Tree Based Predictions and Forecast on Bitcoin Prices

Oğuzhan AYDIN
Middle East Technical University
Ankara, Turkey
Aydin.oguzhan_01@metu.edu.tr

***Abstract*— This paper includes a study on the monthly changes of Bitcoin prices since June 2010 and in the study, better understanding of bitcoin prices and forecasting the new prices are aimed. In the study, one non-tree-based model ARIMA and 6 main tree-based algorithms, decision tree, random forest, gradient boosting, LightGBM, catboost, adaboost, are applied on the data. The reason for applying ARIMA is its strength in capturing temporal dependencies and seasonality inherent in time series data, which complements the strengths of tree-based models. Tree-Based algorithms are evaluated based on evaluation metric criteria, mean absolute scaled error, root mean squared error and mean absolute percentage error. Among these algorithms, three algorithms with lowest mean absolute scaled error value are taken into consideration and new hybrid models are created with the combinations of these three algorithms. At the end of the study, the performance of these hybrid models is analyzed and compared to identify the most effective approach for forecasting Bitcoin prices. Also, the results demonstrate whether there are the superiority of the hybrid models over individual algorithms in terms of forecasting accuracy. Finally, give an understanding about whether there is a bear market or bull market in cryptocurrency.**

Keywords—Tree Based Algorithms, Decision Tree, ARIMA, Adaboost, Random Forest, LightGBM, Catboost, Gradient Boosting, Cryptocurrency, Bitcoin

I. INTRODUCTION

First introduced worldwide with Bitcoin in 2009, the world of cryptocurrency began to gain popularity from 2013 onwards and has now become an element breaking records and subject to laws enacted by governments. As a result, the cryptocurrency exchange, and its pioneer, the Bitcoin exchange, have been seen as an investment opportunity by everyone from famous economists to ordinary people on the street. However, the sudden rises, falls, and stagnations in cryptocurrencies have made some investors wealthy while causing significant financial losses for others. This Project aims exploring the pattern of Bitcoin prices with the dataset that is collected since June 2010, with different variables such as closing price of Bitcoin, volume of Bitcoin and average price of Bitcoin in the month and so on. After identifying the pattern, the second aim is to forecasting the Bitcoin prices for the next couple of months, as an answer to a economy world about whether markets are in bull or bear season.

In the project, fluctuations in the prices are determined by tree-based algorithms, and models are created to make sure patterns make sense. After models are created, every method was compared with each other based on performance evaluation metrics. Comparing all the individual tree-based models gives the best of three models among each other. Hybrid models are created by using various combinations of the best three models and suggest new models. The study then investigates different forecasting models to project future prices within each framework.

The findings from this project highlight the effectiveness of tree-based algorithms in capturing the complex patterns and fluctuations in Bitcoin prices. The study underscores the superiority of hybrid models over individual algorithms, demonstrating their enhanced accuracy in forecasting future prices. Moreover, the project provides valuable insights into the dynamics of the cryptocurrency market, offering a clearer understanding of when the market is in a bull or bear season. These results can guide investors in making informed decisions and contribute to the broader economic discourse on cryptocurrency trends and their implications for the financial world.

II. LITERATURE REVIEW

Abdullah H. Al-Nefaie and Theyazn H. H. Aldhyani explores the use of artificial intelligence (AI) models to predict Bitcoin prices in the article "Bitcoin Price Forecasting and Trading: Data Analytics Approaches". In the study, deep learning techniques are focused, specifically the Gated Recurrent Unit (GRU) and Multilayer Perceptron (MLP) models, to forecast Bitcoin price movements. The study demonstrates that AI models can provide accurate and reliable predictions of Bitcoin prices, which can help investors make informed decisions and potentially stabilize cryptocurrency markets. On another study, "Bitcoin Price Prediction Using Time Series Analysis and Machine Learning Techniques" by Gupta and Nain employs Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) models to predict Bitcoin prices using historical data. By capturing temporal dependencies within the data, the models demonstrated improved accuracy in forecasting future prices. The results suggest that advanced machine learning techniques can provide reliable predictions, aiding investors and traders in making informed decisions. In the study "Forecasting Bitcoin Price Using Interval Graph and ANN Model" by Murugesan, Shanmugaraja, and Vadivel (2022), it is investigated the application of interval graphs and Artificial Neural Networks (ANN) to predict Bitcoin prices. The study combines interval graphs to capture the temporal dynamics of Bitcoin prices and an ANN model to forecast future price movements.

