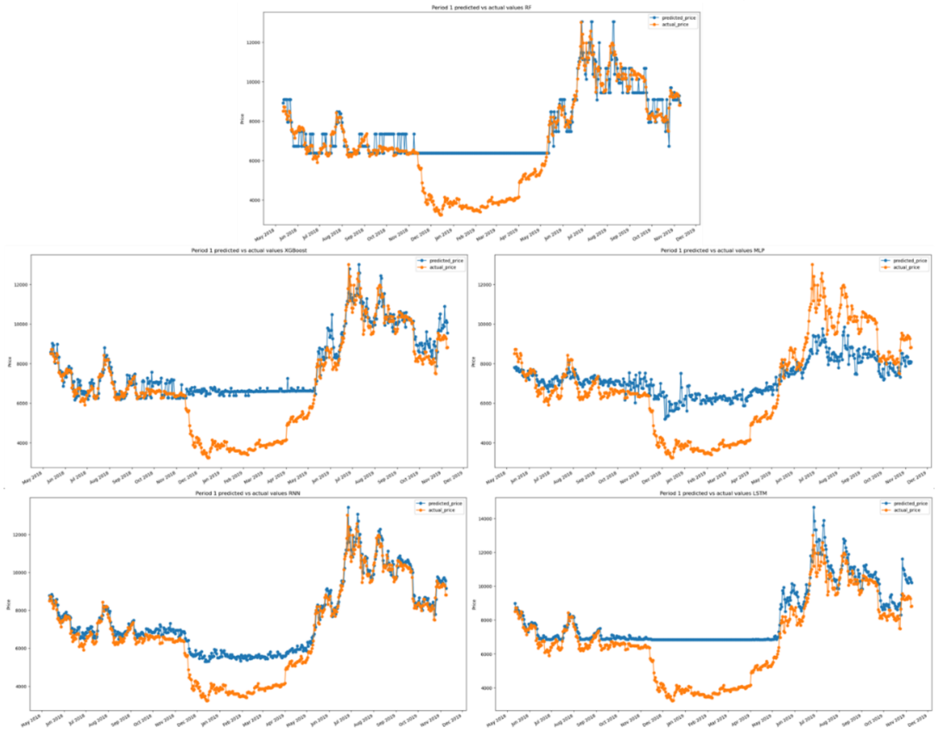


FIGURE 26: *ETHEREUM ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN AND LSTM)*

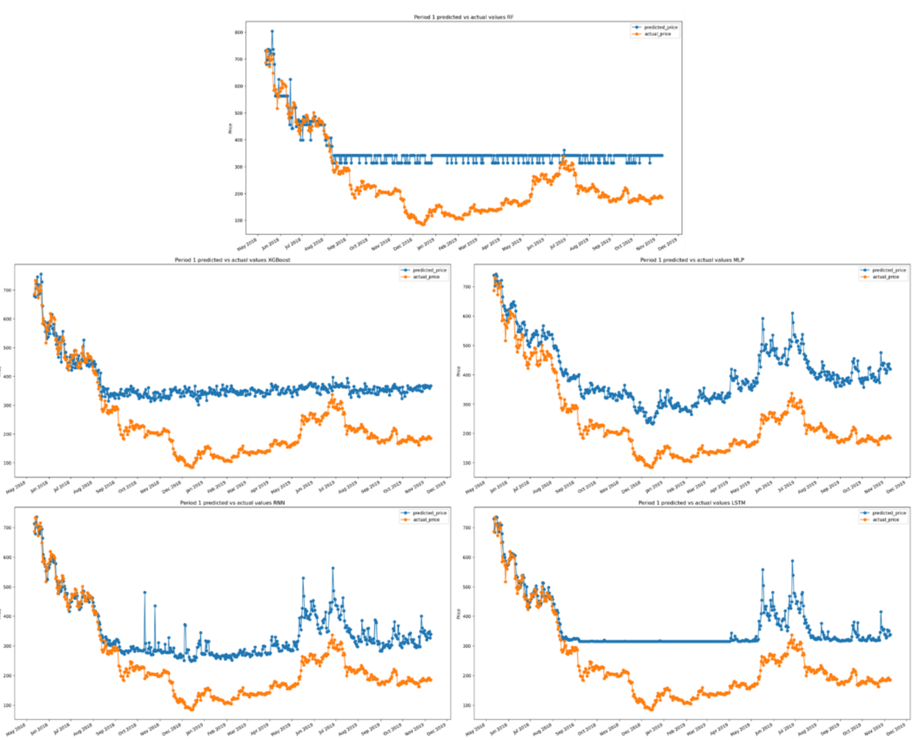


FIGURE 27: *BINANCE COIN ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN, LSTM)*

FIGURE 28: *RIPPLE ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN AND LSTM)*

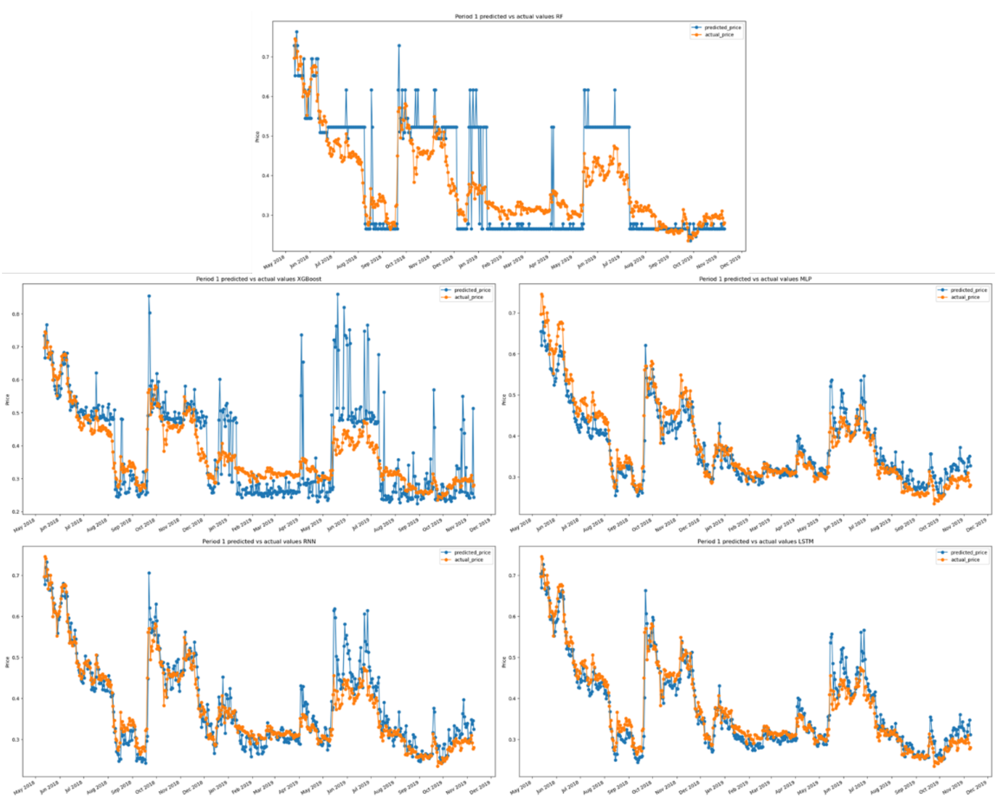


FIGURE 29: *DOGECOIN ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN AND LSTM)*

