**Cryptocurrency Forecasting**

**Using Statistical, Machine learning, AI models**

Nguyen L. T. Hien IS403.N22.HTCL\_Nhom7-EN  
University of Information Technology  
20521308@m.uit.edu.vn

Tran D. Khanh  
IS403.N22.HTCL\_Nhom7-EN  
University of Information Technology  
20521457@gm.uit.edu.vn

Dinh V. X. Hoan  
IS403.N22.HTCL\_Nhom7-EN  
University of Information Technology  
20521337@gm.uit.edu.vn

*Abstract: Cryptocurrencies, such as Bitcoin (BTC), Ethereum (ETH), and Binancecoin (BNB), have become popular investment options with their unpredictable price changes. This paper aims to forecast the direction of crypto prices in USD accurately. Various forecasting models, including Autoregressive Integrated Moving Average (ARIMA), Linear Regression (LN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Exponential Smoothing (ETS), Random Forest (RF), Gaussian Process Regression (GPR), Convolutional Neural Network (CNN), and Extreme Gradient Boosting (XGBoost), will be employed to predict the closing prices of BTC, ETH, and BNB. The evaluation will be based on RMSE, MAPE, and MAE. This study provides valuable insights for investors interested in the cryptocurrency market.*

*Keywords — Cryptocurrency Forecasting,* *ARIMA, Linear Regression, LSTM, GRU, ETS, Random Forest, GPR, CNN, XGBoost.*

# INtroduction

Predicting cryptocurrency prices has always been a formidable task in the realm of digital assets. Accurate forecasting of cryptocurrency prices holds great significance for investors seeking to make informed decisions. In recent years, the application of machine learning algorithms in cryptocurrency price prediction has garnered substantial attention. These algorithms possess the ability to autonomously learn from historical data and identify optimal predictive models based on patterns and variations. This empowers the generation of potential forecasts for cryptocurrency prices by leveraging both current and past information. Based on this premise, this research endeavors to develop a cryptocurrency price prediction model using diverse machine learning algorithms. Historical data encompassing cryptocurrency prices, along with relevant financial indicators and market factors, will be utilized to train the models. Subsequently, the models will be thoroughly evaluated and tested using new data to assess their accuracy and performance.

The study encompasses a range of machine learning algorithms, including ARIMA, Linear Regression, LSTM, GRU, ETS, Random Forest, GPR, CNN, XGBoost. Each algorithm brings forth distinctive capabilities and approaches to capture the underlying patterns and dynamics within cryptocurrency price movements. The research aims to explore the effectiveness of these algorithms in forecasting trends within the cryptocurrency market, providing valuable insights for investors and financial analysts. By harnessing historical data and advanced machine learning techniques, the research aims to enhance the precision and dependability of cryptocurrency price predictions, enabling informed decision-making in the ever-evolving world of digital asset markets.

# related works

Over the past few years, Bitcoin has been a topic of interest of many, from academic researchers to trade investors. Bitcoin is the first as well as the most popular cryptocurrency till date. Since its launch in 2009, it has become widely popular amongst various kinds of people for its trading system without the need of a third party and also due to high volatility of Bitcoin price. Therefore, Shaily Roy, Samiha Nanjiba and Amitabha Chakrabartypropose a suitable model that can predict the market price of Bitcoin best by applying a few statistical analyses. Their work is done on four year's bitcoin data from 2013 to 2017 based on time series approaches especially autoregressive integrated moving average (ARIMA) model and the work finally could acquire an accuracy of 90% for deciding volatility in weighted costs of bitcoin in the short run.[1]

In another article, Nicola Uras​​, Lodovica Marchesi​​, Michele Marchesi, Roberto Tonelli working on forecast daily closing price series of Bitcoin, Litecoin and Ethereum cryptocurrencies, using data on prices and volumes of prior days. They used the Simple Linear Regression (SLR) model for uni-variate series forecast using only closing prices, and the Multiple Linear Regression (MLR) model for multivariate series using both price and volume data. They used two artificial neural networks as well: Multilayer Perceptron (MLP) and long short-term memory (LSTM). In this case the best results are obtained using more than one previous price, thus confirming the existence of time regimes different from random walks. Their models perform well also in terms of time complexity, and provide overall results better than those obtained in the benchmark studies, improving the state-of-the-art.[2]

Another study is Andrés Oviedo-Gómez, Juan Manuel Candelo-Viáfara & Diego Fernando Manotas-Duque evaluate different crypto market variables through a quantile regression model and thus identify the best predictors for Bitcoin price forecasting by machine learning models. The main finding was that the Gaussian Process Regression models allowed the best performance metrics through the following predictors: high and low Bitcoin price, ask-sum, and Bitcoin price lagged. Likewise, the Bitcoin price was predicted for the next seven days, and it was observed a significant approximation by the confidence intervals of Gaussian Process Regression.[3]