All the other studies are focused on specific machine learning techniques and especially RNNs and ANNs. This study uniquely evaluates tree-based models using mean absolute scaled error, root mean squared error, and mean absolute percentage error. It also creates hybrid models from the best-performing tree-based algorithms to enhance forecasting accuracy. Furthermore, this study includes an analysis of market trends to determine if the cryptocurrency market is in a bear or bull phase, adding a practical financial analysis dimension that is not emphasized in the other studies. Compared to the other studies, this study is unique in terms of its comprehensive use of tree-based algorithms, hybrid model creation, and market trend analysis.

III. METHODOLOGY AND RESULTS

A. *Data Description and Preprocessing*

The studied dataset "Bitcoin Price| Daily Price | Weekly Update | USD" was taken from Kaggle. The dataset has 5014 observations and 7 variables which are Date, Price which is the bitcoin's average price in the specific date, Open which is opening price at the start of day's trading period and Close which is closing price, High and Low variables are the highest and lowest price of it in the specific day, Vol which is Bitcoin's trading volume within a specific unit, and finally Change which is the percentage change in Bitcoin's price between the opening and closing price. The data includes the days from 18^{th} June 2010 to 08^{th} April 2024.

There were errors such as punctuation, sign and formatting errors on the observations in the data set. 6 of the variables should be numeric format and one of them should be in date format. However, because of the related errors, these are not in the suggested format. Also, out of total 35.098 observations, there are 6 missing values and all of the missing values in the Volume column. Since it is a daily dataset and volume does not change much from day to day, especially in the year of 2011 that Bitcoin is not popular, these missing observations are filled by LOCF method. After arranging the observations and variables in the required format and filling in the missing values with the appropriate imputation method, the data set is converted to the time series format by indexing the Date column.

Moreover, for easy computation and for the purpose of the study, the data is converted to the monthly. The "open" variable represents the opening value at the start of each month, while the "high" and "low" variables represent the highest and lowest values within the month. The "price," "volume," and "change" variables are calculated as the monthly averages. Additionally, the dataset was checked for duplicate entries, and none were found. The data is divided into two sets: the test set, which includes observations from the last year, and the training set, which includes all previous data.

Lastly, since tree-based algorithms split data based on relative order of feature values rather than their absolute magnitudes and create decision boundaries and compute feature importance based on splits that are invariant to the scale of input features, scaling does not effect their performance and no need to scale the data. On the other hand, ARIMA models are sensitive to the scale of the data. Therefore, ARIMA models are built with scaled data while the tree-based algorithms are built with non-scaled data.

B. *Data Exploration and Exploratory Data Analysis*

B.1 Research Questions

For better understanding of the data, two research questions were generalized.

- What is the trend of Bitcoin prices (Open vs. Average Price) over time?

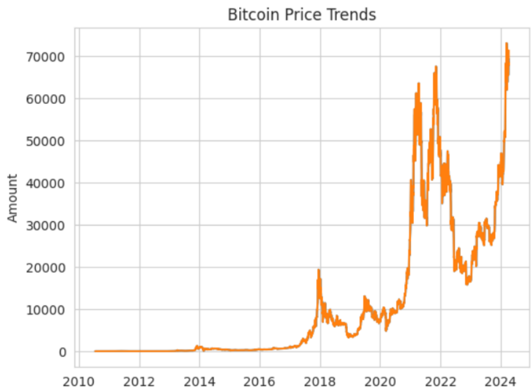

Figure 1: Autoplot of Bitcoin Prices

It is clear that although there are some strict up and downs in the bitcoin prices, there exists a positive increasing trend in the bitcoin prices.

- How does the percent change in Bitcoin price fluctuate over time?

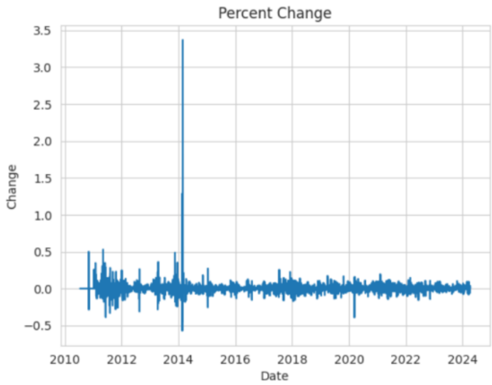

Figure 2: Plot of Bitcoin Price Changes

The percent change in Bitcoin price shows significant fluctuations over time, especially in the early years. From 2010 to around 2014, there are notable spikes and high volatility. The most prominent spike occurs around 2013-2014, where there is an exceptionally high percent change. After this period, from approximately 2015 onwards, the percent change appears to stabilize and becomes less volatile, although minor fluctuations are still present. This suggests that Bitcoin experienced extreme price changes in its early years but has become relatively more stable in recent years.