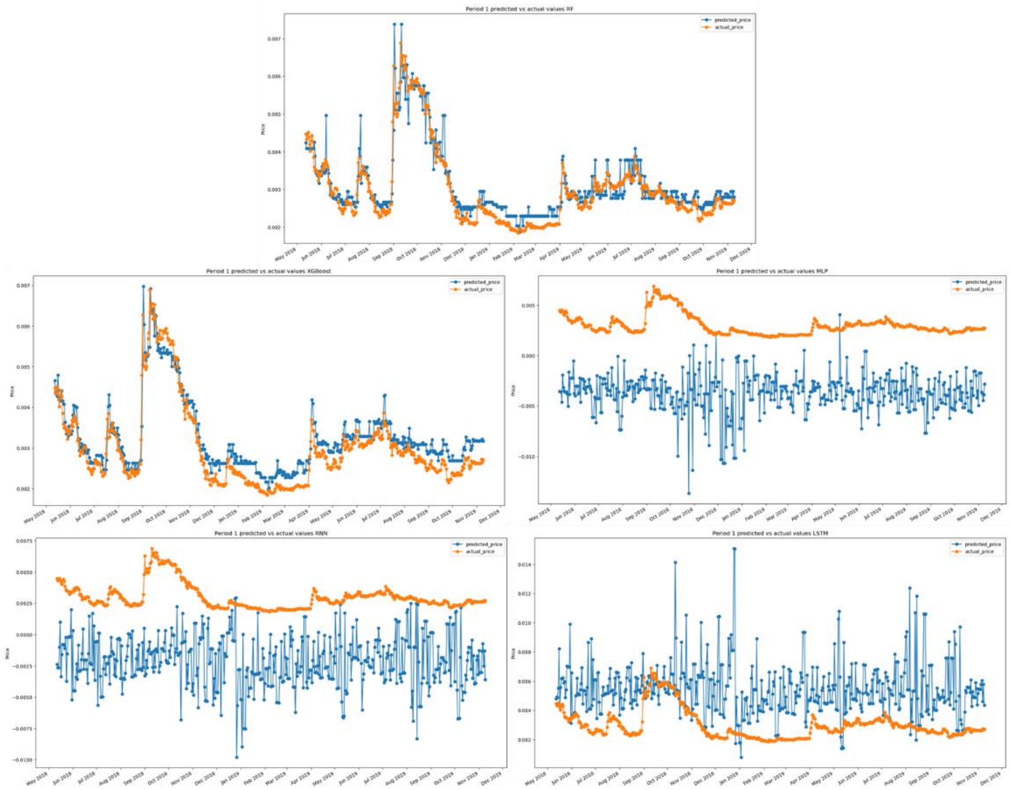


FIGURE 30: *POLKADOT ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN AND LSTM)*

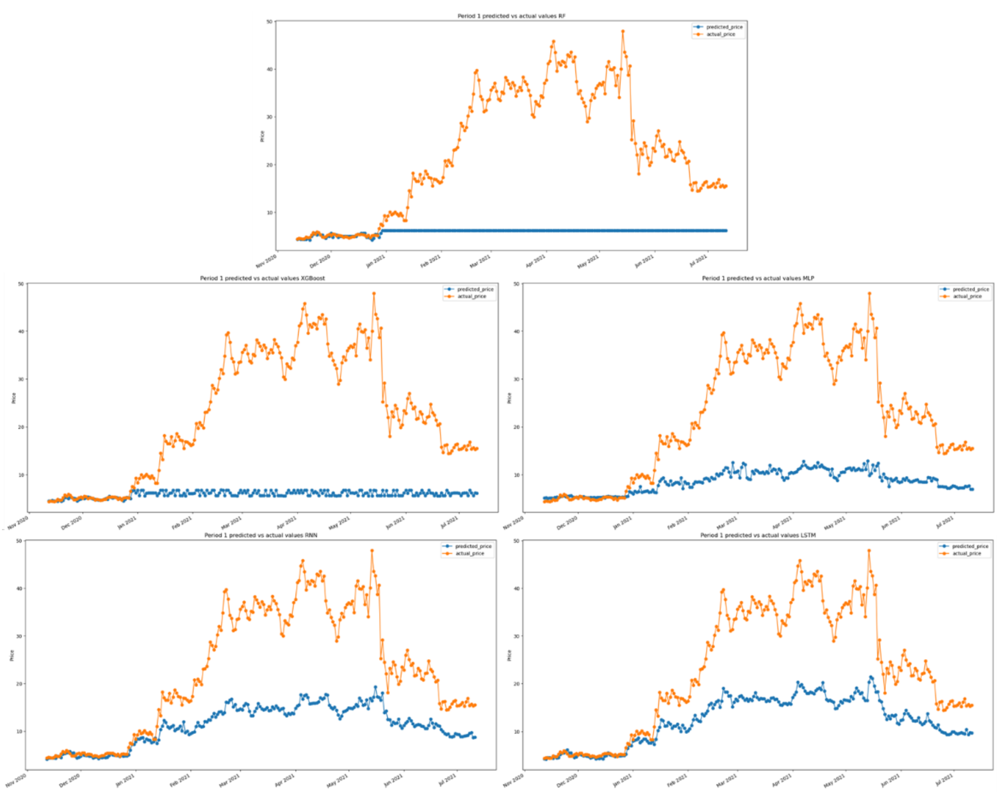


FIGURE 31: *CARDANO ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN AND LSTM)*