Siripurapu Mounika, Podila Anjali Yadav, Tulluru Yashaswi, Chalimadugu Yamini Krishna, Dr. Vuyyuru Krishna Reddy also give us a view through the deep learning models such as Convolutional Neural Networks (CNN). Their aim of the work is to give accurate predictions and forecast and bring the daily trend for crypto currency market. Experimental results show that the proposed system given better accuracy on predictions.[4]

Zheshi Chen, Chunhong Li, Wenjun Sun compared with benchmark results for daily price prediction, they achieve a better performance, with the highest accuracies of the statistical methods and machine learning algorithms of 66% and 65.3%, respectively. Machine learning models including Random Forest, XGBoost, Quadratic Discriminant Analysis, Support Vector Machine and Long Short-term Memory for Bitcoin 5-minute interval price prediction are superior to statistical methods, with accuracy reaching 67.2%. Their investigation of Bitcoin price prediction can be considered a pilot study of the importance of the sample dimension in machine learning techniques.[5]

V. Derbentsev, V. Babenko, K. Khrustalev, H. Obruch, S. Khrustalova give another approach of using Random Forests (RF) and Stochastic Gradient Boosting Machine (SGBM) Their results verify the applicability of the ML ensembles approach for the forecasting of cryptocurrency prices. The out of sample accuracy of short-term prediction daily close prices obtained by the SGBM and RF in terms of Mean Absolut Percentage Error (MAPE) for the three most capitalized cryptocurrencies (BTC, ETH, and XRP) were within 0.92-2.61 %.[6]

# DATA

a) Data sources

This study schange daily historical data of three popular cryptos which are Bitcoin, Ethereum, and BNB from May 18, 2018, to May 18, 2023, such as High, Low, Open, Price, Volume, and Change where:

• High: This represents the highest price that a cryptocurrency reaches during the day.

• Low: This represents the lowest price that a cryptocurrency reaches during the day.

• Open: This represents the price of a cryptocurrency at the beginning of the day

• Price: This represents the current price of a cryptocurrency. By the end of the day, this will be known as the closing price.

• Volume: This represents the total number of units of a cryptocurrency that were traded during the day.

• Change: This represents the percentage change in the price of a cryptocurrency compared to its previous price. A positive change shows a price increase, while a negative change shows a price decrease.

b) Descriptive Statistics

A picture containing text, screenshot, font, number

Description automatically generated

*Table 1. Descriptive statistics of BTC.*

A picture containing text, screenshot, font, number

Description automatically generated

*Table 2. Descriptive statistics of ETH.*

*A picture containing text, screenshot, font, number

Description automatically generated*

*Table 3. Descriptive statistics of BNB.*

c) Visualization

1) BTC

A graph showing the price of bitcoin

Description automatically generated with medium confidence

*Figure 1. Visualization of BTC close price.*

2) ETH

A picture containing text, screenshot, plot, line

Description automatically generated

*Figure 2. Visualization of ETH close price.*

3) BNB

A picture containing screenshot, plot, text, line

Description automatically generated

*Figure 3. Visualization of BNB close price.*

# methodlogy

a) ARIMA

The ARIMA model is derived by general modification of an autoregressive moving average ARMA model. This model type is classified as ARIMA(p,d,q), where p denotes the autoregressive parts of the data set, d refers to integrated parts of the data set and q denotes moving average parts of the data set and p,d,q is all nonnegative integers

ARIMA model includes AR(p), I(d), MA(q):

An autoregressive model of order p can be written as:

Where  is the current value, is the constant term, p is the number of orders; is the autoregressive coefficient and  is the error.

A moving average model of order q can be written as:

It involves taking differences between the time series values to make the series stationary. Differencing is a method for removing trends and seasonality from a time series, which can make it easier to model.

Last, the I part is Integrated, and d is the number of

differences (order) required to make it a stationary sequence.

After combining them, we will have the ARIMA (p, d, q) express as follow:

The first step to apply ARIMA model is identification of the

time series. An Augmented Dicky–Fuller (adf) unit-root test

shows if the dataset is stationary or not. If unit root exists, then the time series is non-stationary. If the time series is found to be stationary, then we can use the ARMA model to estimate and forecast. But if it is not stationary then to apply ARIMA it has to be converted into stationary by differencing. After identification, ARIMA models are estimated for the specific stationary time series.

b) LN

Regression analysis is a tool for building statistical models that characterize relationships among a dependent variable and one or more independent variables, all of which are numerical.