From the research question, it is implied that there exists an increasing trend in the bitcoin prices although there exists strict changes in the amount. However, these changes are being more stabilized since 2015 compared to the early years of Bitcoin.

B.2 Anomaly Detection

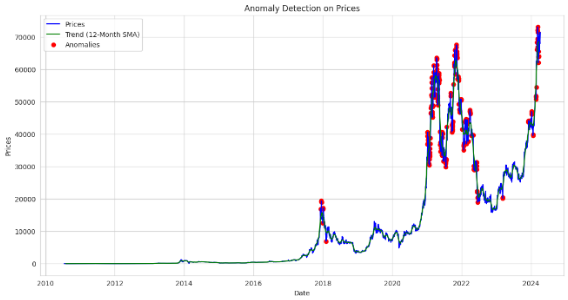

Figure 3: Anomaly Detection Plot

Anomaly detection on Bitcoin prices and residual component are applied and observed that the data has anomalies. With the related functions, anomalies are removed and replaced with interpolated values.

B.3 Feature Engineering

After converting the data set in the monthly time series format, month and year column is also added to the original data set in order to observe the seasonality and stationarity on the data better.

C. Time Series Plots and Interpretations

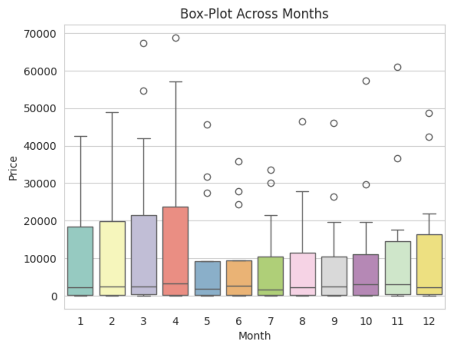

Figure 4: Boxplot across months

The median values for each month are close to each other and in the same range. Therefore, it can be said that there is no seasonality. In addition to the median values being close to each other, the box plots show significant variability in Bitcoin prices across different months. Some months exhibiting a wider range of prices than others. Other than indicating no seasonality, this also indicates that the overall price volatility can differ substantially from month to month. The presence of outliers in almost every month suggests that there have been several extreme price changes. These changes could be due to various market events or external factors affecting Bitcoin's price.

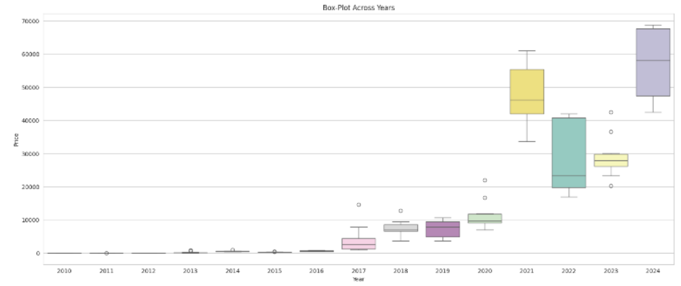

Figure 5: Boxplot across years

There exists up and down in the prices. If we compare the beginning and final, it can be said that there is an increasing trend. In addition to the observed upward trend from the beginning to the end of the period, the box plot reveals significant price volatility in the Bitcoin market, especially in recent years. The increasing size of the interquartile range from 2017 onwards indicates greater variability and higher price fluctuations. The presence of outliers highlights the occurrence of extreme price movements and market anomalies. This trend reflects the growing interest and speculative behavior in the Bitcoin market, contributing to its high volatility. Also, from the graph and autoplot in the B1.1, it can be observed that the series is not stationary.

ACF/PACF plots and related tests can be checked to be sure about if the series is stationary.

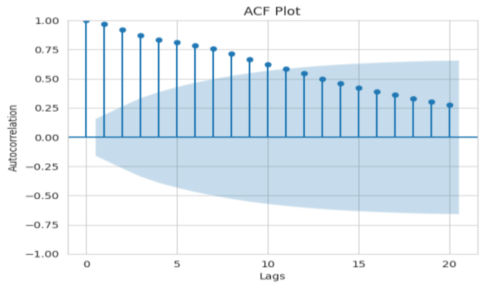

Figure 6: ACF Plot of Time Series

ACF/PACF plots show that there is slow linear decay in ACF plot which indicates that the series is not stationary. Then following tests are checked if there is stationarity in the series, if there is a unit root in the data set, if there is a unit root, how many difference should be taken.

- KPSS level test
- KPSS Trend Test
- PP Test
- ADF Test

All these tests suggest that the series is stationarity. In order to overcome the stationarity, one regular difference is taken. After taking the difference, the related tests and ACF/PACF plots show that there is no stationarity in the series. Also, autoplot seems stationary around mean 0.