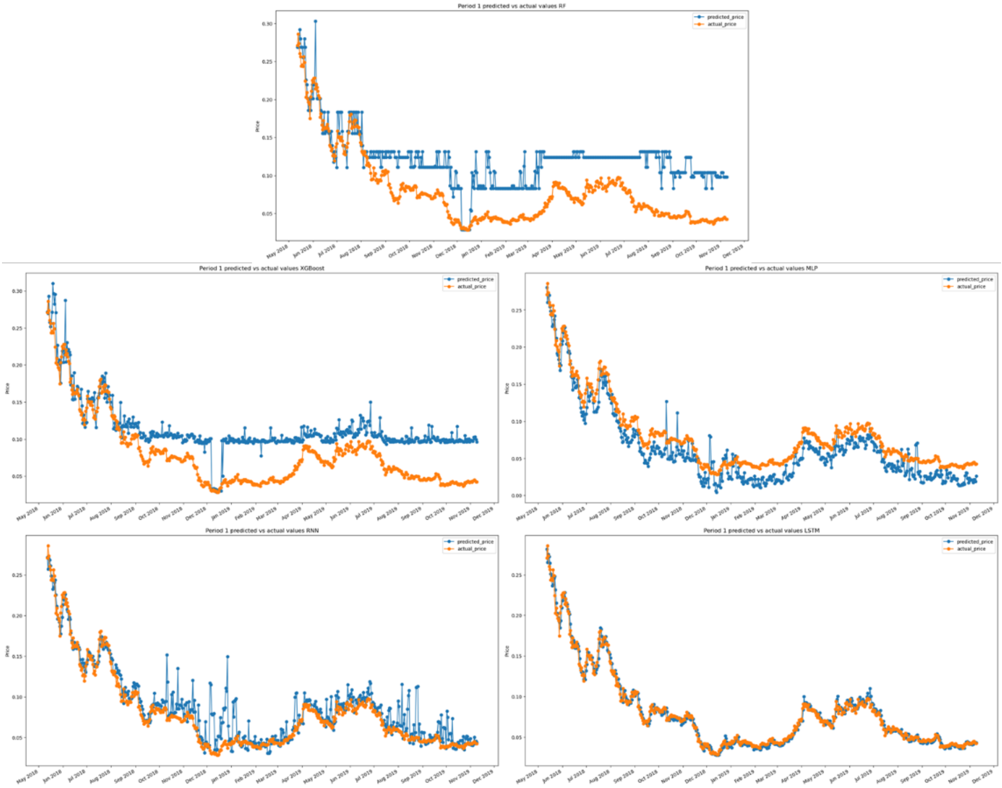


FIGURE 32: *POLYGON ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN AND LSTM)*

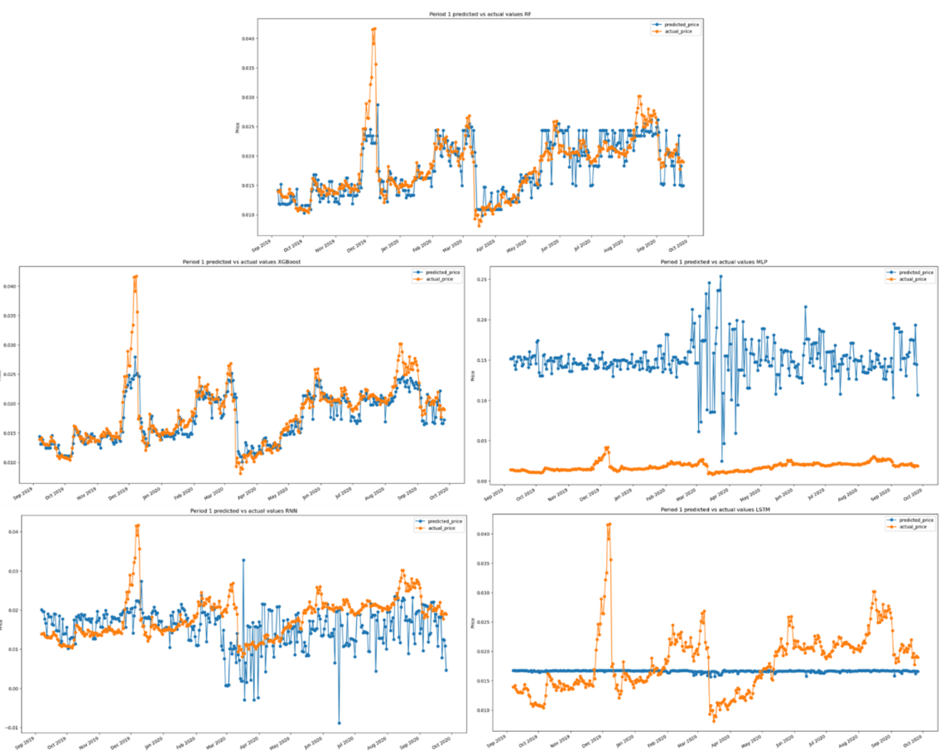


FIGURE 33: *LITECOIN ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN AND LSTM)*

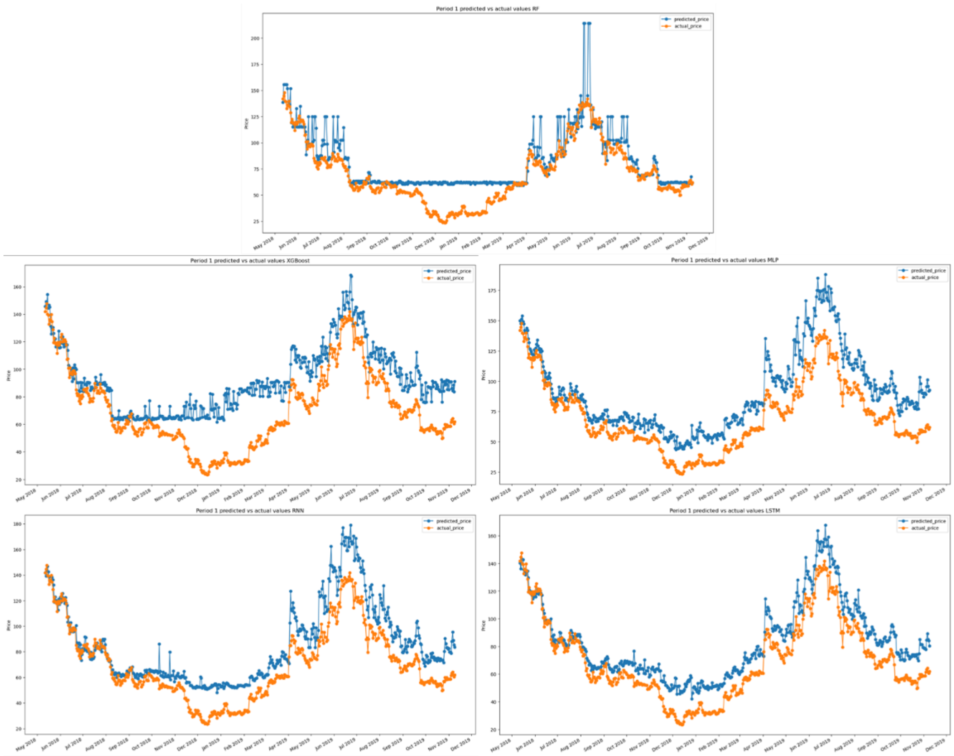
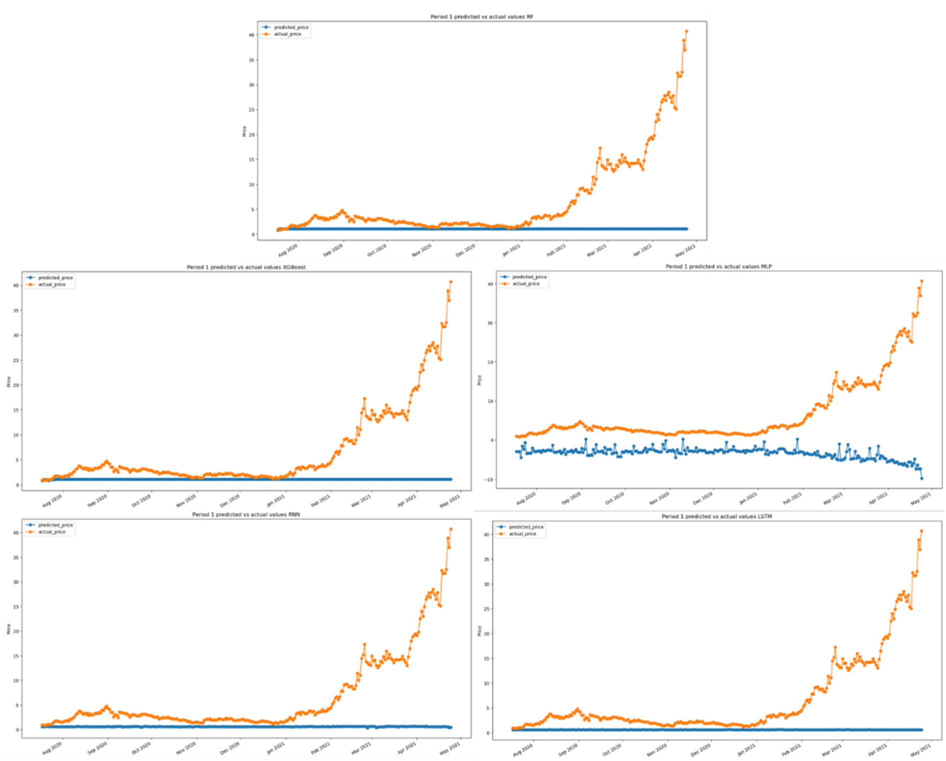