There are Simple Linear Regression and Multiple Linear Regression:

Simple Linear Regression estimates the relationship between a scalar response y and a single explanatory variable x (also called dependent variable y and independent variable 𝑥), given a set of data that includes observations for both variables for a particular sample.

Multiple Linear Regression is a generalization of simple linear regression in which is used to estimate the relationship between two or more independent variables and one dependent variable. The Multivariable Linear Regression formula:

*yi*​ = *β*0​ + *β*1​*xi*​+ *β*2​*xi*2​+...+ *βp*​*xip*​+ *ϵ*[8]

where, for i=n observations:

*yi*​ is the dependent or predicted variable.

*β*0 is the y-intercept, i.e., the value of y when both *xi* and *x*2 are 0.

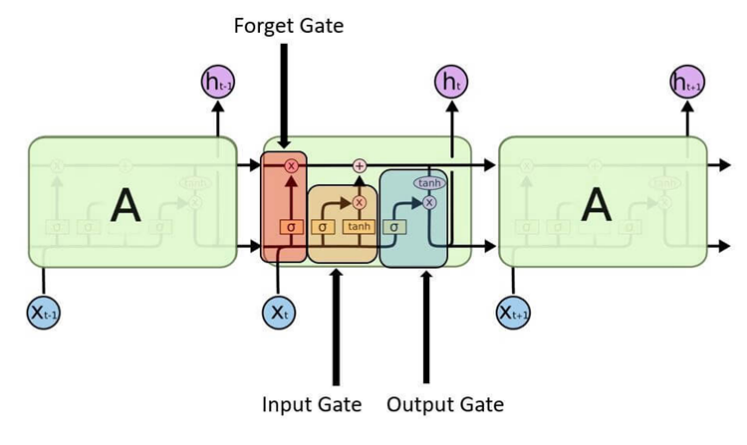
*β*1 and *β*2​ are the regression coefficients representing the change in *y* relative to a one-unit change in ​ *xi*1 and ​ *xi*2, respectively.

*βp*​ is the slope coefficient for each independent variable.

**ϵ** is the model’s random error (residual) term.

c) LSTM

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) that is specifically designed to handle sequential data, such as time series, speech, and text. LSTM networks are capable of learning long-term dependencies in sequential data, which makes them well suited for tasks such as language translation, speech recognition, and time series forecasting.[9]



*Figure 4. Structure of an LSTM*[10]

The structure of an LSTM is shown in the figure above. It consists of three interacting gates (input gate, output gate and forget gate) and numerous memory blocks known as cells, which store important information during the processing of sequential data.

The input gate controls what information is added to the memory cell. The forget gate controls what information is removed from the memory cell. And the output gate controls what information is output from the memory cell. This allows LSTM networks to selectively retain or discard information as it flows through the network, which allows them to learn long-term dependencies.

Forget gate:

Input gate:

Output gate:

d) GRU

Gated recurrent units (GRUs) are a gating mechanism in recurrent neural networks, introduced in 2014 by Kyunghyun Cho et al. The GRU is like a long short-term memory (LSTM) with a forget gate, but has fewer parameters than LSTM, as it lacks an output gate.

The GRU model is a modified version of the LSTM model, it not only merges the forget gate and the input gate into an update gate but also drops the cell state, achieving reduction of number of parameters.

In the first step, reset gate is calculated using both the hidden state from the previous time step and the input data at the current time step, it be reserved by applying a sigmoid function

Where:

: is input data at the current time step.

: is the hidden state from the previous time step.

,: are the weighting vectors respectively.

Next step, decide the information which will be kept from the previous time steps together with the new inputs following.

Second, the update gate is computed using the previous hidden state and current input data using the same formula, like the reset gate. But each weight multiplied with the input and hidden state is independent and unique to each gate, which means the final vectors for the update gate are different from the reset gate.

Finally, summarize the output.

e) ETS

The Exponential Smoothing (ETS) is a time series forecasting technique that makes future predictions based on historical data. Exponential smoothing was proposed in the late 1950s (Brown, 1959; Holt, 1957; Winters, 1960), and has motivated some of the most successful forecasting methods.

The simpliest model is called the simple exponential smoothing (SES) model. This forecasting method is the most widely used of all forecasting techniques. This method is suitable for forecasting data with no clear trend or seasonal pattern.

Simple Exponential Smoothing equation:

: forecast value at time t + 1.

: forecast value at time t.

: actual value at the time t.

