TREND ANALYSIS AND PREDICTION OF ETHEREUM PRICES USING MACHINE LEARNING APPROACHES

AUGUST 2022

TABLE OF CONTENTS

This page is intentionally left blank

ABSTRACT

A remarkable technology named Ethereum (ET) is growing in the market that allows several global payments and applications. This technology came into existence with the goal of making it easier to build projects on its platform. Ethereum is a blockchain (BC) based software platform that provides a peer- to- peer network and runs securely to verify the application code known as smart contracts. With the rise in the popularity of Ethereum, its price began to compete with all other Cryptocurrencies which made it the most valuable of them thus occupying the place of the second largest Cryptocurrency (CC) in the world. Therefore, the goal of the Cryptocurrency study is to establish a clear picture and forecast its value based on the increase in the coin's historical price. In this study, price prediction is carried out utilizing a stage graph of typical ether Cryptocurrency closure costs and three machine learning techniques (ML) named LR (Linear Regression), Decision trees (DT), and Exponential smoothing (ES). These three machine learning algorithms are adopted to predict the accuracy of trend analysis. One of these models is the Decision Tree algorithm, which offers various advantages but is less reliable in comparison to the LR model. Additionally, the error of the suggested model has been lowered by implementing certain time series models, and further; more models can be implemented to enhance the accuracy of the system and to perform accurate trend analysis and the prediction of Ethereum prices more robustly with high performance.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Machine Learning……..ML

Linear Regression……..LR

Decision Tree………….DT

Artificial Intelligence….AI

Artificial Neural Network…ANN

Support Vector Machine…..SVM

Bitcoin…………………BT

Simple Moving Average….SMA

Exponential moving average….EMA

Long Short-term Memory…..LSTM

Mean Square Error………….MSE

Root Mean Square Error……RMSE

Normalize Root Mean Square Error....NRMSE

k- Nearest Neighbor……….KNN

Multiple Polynomial Regression……. MPR

Receiver Operating Curve... ROC

Multilayer Perceptron……. MLP

Binary Autoregressive Tree model…. BART

Random Forest….RF

Bi- Long Short-term Memory….Bi-LSTM

Stacked Long Short-term Memory...s-LSTM

Gated Recurrent Unit.…GRU

Convolution Neural Network ….CNN

Recurrent Neural Networks ….RNN

Exponential Smoothing…..ES

Regression Analysis….RA

Arithmetic Moving Average ….AMA

Exponentially Weighted Moving Average….EWMA

Standard Normal Distribution.....SND

Attribute Selection Measure ......ASM

Information Gain……IG

Gini Index……GI

Classification and Regression Tree……CART

Simple Linear Regression…..SLR

Multi Linear Regression MLR

Single Exponential Smoothing......SES

Double Exponential Smoothing......DES

Triple Exponential Smoothing......TES

Friedman’s rank Test……FRT

Cross Validation……CV

Blockchain……BC

Ethereum…….ET

Cryptocurrency……CC

Yahoo Finance….YF

Us Dollars…..USD

Graphical User Interface…….GUI

CHAPTER

1

INTRODUCTION

1.1 Background Overview

Ethereum (ET) is one of the most widely used platforms which offers high flexibility to develop decentralized applications. A programmer namely Vitalik Buterin has chiefly put forward this technology in 2013 and is also known as the co-founder of Bitcoin Magazine. In a conference held in Miami regarding Bitcoin in January 2014, Ethereum was announced (presented in Figure 1.1), which has added another technological development in the field of Cryptocurrency (CC). This technology was named Ethereum based on the search performed on the elements from science fiction (Vujičić, Jagodić and Randić, 2018) .

Figure 1.1. Ethereum (Vujičić, Jagodić and Randić, 2018)

The analysis of previous data states that a Swiss company has started the development of the software as the building block of Ethereum in 2014. The entire procedure of this technology was funded by organizing a public sale from July to August 2014 where the people used another digital currency namely Bitcoin in order to buy the Ethereum value token i.e. Ether. Afterward, various prototypes were developed from time to time to make the network upgrade to achieve more transaction rates by different agencies, institutes, and companies. Frontier was launched as the foremost release of Ethereum in 2015. Till then, this platform is rapidly growing and has several developers associated with it. This was a great advancement in the digital era but a number of questions have also come to attention concerning its security and robustness (*An*

Introduction to Ethereum and Smart Contracts: an Authentication Solution, no date). About $50 was stolen in Ether in 2016 by an unknown hacker which resulted in division within the community of Ethereum into two block chains i.e. Ethereum and Ethereum Classic. With time, rapid fluctuations have been observed in the ether price but the currency of Ethereum has grown more than 13000 in percentage in 2017 (*An Introduction to Ethereum and Smart Contracts: an Authentication Solution*, no date).