FIGURE 34: *SOLANA ERROR PATTERN VISUALIZATION FOR PERIOD 1 (RF, XGBOOST, MLP, RNN AND LSTM)*



APPENDIX C: Error pattern visualization for period 2

This appendix includes the error pattern visualization (predicted vs. actual) for period 2. The figure on top is the RF model and from top to bottom left to right XGBoost, MLP, RNN and LSTM is shown.

FIGURE 35: *BITCOIN ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN AND LSTM)*

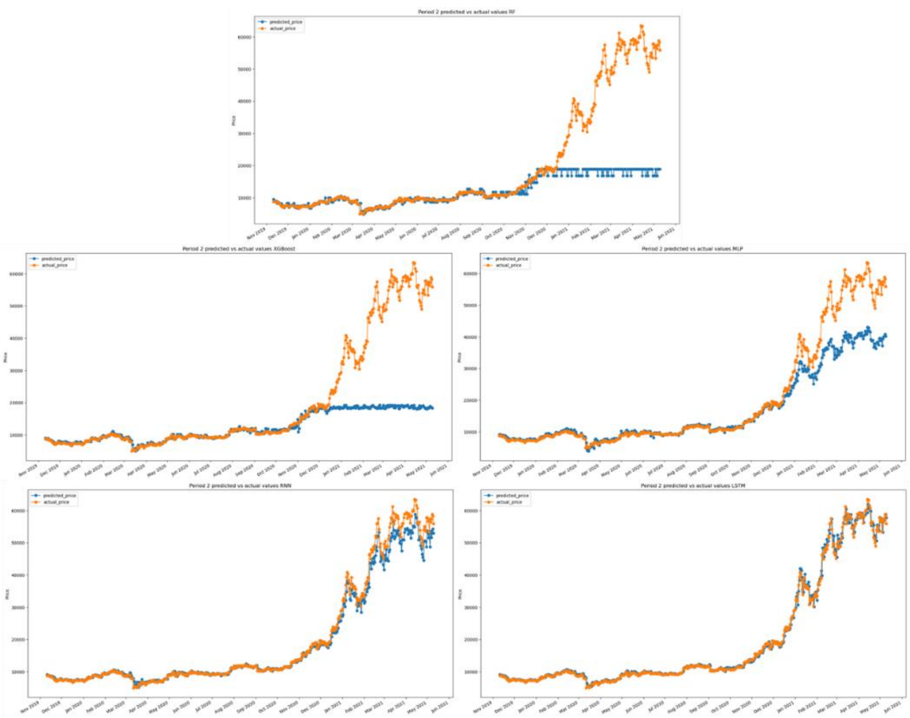


FIGURE 36: *ETHEREUM ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN AND LSTM)*

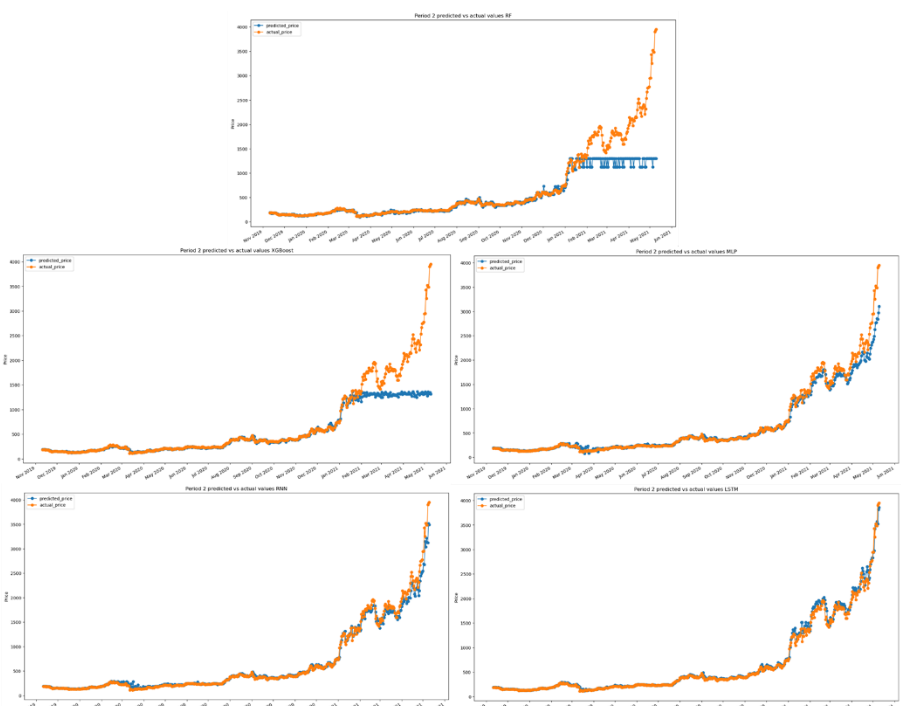


FIGURE 37: *BINANCE COIN ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN, LSTM)*