: smoothing parameter, between 0 and 1

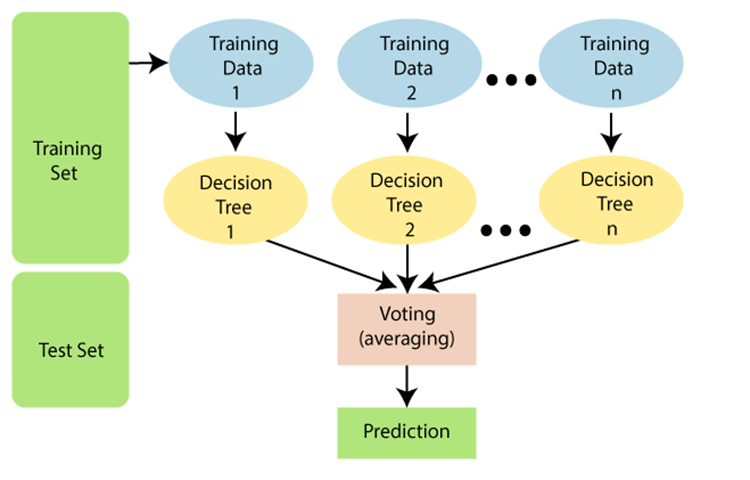
For any α between 0 and 1, the weights attached to the observations decrease exponentially as we go back in time.

Small α value (close to 0): results in more smooth and slower response to changes in the time series.

Large α value (close to 1): results less smoothing and a faster response to changes in the time series.

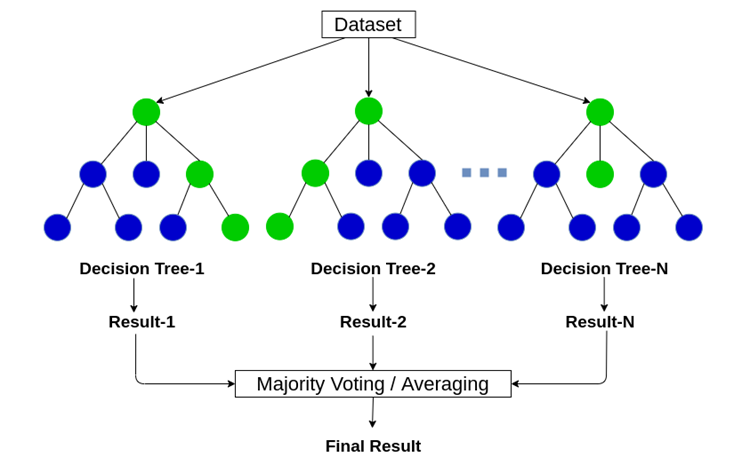
f) RF

Random Forest is one of the most popular and commonly used algorithms by Data Scientists. Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.



*Figure 5. Random Forest Algorithm*[13]

Random Forest uses an ensemble technique called Boostrap Aggregation or also known as Bagging. Bagging, it chooses a random sample and creates a different training subset from training data, the final ouput is based on majority voting.



*Figure 6. Majority Voting/ Averaging* [14]

g) GPR

Gaussian process regression (GPR) is a nonparametric, Bayesian approach to regression that is making waves in machine learning. GPR has several benefits, working well on small datasets and having the ability to provide uncertainty measurements on predictions. An important aspect in implementing the GPR model is to search for a suitable kernel function that can accurately predict the output data and its corresponding covariance. In this case, we have chosen the Rational Quadratic kernel.

[15]

Where: = [, . . ., ]: It represents the input data points, where , is the i-th data point, represents the Euclidean distance between two input data points. the variance, the lengthscale, the scale-mixture ( > 0). The predictive equations for Gaussian processes regression: the prediction for the test data can be calculated as:

[16]

Prediction the uncertainty (variance) of the predictions can be calculated as:

[16]

Where: *“”* is identity matrix, variance depend on the inputs and .

h) CNN

A convolutional neural network (CNN) is essentially a neural network that employs the convolution operation (instead of a fully connected layer) as one of its layers. CNNs are an incredibly successful technology that has been applied to problems where in the input data on which predictions are to be made has a known grid-like topology, like a time series (a 1-D grid) or an image (a 2-D grid). CNNs ushered deep learning into modern times, solving one of the most crucial computational problems in the digital era of computer vision. With the popularity of CNNs, a surge in the research for deep learning was witnessed that continues today.

CNN is made up of three primary layers a convolution layer, a pooling layer, and a fully connected layer. The convolution layer makes an effort to retrieve the best features from the 1-D matrix and perform calculation to provide a convoluted output, as shown in equation below.