According to a study done by (Berentsen and Wiedmer, 2018) 55% of businesses are studying, investigating, or developing Blockchain (BC) solutions. Expenses, settlement structures, authorization, fundraising, privacy, and credits are just a few of the banking-specific services that could be affected by Blockchain. Ethereum is based on the concept of blockchain which is a type of distributed ledger technology that facilitates secure digital asset transfers without the use of a password. Banks, for example, are an example of a trusted authority. Blockchain (BC)-related innovations are gaining traction at a breakneck pace. An identifier of a chain is associated with the block that ensures the validity of that block. Each time a transaction is added according to the ordered sequence whenever a block is added by a node to its chain. The small network subsets are responsible for the communication of each node known as peers. Therefore, Ethereum is coming forward as a young and innovative technology that allows developers to develop and deploy any type of decentralized app.

1.2. Ethereum Vs Bitcoin

Despite various similarities between Ethereum (ET) and Bitcoin (BT), there are certain differences mentioned (Jani, 2017):

- Ethereum makes use of various exchange methods such as Cryptocurrency, ET machines, and small contracts while BT only includes Cryptocurrency.
- Proof of stake system is used by ET whereas BT used a proof of work system.
- Both permissible and non-permissible transactions can be performed using ET but BT only allows permissible transactions.
- ET average block time (12 seconds) is less than that of BT (10 minutes).

Thus, the significant features offered by Ethereum can be used to deploy the concept of Cryptocurrency (CC) in a more effective manner.

1.3 Problem Statement

Like the rest of the CC industry, the Cryptocurrency market is extremely volatile. In a span of months, days, or even a year, the Cryptocurrency market may change rapidly. As a result, scientific papers may become outdated and ineffective just a few months after they are released. In addition, as Ethereum is gaining prominence, other Cryptocurrencies are sliding through the cracks. They just need to complete a small amount of the Ethereum research. Finally, there is no comprehensive model for forecasting Ethereum's price drivers and the relationship between Ethereum and bitcoin price movements. Due to its speculative nature, previous literature does not support the prediction model of Ethereum prices in the market, which is the key focus of this research study.

1.4 Research Gaps

This research work identified certain gaps that motivated us to peruse further research in this domain as:

- Lack of quantitative evaluation of Ethereum prices in the market.
- Lack of a comprehensive model for Ethereum's price drivers forecast.
- Less exploration of machine learning platforms in the related area.
- Lack of use of a robust prediction model to analyze Ethereum price trends.

1.5 Proposed Objectives

This thesis work supports the development of a viable model for Ethereum price prediction, as well as the evaluation of Ethereum and Cryptocurrency responses. The best practices, such as applying Data Science methodologies and effective Machine Learning algorithms have been rigorously followed in this research project. This research work specifically focused on resolving the major problem of analysis of the trend in Ethereum and devising some prediction strategies. This study investigates how ether Cryptocurrency prices are projected and what trends can be observed in their pricing over time. Therefore, based on the gaps obtained, this work proposed some objectives that need to be accomplished as highlighted below:

1. To perform a comprehensive review of literature for a better understanding of the considered research area and the amount of work performed.
2. To propose an effective prediction model based on machine learning techniques i.e. linear regression, decision trees, and exponential smoothing
3. To analyze the price trends according to the results obtained using the above models and find the best one.
4. To make a comparison of the employed machine learning approaches to each other for

performance evaluation based on error rate.

1.6 Research Questions

Why do we need this Ethereum Analysis?

The rapid rise of the blockchain industry has directed focus towards the most popular Cryptocurrencies which are attracting a number of people worldwide. Ethereum is extensively utilized, and investing in it will be simple, and has a high possibility of success, thanks to the usage of machine learning algorithms. A pictorial representation of identified gaps, proposed objectives, and designed research questions is given in Figure.1.2.