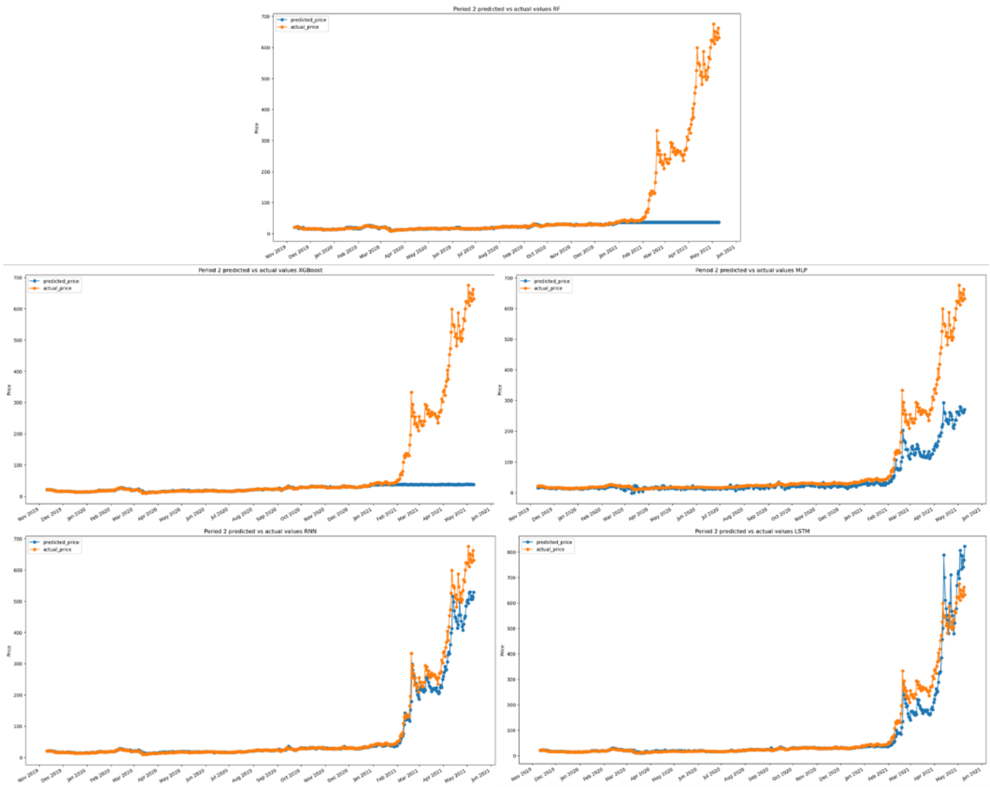


FIGURE 38: *RIPPLE ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN AND LSTM)*

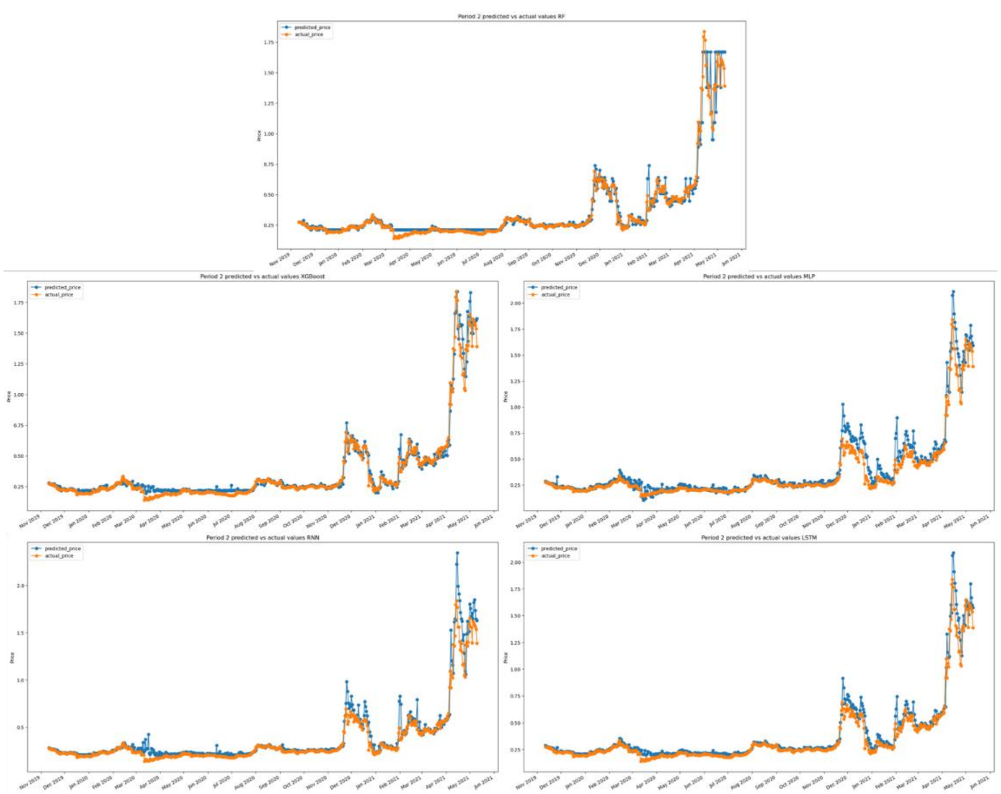


FIGURE 39: *DOGECOIN ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN AND LSTM)*