[17]

Where is the convolution output, activation function is , is input value, is the weight, and is the bias.

A diagram of convolution structure

Description automatically generated with medium confidence

Figure 7. CNN one-dimensional structure.[17]

A The pooling layer takes the output of the convolutions as an input. The max pooling function is used to choose the heavily weighted features in the pooling layer. The pooling layer’s output is passed to the flattening layer. The flatten layer’s primary function is to convert the data into a single array form. The fully connected layer receives the flattening layer’s output and processes it to obtain the results.

i) XGBoost

XGBoost is a powerful machine learning library called Extreme Gradient Boosting. It helps us build models that can handle large amounts of data and be distributed across multiple computers. It uses a technique called gradient boosting with decision trees, which is great for solving problems related to predicting values or classifying things.

Gradient Boosting Decision Trees (GBDT) is a type of algorithm that combines multiple decision trees to make better predictions. It's similar to another algorithm called random forest, but with some differences. Random forest focuses on reducing variability and overfitting, while GBDT focuses on reducing bias and underfitting. By combining multiple machine learning algorithms, ensemble learning algorithms like GBDT can create more accurate models.

A picture containing diagram, sketch, pattern

Description automatically generated

*Figure 8. Boosting* [18]

XGBoost is used for supervised learning problems, where we use the training data (with multiple features) to predict a target variable .

Objective: [19]

Where:

: training loss measures well model fit on training data.

: Regularization, measures complexity of trees, K is the number of trees.

Following additive training to learn tree ensembles, add a new function each time at a result, we will have:

: model at training round t

: new function at round t

Atfer all we will have the prediction at round t:

# Experiment

a) Dataset spliting

The dataset is divided into training, test, and validate sets based on three ratios include 70-20-10%, 60-30-10%, and 50-30-20%. The training set is used to create the model, and its performance is evaluated using the test set and validate set. To improve the dataset quality, certain preprocessing techniques are applied, including data cleansing, feature selection, data reduction, and data transformation.

b) Evaluation

In this research, predictive models are evaluated according to three criteria: MAPE, RMSE, and MAE.

In the following formulas:

n is the number of observations.

*Xi* element is the predicted *ith* value.

*Yi* element is the actual *ith* value.

- Mean Absolute Percentage Error – MAPE

[20]

(Best value = 0; worst value = +∞)

- Root Mean Squared Error - RMSE

[20]

(Best value = 0; worst value = +∞)

- Mean Absolute Error - MAE

[20]

(Best value = 0; worst value = +*∞*)

c) Result

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| BTC dataset | | | | |
| Model | Proportion | RMSE | MAPE | MAE |
| ARIMA | 7:2:1 | 33455.97 | 124.35% | 31265.1 |
| 6:3:1 | 14242.86 | 43.55% | 11506.54 |
| 5:3:2 | 29620.21 | 59.91% | 27435.14 |
| LN | 7:2:1 | 22897.28 | 81.99% | 18932.26 |
| 6:3:1 | 16692.12 | 48.81% | 13681.81 |
| 5:3:2 | 32634.15 | 68.53% | 30731.53 |
| GRU | 7:2:1 | 15771.27 | 37.88% | 12358.92 |
| 6:3:1 | 18573.53 | 53.76% | 14843.68 |
| 5:3:2 | 20271.5 | 69.02% | 16796.31 |
| LSTM | 7:2:1 | 15542.96 | 38.14% | 12226.16 |
| 6:3:1 | 18477.78 | 49.09% | 15037.39 |
| 5:3:2 | 14374.15 | 31.38% | 11509.19 |
| ETS | 7:2:1 | 57411.04 | 212.6% | 52285.84 |
| 6:3:1 | 12365.06 | 36.91% | 10215.43 |
| 5:3:2 | 16025.81 | 27.82% | 12871.72 |
| RF | 7:2:1 | 13225.67 | 39.28% | 11294.10 |
| 6:3:1 | 17393.72 | 52.70% | 14232.47 |
| 5:3:2 | 30060.92 | 61.10% | 27910.38 |
| GPR | 7:2:1 | 21679.50 | 53.85% | 18753.44 |
| 6:3:1 | 12065.66 | 22.83% | 9463.12 |
| 5:3:2 | 35914.3 | 75.3% | 33888.61 |
| CNN | 7:2:1 | 16549.7 | 44.77% | 13159.92 |
| 6:3:1 | 18791.54 | 47.92% | 15113.7 |
| 5:3:2 | 15535.77 | 33.36% | 12402.7 |
| XGBOOST | 7:2:1 | 12911.91 | 42.27% | 11293.64 |
| 6:3:1 | 16995.69 | 52.83% | 13845.5 |
| 5:3:2 | 29838.67 | 60.5% | 27670.86 |