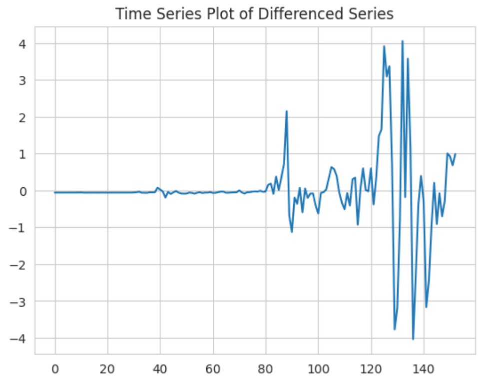

Figure 7: Autoplot of differenced series

D. Model Suggestions

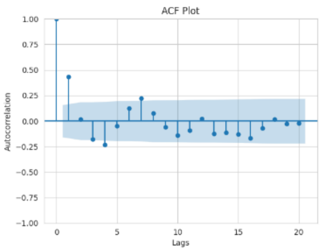

Figure 8: ACF Plot of Differenced Series

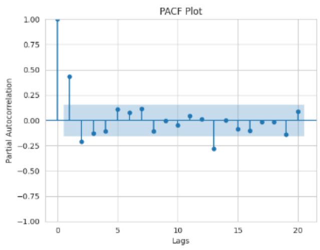

Figure 8: PACF plot of Differenced Series

From the ACF/PACF Plots, ARIMA(1,1,2) and ARIMA(4,1,2) models are decided. All the models that are going to be built are in the following:

- ARIMA(1,1,2)
- ARIMA(4,1,2)
- Decision Tree
- Random Forest
- Gradient Boosting
- AdaBoost
- CatBoost
- LightGBM
- Hybrid Models with the best 3.

E. Modelling

As it is implied in Data Description and Preprocessing part, the data was divided into two sets: the test set, which includes observations from the last year, and the training set, which includes all previous data.

1) ARIMA Models

Both ARIMA (1,1,2) and ARIMA (4,1,2) models are built. Before building it, diagnostic checking are applied to data set.

- Errors must follow a normal distribution with a mean of 0: Residuals should ideally follow a normal distribution. This can be visually assessed using a Q-Q plot and formally tested using methods like the Shapiro-Wilk and Jarque-Bera tests. Although the Q-Q plot may appear symmetric with an S-shape, outliers indicate that the residuals might not be normally distributed. Formal tests should be conducted to confirm non-normality. If the p-values for both the Shapiro-Wilk and Jarque-Bera tests are less than 0.05, it suggests that the residuals are not normally distributed, necessitating a transformation on the residuals.
- Errors must be independent from each other (White Noise): The presence of significant spikes outside the white noise band might indicate autocorrelation. Formal tests are needed to confirm whether the residuals are indeed normally distributed and independent.
- Error variance must be constant (Homoscedasticity): Significant spikes outside the white noise band might also indicate heteroscedasticity. Formal tests should be conducted to verify the presence of heteroscedasticity. If the p-values for the studentized Breusch-Pagan test and ARCH LM test are below the critical value of 0.05, it indicates a heteroscedasticity problem. This means the variance is not constant over time, and ARCH(lag) effects are present, requiring appropriate modeling.

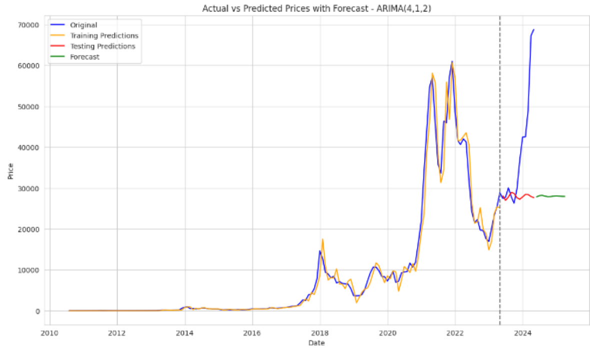

Figure 9: ARIMA(4,1,2) Plot

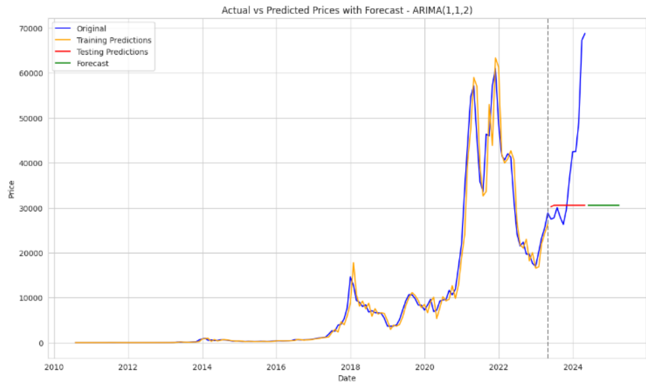

Figure 10 ARIMA (1,1,2) Plot

Both of the ARIMA models performed poorly for testing and training predictions.