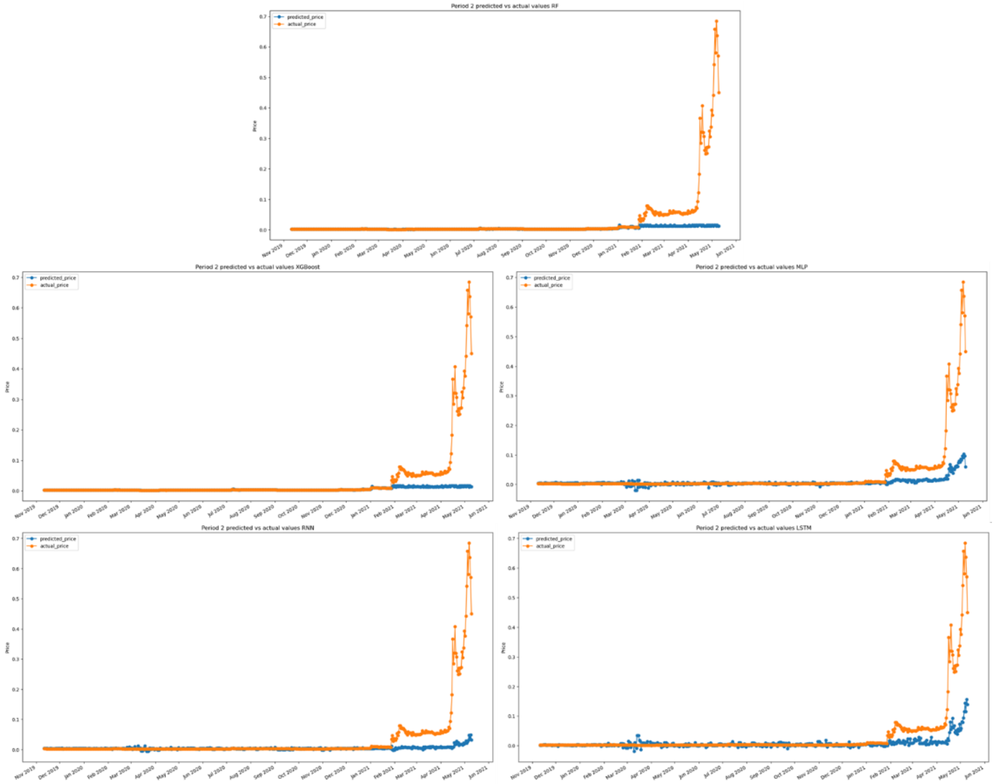


FIGURE 40: *POLKADOT ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN AND LSTM)*

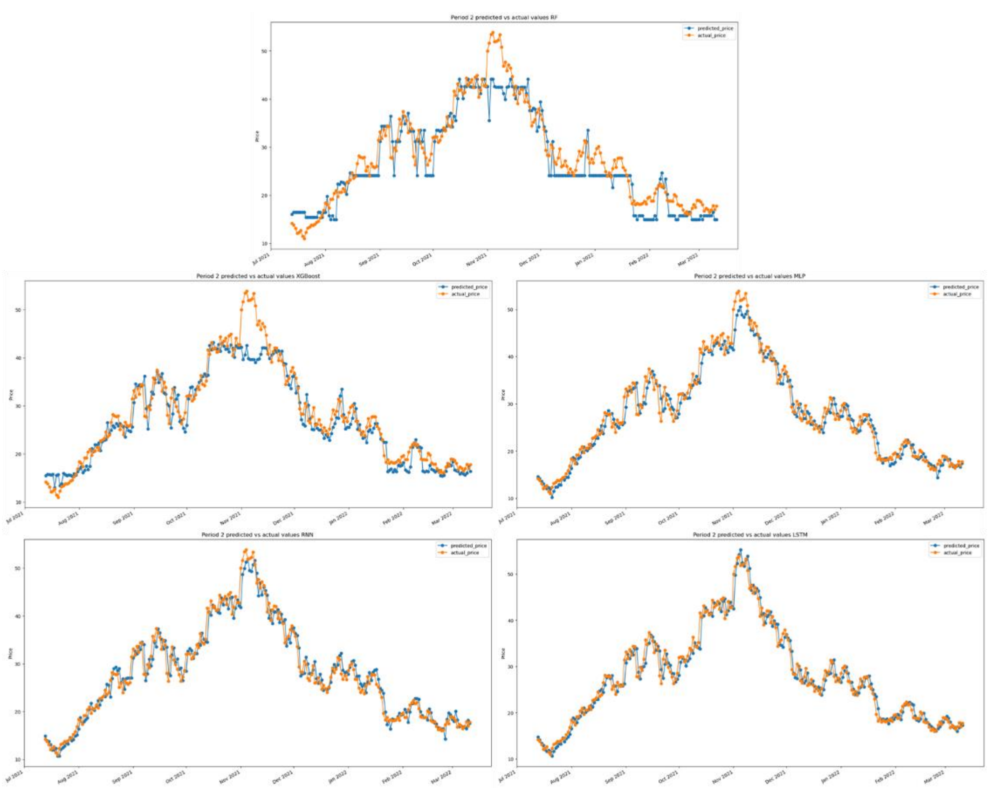


FIGURE 41: *CARDANO ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN AND LSTM)*

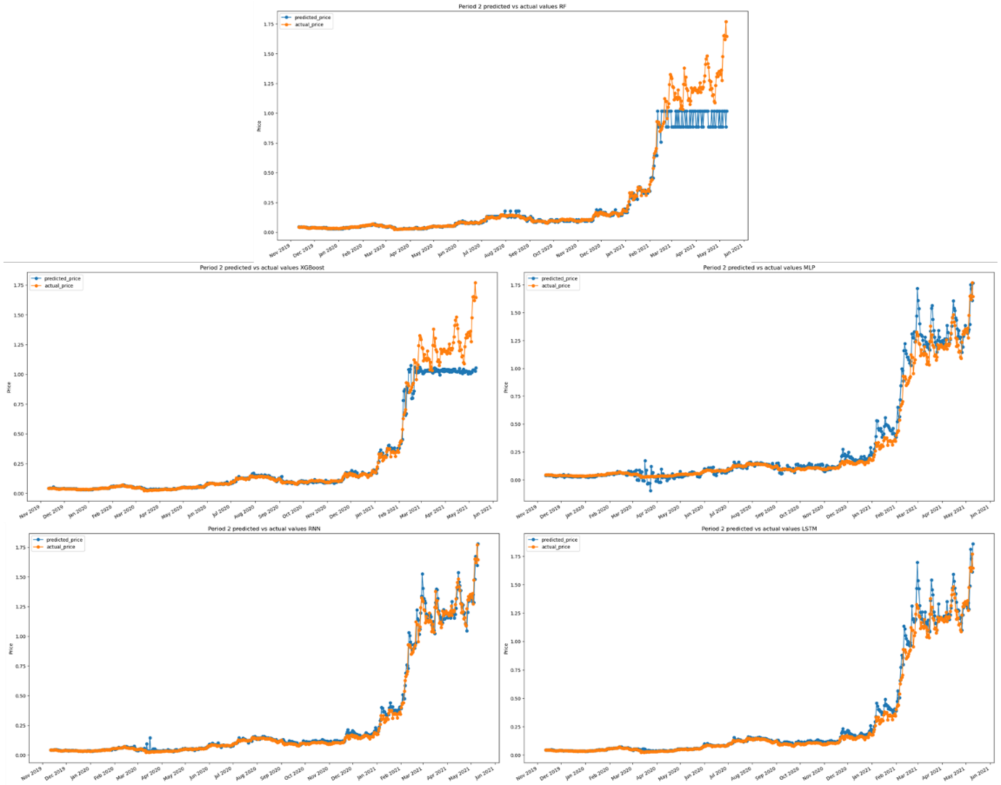


FIGURE 42: *POLYGON ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN AND LSTM)*