*Table 4. Evaluation of BTC dataset.*

Based on the table we conclude that the best model for forecasting the next 30 days BTC closing price is GPR model with the proportion 6:3:1 because it has the lowest RMSE, MAPE, MAE value.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ETH dataset | | | | |
| Model | Proportion | RMSE | MAPE | MAE |
| ARIMA | 7:2:1 | 4461.28 | 225.95% | 3673 |
| 6:3:1 | 6041.72 | 284.09% | 5115.96 |
| 5:3:2 | 3490.67 | 112.06% | 3151.55 |
| LN | 7:2:1 | 1237.47 | 60.33% | 1089.10 |
| 6:3:1 | 1628.63 | 44.14% | 1318.19 |
| 5:3:2 | 2621.62 | 88.29% | 2398.75 |
| GRU | 7:2:1 | 1273.83 | 42.98% | 998.50% |
| 6:3:1 | 1454.17 | 54.58 % | 1181.39 |
| 5:3:2 | 1356.16 | 72.27 % | 1087.92 |
| LSTM | 7:2:1 | 1262.48 | 43.47% | 992.68 |
| 6:3:1 | 1594.2 | 60.62% | 1290.33 |
| 5:3:2 | 1594.14 | 86.73 % | 1273.48 |
| ETS | 7:2:1 | 5027.52 | 270.74% | 4501.69 |
| 6:3:1 | 6711.44 | 315.20% | 5702.24 |
| 5:3:2 | 2098.94 | 66.13% | 1874.47 |
| RF | 7:2:1 | 1218.92 | 42.55% | 959.04 |
| 6:3:1 | 1287.35 | 53.64% | 1064.96 |
| 5:3:2 | 2411.31 | 76.58% | 2168.04 |
| GPR | 7:2:1 | 608.29 | 29.34% | 538.93 |
| 6:3:1 | 1545.42 | 40.69% | 1223 |
| 5:3:2 | 2564.53 | 82.57% | 2319.88 |
| CNN | 7:2:1 | 1369.90 | 53.22% | 1091.27 |
| 6:3:1 | 1464.30 | 54.68% | 1185.80 |
| 5:3:2 | 1481.38 | 71.42% | 1194.30 |
| XGBOOST | 7:2:1 | 1220.98 | 43.05% | 965.22 |
| 6:3:1 | 1458.15 | 68.27% | 1229.26 |
| 5:3:2 | 2421.84 | 77.18% | 2179.75 |

*Table 5. Evaluation of ETH dataset.*

Based on the table we conclude that the best model for forecasting the next 30 days ETH closing price is GPR model with the proportion 7-2-1 because it has the lowest RMSE, MAPE, MAE value.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| BNB dataset | | | | |
| Model | Proportion | RMSE | MAPE | MAE |
| ARIMA | 7:2:1 | 288.83 | 88,22% | 269.59 |
| 6:3:1 | 942.54 | 258.59% | 828.23 |
| 5:3:2 | 369.66 | 81.82% | 327.65 |
| LN | 7:2:1 | 133.00 | 33.12% | 114.95 |
| 6:3:1 | 211.59 | 42.16% | 176.07 |
| 5:3:2 | 368.89 | 83.07% | 327.50 |
| GRU | 7:2:1 | 128.05 | 25.74% | 98.03 |
| 6:3:1 | 147.42 | 33.12% | 117.29 |
| 5:3:2 | 208.01 | 192.51% | 161.53 |
| LSTM | 7:2:1 | 126.20 | 25.74% | 97.06 |
| 6:3:1 | 150.63 | 37.10% | 123.34 |
| 5:3:2 | 186.34 | 160.02% | 149.22 |
| ETS | 7:2:1 | 512.50 | 153.89% | 466.83 |
| 6:3:1 | 979.72 | 268.22% | 857.20 |
| 5:3:2 | 334.26 | 73.26% | 293.85 |
| RF | 7:2:1 | 116.61 | 23.94% | 89.60 |
| 6:3:1 | 229.90 | 66.22% | 208.26 |
| 5:3:2 | 361.85 | 79.23% | 319.76 |
| GPR | 7:2:1 | 67.45 | 16.20% | 52.35 |
| 6:3:1 | 240.50 | 56.78% | 214.19 |
| 5:3:2 | 377.58 | 84.88% | 335.53 |
| CNN | 7:2:1 | 145.21 | 32.80% | 114.71 |
| 6:3:1 | 149.71 | 33.21% | 118.39 |
| 5:3:2 | 240.95 | 178.60% | 188.49 |
| XGBOOST | 7:2:1 | 115.87 | 23.80% | 89.11 |
| 6:3:1 | 239.82 | 69.61% | 219.34 |
| 5:3:2 | 362.13 | 79.28% | 320.03 |

*Table 6. Evaluation of BNB dataset.*

Based on the table we conclude that the best model for forecasting the next 30 days BNB closing price is GPR model with the proportion 7-2-1 because it has the lowest RMSE, MAPE, MAE value.

f) Visualize

Visualizing the predicted values and the actual values of the GPR model and the next 30 days forecasting values.