2) Decision Tree

The target variable, Price, and features, Open, High, Low, Volume and Change, are separated. Hyperparameter tuning is applied using GridSearchCV in order to optimize the performance of the model. Best parameters are found as 1 min samples leaf, 2 min samples split and 20 max depth. The training predictions accurately replicate the original data patterns while there were some differences in the testing predictions.

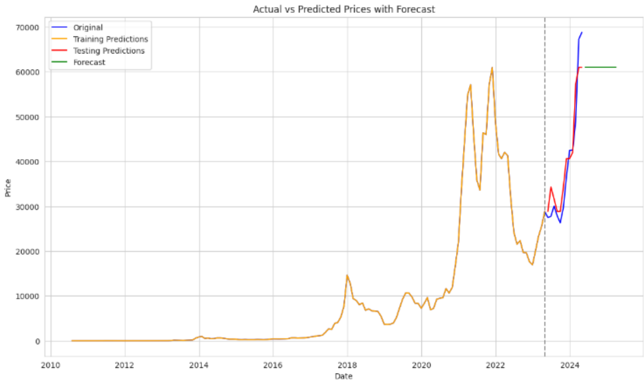

Figure 11: Decision Tree Forecasting

3) Random Forest

The target variable, Price, and features, Open, High, Low, Volume and Change, are separated. Hyperparameter tuning is applied. As distinct from Decision Tree parameters, best parameter for n_estimators is selected as 200 for Random Forest. Similar to the Decision Tree plot, the training predictions closely align with the original data patterns while there were some differences in the testing predictions. Comparing to the Decision Tree, testing predictions of Random Forest was better.

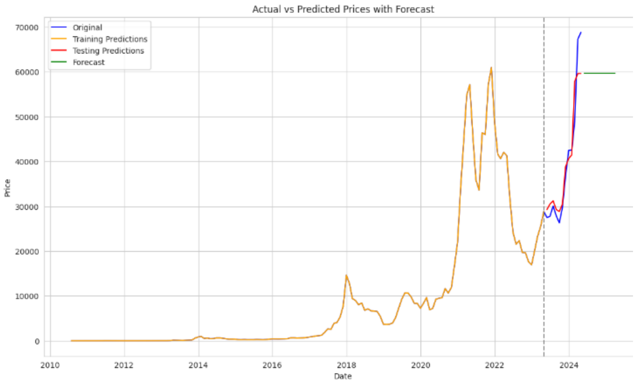

Figure 12: Random Forest Forecasting

4) Gradient Boosting

The target variable, Price, and features, Open, High, Low, Volume and Change, are separated. Hyperparameter tuning is applied. Best parameters are decided as following. For n_estimators, it is selected as 300. For min samples split, it is selected as 5. Max depth is selected as 7 and learning rate is decided as 0.1. Similar to the Decision Tree and Random Forest plot,

The training predictions effectively capture and reflect the original data patterns. For testing predictions, it is performed better than Decision Tree and Random Forest.

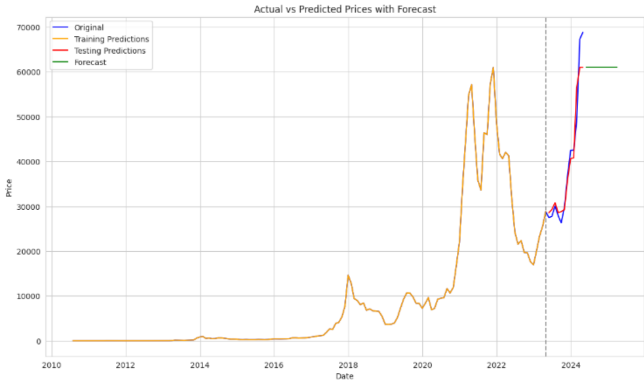

Figure 13 Gradient Boosting Forecasting

5) AdaBoost

The target variable, Price, and features, Open, High, Low, Volume and Change, are separated. Hyperparameter tuning is applied. Best values for parameters learning rate and n_estimators are decided as 0.01 and 200, respectively. However, the model performed poorly compared to the last 3 tree-based algorithms in terms of both training and testing predictions.