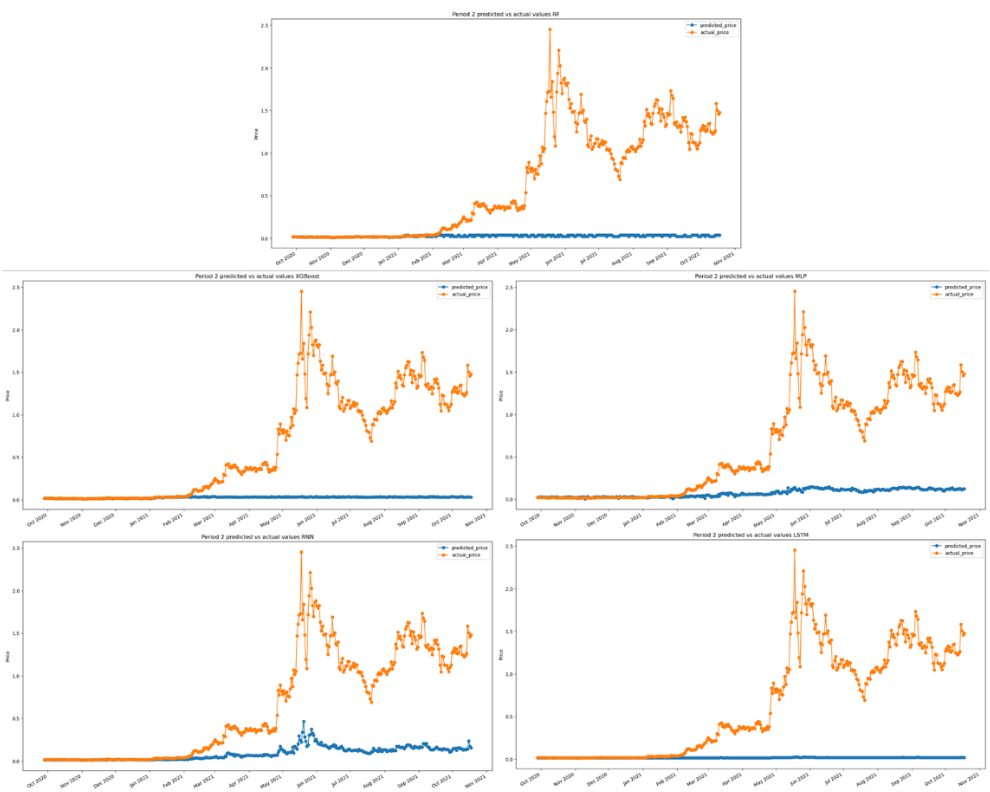


FIGURE 43: *LITECOIN ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN AND LSTM)*

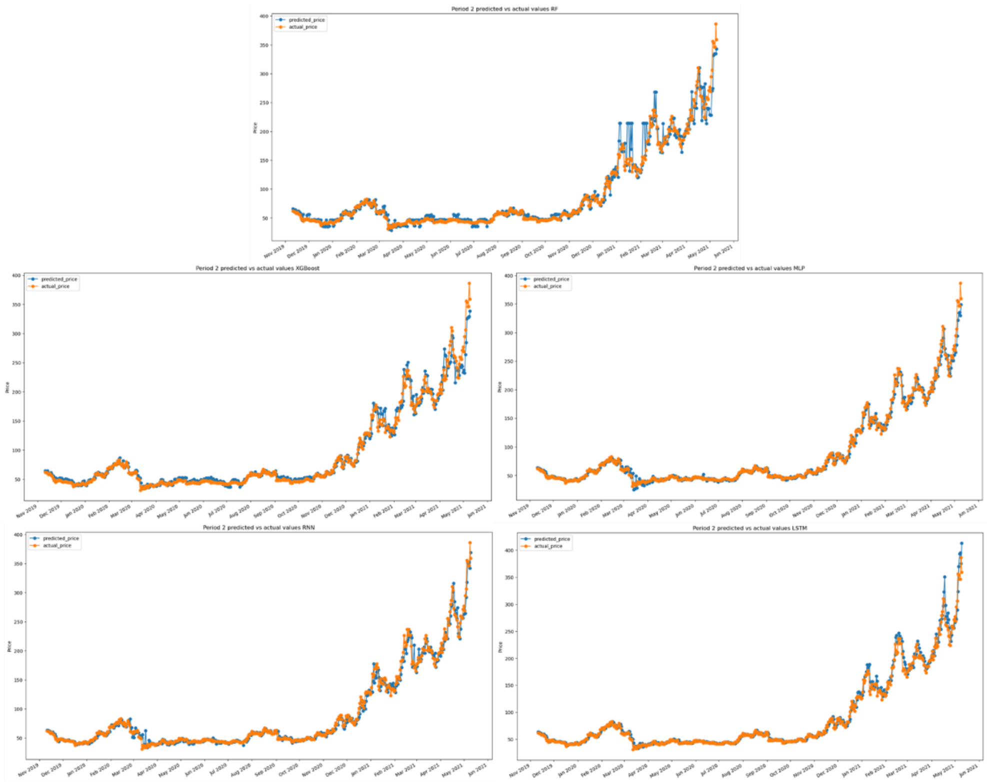
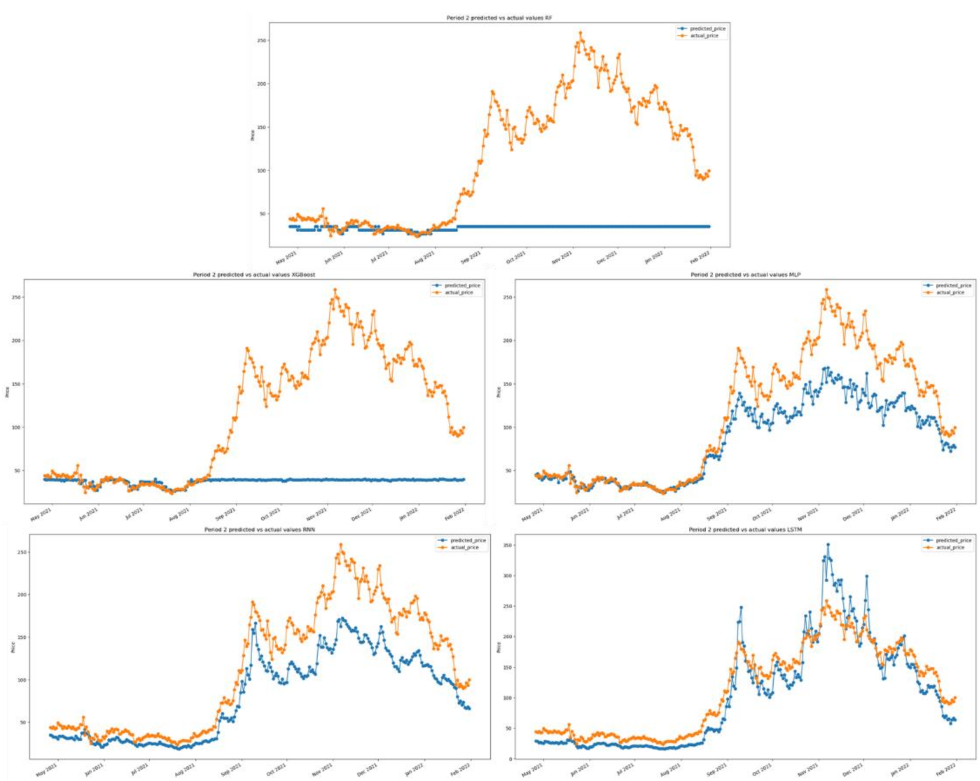