A picture containing text, screenshot, plot, diagram

Description automatically generated*Figure 9. Predictions of the GPR model with BTC dataset and rate of 6:3:1*

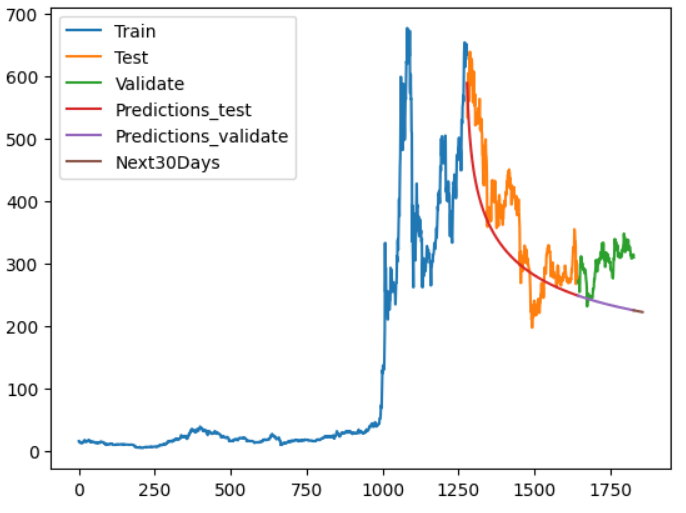
Visualizing the predicted values and the actual values of GPR model on ETH dataset and the next 30 days forecasting values.

A graph with blue and orange lines

Description automatically generated with low confidence

*Figure 10. Predictions of the GPR model with ETH dataset and rate of 7:2:1*

Visualizing the predicted values and the actual values of GPR model on BNB dataset and the next 30 days forecasting values.



*Figure 11. Predictions of the GPR model with BNB dataset and rate of 7:2:1*

# Conclusion

In conclusion, this study compared the performance of various models in predicting the future prices of BTC, ETH, and BNB cryptocurrencies based on the resulting time series. Among the models tested, the Gaussian Process Regression (GPR) model emerged as the most suitable for this task. Conversely, the Linear Regression, ARIMA, LSTM, GRU, Random Forest, ETS, and XGBOOT models did not exhibit comparable performance. These findings underscore the significance of employing diverse modeling approaches in financial analysis and highlight the potential value of utilizing the GPR model for forecasting cryptocurrency prices in the future. Further research is warranted to validate these results and explore the performance of alternative models in different types of cryptocurrency price prediction tasks.

ACKNOWLEDGMENT

We would like to extend our sincere thanks to ***Associate Professor Dr. Nguyen Dinh Thuan, Mr. Nguyen Minh Nhut, and Ms. Nguyen Thi Viet Huong***. Your constant care and support throughout this course have been greatly appreciated by our team. We value your presence and are genuinely grateful for your invaluable assistance and the sharing of your extensive experience. We believe that your guidance will undoubtedly contribute to our project's success.

##### References

[1] S. Roy, S. Nanjiba, and A. Chakrabarty, “Bitcoin Price Forecasting Using Time Series Analysis,” in *2018 21st International Conference of Computer and Information Technology (ICCIT)*, Dec. 2018, pp. 1–5. doi: 10.1109/ICCITECHN.2018.8631923.

[2] N. Uras, L. Marchesi, M. Marchesi, and R. Tonelli, “Forecasting Bitcoin closing price series using linear regression and neural networks models,” *PeerJ Comput. Sci.*, vol. 6, p. e279, Jul. 2020, doi: 10.7717/peerj-cs.279.

[3] A. Oviedo-Gómez, J. M. Candelo-Viáfara, and D. F. Manotas-Duque, “Bitcoin Price Forecasting Through Crypto Market Variables: Quantile Regression and Machine Learning Approaches,” in *Handbook on Decision Making: Volume 3: Trends and Challenges in Intelligent Decision Support Systems*, J. A. Zapata-Cortes, C. Sánchez-Ramírez, G. Alor-Hernández, and J. L. García-Alcaraz, Eds., in Intelligent Systems Reference Library. Cham: Springer International Publishing, 2023, pp. 253–271. doi: 10.1007/978-3-031-08246-7\_11.