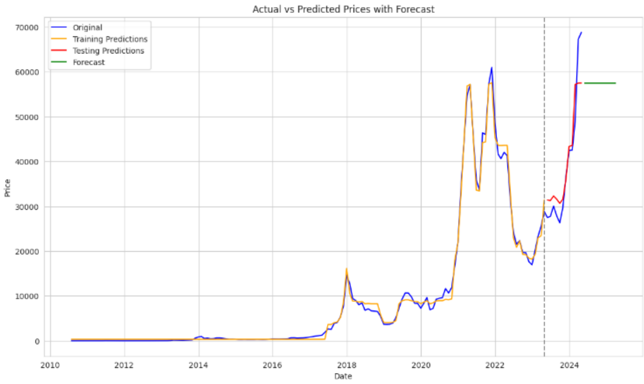

Figure 14: AdaBoost Forecasting

6) CatBoost

The target variable, Price, and features, Open, High, Low, Volume and Change, are separated. Hyperparameter tuning is applied. Best values for parameters learning rate, iterations, and depth are decided as 0.2, 200 and 4, respectively. The model performed better than Random Forest, Decision Tree and AdaBoost and performed worse than Gradient Boosting in terms of both training and testing predictions.

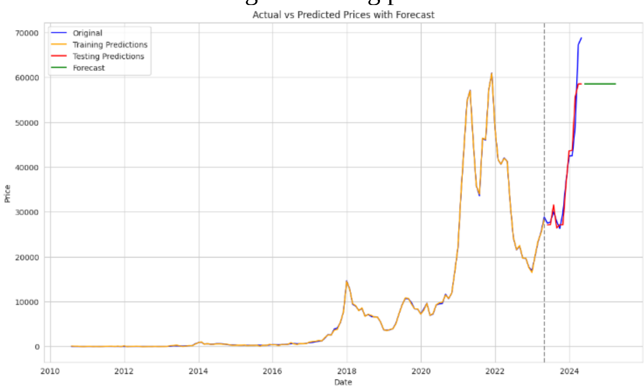

Figure 15: CatBoost Forecasting

7) *LightGBM*

The target variable, Price, and features, Open, High, Low, Volume and Change, are separated. Hyperparameter tuning is applied. Best values for parameters learning rate, number of leaves, and max depth are decided as 0.2, 31 and -1, respectively. Also, boosting type is selected as “gbdt”. The model performed the worst among all the models in terms of both training and testing predictions.

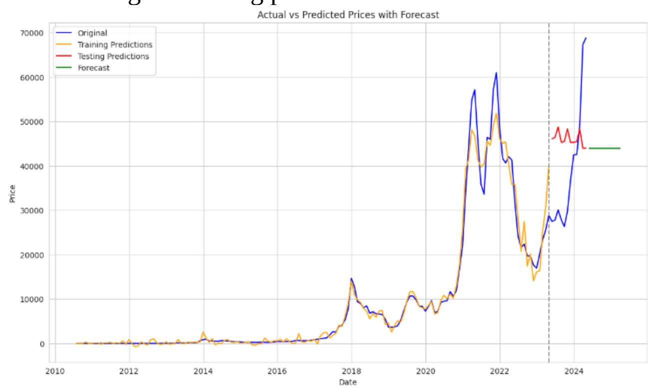

Figure 16: LightGBM Forecasting

F. *Hybrid Models*

Based on plots and Model Evaluation Criteria MAPE scores, the best three models are Gradient Boosting, CatBoost and Random Forest, respectively. After deciding the best models, hybrid models are applied as combination of them.

1) *Gradient Boosting - CatBoost Hybrid Model*

The target variable, Price, and features, Open, High, Low, Volume and Change, are separated. Hyperparameter tuning is applied in order to optimize the performance of the model. Best parameters are found as 64 border count, 300 iterations and 4 depth and 0.2 learning rate. The new hybrid model performs training predictions similar to the original case. For testing predictions, it performed better than CatBoost and worse than Gradient Boosting.

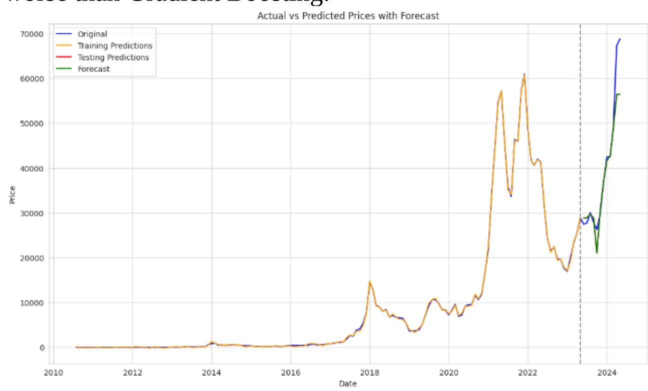

Figure 17: Gradient Boosting - CatBoost Hybrid Model

2) *Random Forest - Gradient Boosting Hybrid Model*

The target variable, Price, and features, Open, High, Low, Volume and Change, are separated. Hyperparameter tuning is applied in order to optimize the performance of the model. The new hybrid model performs training predictions similar to the original case. For testing predictions, it performed better than Random Forest and worse than Gradient Boosting.