FIGURE 44: *SOLANA ERROR PATTERN VISUALIZATION FOR PERIOD 2 (RF, XGBOOST, MLP, RNN AND LSTM)*



APPENDIX D: Cryptocurrency model hyperparameters

In Appendix D the hyperparameters that are used for each model and cryptocurrency are showcased.

TABLE 17: *BITCOIN RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS*

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	1000
	min_samples_split	2, 5, 10	5
	min_samples_leaf	1, 2, 4	4
	max_depth	10, 20, 50, 100	50
XGBoost	n_estimators	100, 250, 500, 1000, 2000	250
	max_depth	3, 6, 9, 12	9
	learning_rate	0.01, 0.03, 0.05, 0.1	0.03
MLP	hidden_layer_sizes	50, 100, 200, 500	500
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.001
RNN	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	8
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005

TABLE 18: *ETHEREUM RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS*

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	500
	min_samples_split	2, 5, 10	10
	min_samples_leaf	1, 2, 4	1
	max_depth	10, 20, 50, 100	50
XGBoost	n_estimators	100, 250, 500, 1000, 2000	2000
	max_depth	3, 6, 9, 12	6
	learning_rate	0.01, 0.03, 0.05, 0.1	0.05
MLP	hidden_layer_sizes	50, 100, 200, 500	500
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.005
RNN	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	8
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005

TABLE 19

BINANCE COIN RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	100
	min_samples_split	2, 5, 10	5
	min_samples_leaf	1, 2, 4	4
	max_depth	10, 20, 50, 100	100
XGBoost	n_estimators	100, 250, 500, 1000, 2000	1000
	max_depth	3, 6, 9, 12	12
	learning_rate	0.01, 0.03, 0.05, 0.1	0.01
MLP	hidden_layer_sizes	50, 100, 200, 500	500
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.001
RNN	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	4
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.0005

TABLE 20

RIPPLE RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	100
	min_samples_split	2, 5, 10	5
	min_samples_leaf	1, 2, 4	4
	max_depth	10, 20, 50, 100	100
XGBoost	n_estimators	100, 250, 500, 1000, 2000	500
	max_depth	3, 6, 9, 12	3
	learning_rate	0.01, 0.03, 0.05, 0.1	0.03
MLP	hidden_layer_sizes	50, 100, 200, 500	500
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.005
RNN	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	8
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	32
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005

TABLE 21

DOGECOIN RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	500
	min_samples_split	2, 5, 10	5
	min_samples_leaf	1, 2, 4	1
	max_depth	10, 20, 50, 100	10
XGBoost	n_estimators	100, 250, 500, 1000, 2000	1000
	max_depth	3, 6, 9, 12	12
	learning_rate	0.01, 0.03, 0.05, 0.1	0.01
MLP	hidden_layer_sizes	50, 100, 200, 500	500
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.001
RNN	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	8
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	8
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.0005

TABLE 22

CARDANO RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	100
	min_samples_split	2, 5, 10	5
	min_samples_leaf	1, 2, 4	4
	max_depth	10, 20, 50, 100	100
XGBoost	n_estimators	100, 250, 500, 1000, 2000	500
	max_depth	3, 6, 9, 12	3
	learning_rate	0.01, 0.03, 0.05, 0.1	0.03
MLP	hidden_layer_sizes	50, 100, 200, 500	500
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.001
RNN	epochs	30, 60, 120, 150	60
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	60
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005

TABLE 23

POLYGON RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	500
	min_samples_split	2, 5, 10	5
	min_samples_leaf	1, 2, 4	1
	max_depth	10, 20, 50, 100	10
XGBoost	n_estimators	100, 250, 500, 1000, 2000	1000
	max_depth	3, 6, 9, 12	12
	learning_rate	0.01, 0.03, 0.05, 0.1	0.01
MLP	hidden_layer_sizes	50, 100, 200, 500	500
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.005
RNN	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.0005

TABLE 24

POLKADOT RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	100
	min_samples_split	2, 5, 10	5
	min_samples_leaf	1, 2, 4	4
	max_depth	10, 20, 50, 100	100
XGBoost	n_estimators	100, 250, 500, 1000, 2000	1000
	max_depth	3, 6, 9, 12	12
	learning_rate	0.01, 0.03, 0.05, 0.1	0.01
MLP	hidden_layer_sizes	50, 100, 200, 500	500
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.001
RNN	epochs	30, 60, 120, 150	60
	batch_size	4, 8, 16, 32, 64, 128	8
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	60
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.0005

TABLE 25

LITECOIN RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	500
	min_samples_split	2, 5, 10	2
	min_samples_leaf	1, 2, 4	1
	max_depth	10, 20, 50, 100	20
XGBoost	n_estimators	100, 250, 500, 1000, 2000	1000
	max_depth	3, 6, 9, 12	3
	learning_rate	0.01, 0.03, 0.05, 0.1	0.01
MLP	hidden_layer_sizes	50, 100, 200, 500	200
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.005
RNN	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	8
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.0005

TABLE 26

SOLANA RF, XGBOOST, MLP, RNN AND LSTM HYPERPARAMETERS

Model	Hyperparameter	Input values	Used values
Random Forest	n_estimators	100, 250, 500, 1000, 2000	100
	min_samples_split	2, 5, 10	5
	min_samples_leaf	1, 2, 4	4
	max_depth	10, 20, 50, 100	100
XGBoost	n_estimators	100, 250, 500, 1000, 2000	1000
	max_depth	3, 6, 9, 12	12
	learning_rate	0.01, 0.03, 0.05, 0.1	0.01
MLP	hidden_layer_sizes	50, 100, 200, 500	200
	learning_rate_init	0.005, 0.001, 0.0005, 0.0001	0.001
RNN	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	8
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.005
LSTM	epochs	30, 60, 120, 150	120
	batch_size	4, 8, 16, 32, 64, 128	16
	learning_rate	0.005, 0.001, 0.0005, 0.0001	0.0005