[4] S. Mounika, “Crypto-Currency Price Prediction using CNN and LSTM Models,” *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, no. 3, pp. 107–114, Mar. 2021, doi: 10.22214/ijraset.2021.33191.

[5] Z. Chen, C. Li, and W. Sun, “Bitcoin price prediction using machine learning: An approach to sample dimension engineering,” *J. Comput. Appl. Math.*, vol. 365, p. 112395, Feb. 2020, doi: 10.1016/j.cam.2019.112395.

[6] V. Derbentsev, V. Babenko, K. Khrustalev, H. Obruch, and S. Khrustalova, “Comparative Performance of Machine Learning Ensemble Algorithms for Forecasting Cryptocurrency Prices,” *Int. J. Eng.*, vol. 34, no. 1, pp. 140–148, Jan. 2021, doi: 10.5829/ije.2021.34.01a.16.

[7] N. Abu Bakar and S. Rosbi, “Autoregressive Integrated Moving Average (ARIMA) Model for Forecasting Cryptocurrency Exchange Rate in High Volatility Environment: A New Insight of Bitcoin Transaction,” *Int. J. Adv. Eng. Res. Scinece*, vol. 4, Nov. 2017, doi: 10.22161/ijaers.4.11.20.

[8] “Multiple Linear Regression,” *Corporate Finance Institute*. https://corporatefinanceinstitute.com/resources/data-science/multiple-linear-regression/ (accessed Jun. 21, 2023).

[9] “Deep Learning | Introduction to Long Short Term Memory,” *GeeksforGeeks*, Jan. 16, 2019. https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/ (accessed Jun. 21, 2023).

[10] D. Kalita, “An Overview on Long Short Term Memory (LSTM),” *Analytics Vidhya*, Mar. 11, 2022. https://www.analyticsvidhya.com/blog/2022/03/an-overview-on-long-short-term-memory-lstm/ (accessed Jun. 21, 2023).

[11] C.-J. Chen, F.-I. Chou, and J.-H. Chou, “Temperature Prediction for Reheating Furnace by Gated Recurrent Unit Approach,” *IEEE Access*, vol. 10, pp. 33362–33369, 2022, doi: 10.1109/ACCESS.2022.3162424.

[12] *Chapter 8 Exponential smoothing | Forecasting: Principles and Practice (3rd ed)*. Accessed: Jun. 19, 2023. [Online]. Available: <https://otexts.com/fpp3/expsmooth.html>

[13] “Machine Learning Random Forest Algorithm - Javatpoint.” https://www.javatpoint.com/machine-learning-random-forest-algorithm (accessed Jun. 21, 2023).

[14] C. GOYAL, “Bagging- 25 Questions to Test Your Skills on Random Forest Algorithm,” *Analytics Vidhya*, May 31, 2021. https://www.analyticsvidhya.com/blog/2021/05/bagging-25-questions-to-test-your-skills-on-random-forest-algorithm/ (accessed Jun. 21, 2023).

[15] “Gaussian processes (3/3) - exploring kernels.” https://peterroelants.github.io/posts/gaussian-process-kernels/ (accessed Jun. 21, 2023).

[16] J. Wang, “An Intuitive Tutorial to Gaussian Processes Regression.” arXiv, Apr. 18, 2022. Accessed: Jun. 21, 2023. [Online]. Available: http://arxiv.org/abs/2009.10862

[17] S. Zaheer, N. Anjum, A. Algarni, J. Iqbal, S. Bourouis, and S. S. Ullah, “A Multi Parameter Forecasting for Stock Time Series Data Using LSTM and Deep Learning Model,” *Mathematics*, vol. 11, p. 590, Jan. 2023, doi: 10.3390/math11030590.

[18] “Flow diagram of gradient boosting machine learning method. The ensemble...,” *ResearchGate*. https://www.researchgate.net/figure/Flow-diagram-of-gradient-boosting-machine-learning-method-The-ensemble-classifiers\_fig1\_351542039 (accessed Jun. 21, 2023).

[19] “Introduction to Boosted Trees — xgboost 1.7.6 documentation.” https://xgboost.readthedocs.io/en/stable/tutorials/model.html (accessed Jun. 21, 2023).

[20] D. Chicco, M. J. Warrens, and G. Jurman, “The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation,” *PeerJ Comput. Sci.*, vol. 7, p. e623, Jul. 2021, doi: 10.7717/peerj-cs.623.