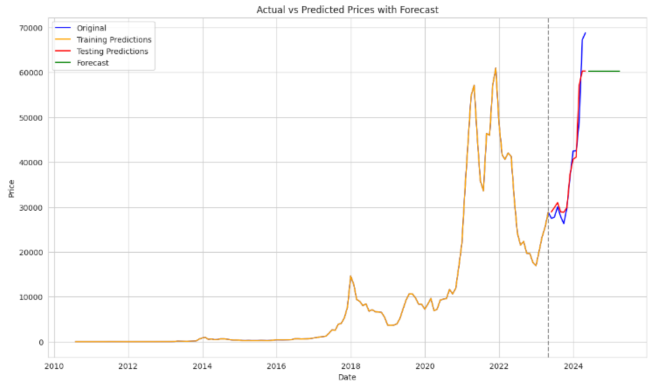

Figure 18: Random Forest - Gradient Boosting Hybrid Model

3) *Random Forest – CatBoost Hybrid Model*

The target variable, Price, and features, Open, High, Low, Volume and Change, are separated. Hyperparameter tuning is applied in order to optimize the performance of the model. The new hybrid model performs training predictions similar to the original case. For testing predictions, it performed better than each individual models, Random Forest and CatBoost. Among these 3 couple-hybrid models, it performs the best. However, Gradient Boosting is still better than the hybrid ones.

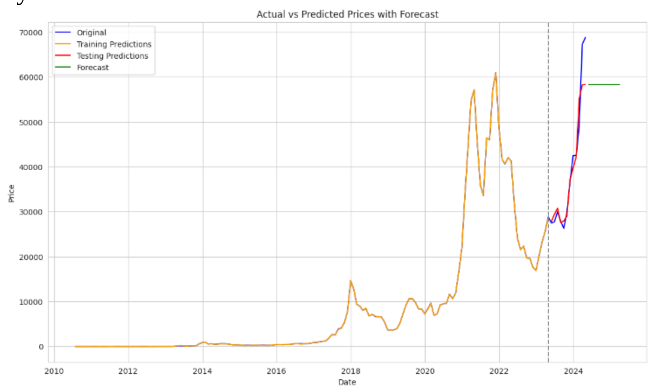

Figure 19: Random Forest – CatBoost Hybrid Model

4) *Gradient Boosting – Random Forest – CatBoost Triple-Hybrid Model*

Finally, the all the best 3 tree-based algorithms are combined, and hybrid model was created. It is performed grid search for each model (Random Forest, CatBoost, and Gradient Boosting) to find the best parameters. Each model is trained with the best parameters and made predictions. It is averaged the predictions from the three models to form the hybrid model. Finally, evaluation metrics (RMSE, MAPE, and MASE) are calculated for both training and testing data, and the date index is extended for the forecast period and make predictions for the forecast period. These metrics indicate that while the model fits the training data very well, testing predictions are worse than its training prediction reflecting the high volatility and unpredictability of Bitcoin prices. Test prediction of triple-hybrid model gives the best MAPE results among all the models.

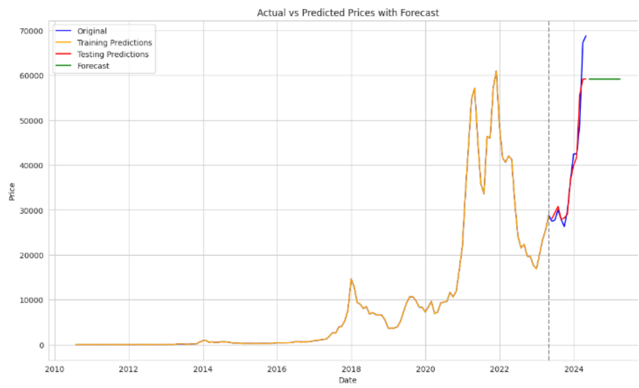

Figure 20: Triple-hybrid Model

IV. DISCUSSION

Mean Absolute Scaled Error are calculated for both train and test sets. Below are the MASE scores of individual models.

Table 1: MASE Values of Individual Models

Mean Absolute Scale Error		
Model	**Train**	**Test**
Gradient Boosting	0,01	2,019
CatBoost	0,06	2,19
Random Forest	0,01	2,50
Decision Tree	0,01	2,82
AdaBoost	0,47	3,12
ARIMA (1,1,2)	0,87	8,34
ARIMA (4,1,2)	0,86	9,02
LightGBM	0,85	10,61

Based on the low Mean Absolute Scaled Error (MASE) on both training and testing datasets, the table indicates that Gradient Boosting and CatBoost exhibit superior performance in forecasting Bitcoin prices. These models demonstrate excellent generalization and predictive accuracy. Conversely, Random Forest and Decision Tree models, while performing well on training data, show signs of overfitting with higher test errors. ARIMA models and LightGBM exhibit poor generalization, with significantly higher MASE on the test data, indicating challenges in accurately predicting unseen data. This underscores the robustness and effectiveness of ensemble methods over single models and ARIMA in capturing complex data patterns.

Below are the MASE scores of Hybrid Models.

Table 2: MASE Values of Hybrid Models

Mean Absolute Scale Error		
Model	**Train**	**Test**
Gradient Boosting - CatBoost	0,10	2,116
GradientBoosting - RF	0,01	2,176
Random Forest - CatBoost	0,02	2,110
Gradient Boosting - RF - CatBoost	0,01	2,016

Based on the Mean Absolute Scaled Error (MASE) values for the hybrid models indicates that the combination of Gradient Boosting, Random Forest, and CatBoost achieves the best performance on the test data, while maintaining a low training MASE of 0.01, demonstrating excellent generalization and predictive accuracy. The hybrid model combining Random Forest and CatBoost also performs well, with a test MASE of 2.110, followed by the Gradient Boosting and CatBoost combination at 2.116. The Gradient Boosting and Random Forest combination shows a slightly higher test MASE of 2.176, though it still performs well. These results underscore the effectiveness of hybrid models in improving predictive performance, with the Gradient Boosting - Random Forest - CatBoost combination being the most robust and accurate for forecasting Bitcoin price.

Lastly, every model forecast that the Bitcoin prices will remain stable for a while. The data has ended first week of the April. Since then, the Bitcoin prices seems stable. (Between the period 5th April-9th June, Bitcoin prices has changed 1,01%). Also, model implies cryptocurrency is not in a bull season. Latest economic news suggests that the bull season will start at the end of the 2024 or end of the 2025.

""Bull" and "bear" are typically used to describe how stock markets are performing — whether they are appreciating or depreciating in value. In this context, a rising market is called a bull market, while a declining one is called a bear market."

V. CONCLUSION

In this study, pattern of monthly Bitcoin prices are analyzed and next year is tried to be forecasted using Tree-Based algorithms. All the models are evaluated based on Mean Absolute Scale Error. Out of all 6 Tree-Based Algorithm and two ARIMA models, Gradient Boosting, Random Forest and CatBoost has given the best result as individual. Among these models, new hybrid models are created with their combinations. At the end of the study, there are 12 models builded on the dataset and out of these 12 models, best result is given by the triple hybrid model by Gradient Boosting, Random Forest and CatBoost. The study leverages the complementary strengths of these models to capture both temporal dependencies and complex patterns in Bitcoin price data. The creation of hybrid models, combining the best-performing algorithms, further enhances the predictive power, demonstrating superior accuracy over individual models. Moreover, the study provides insights into market trends. For example, all the models indicate that the cryptocurrency market is currently stable and not in a bull season. These results offer valuable guidance for investors and contribute to the broader understanding of cryptocurrency price dynamics, emphasizing the robustness and effectiveness of hybrid models in financial forecasting.

VI. REFERENCES

Al-Nefaie, A. H., & Aldhyani, T. H. H. (2022, December 8). Bitcoin price forecasting and trading: Data Analytics approaches. *MDPI*. https://www.mdpi.com/2079-9292/11/24/4088

Gupta, A., & Nain, H. (n.d.). Bitcoin Price Prediction Using Time Series Analysis and Machine Learning Techniques. *In Proceedings of the International Conference on Innovative Computing and Communications* (pp. 1-12). Springer. https://link.springer.com/content/pdf/bfm%3A978-981-15-7106-0%2F1.pdf

Murugesan, R., Shanmugaraja, V., & Vadivel, A. (2022, August 2). Forecasting bitcoin price using interval graph and Ann Model: A novel approach. *SN Computer Science*. SpringerLink. https://link.springer.com/article/10.1007/s42979-022-01291-x