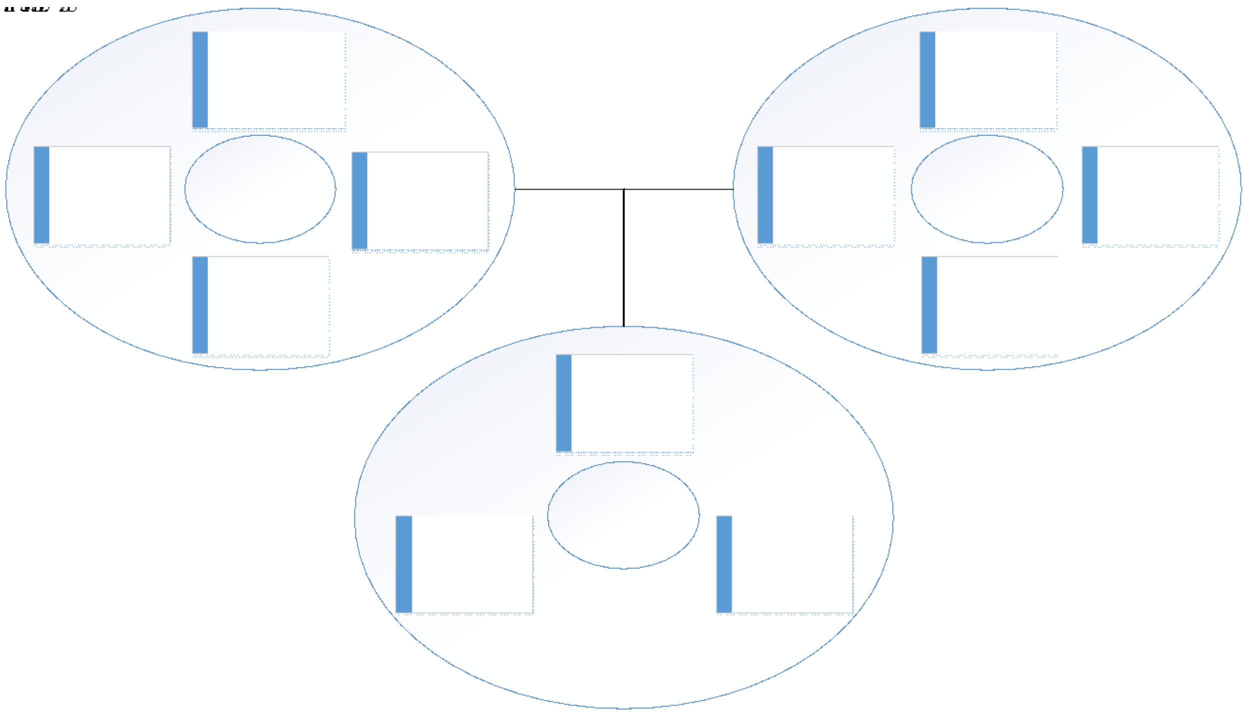


Figure 1.2. A depiction of identified gaps, proposed objectives, and designed research questions

What is Machine Learning and why use it?

Machine learning is a subfield of artificial intelligence (AI) and computer science which focuses on using data and algorithms to simulate how humans learn, with the help of data from history it tries to make future predictions. The task of Machine Learning (ML) is to make the machines intelligent to work like humans and perform various complex tasks very efficiently. This platform involves the use of various models such as Artificial Neural Network (ANN), Support Vector Machine (SVM), Decision Trees (DT), etc. to achieve the desired goals of prediction and classification (Cioffi *et al.*, 2020).

How will Machine Learning help with Ethereum analysis?

Machine learning (ML) has a great role in the prediction of market price trends with high efficiency. We have developed and employed statistical and machine learning approaches to use them for price fluctuation analysis. Optimistic results support the idea that machine learning may be used to investigate the predicted value of Cryptocurrencies and design profitable transaction techniques in these markets, even if they are still in the early stages of development (Grinberg, 2012).

1.7 Scope of study

A worldwide network of computers that collaborate as a supercomputer basically powers Ethereum. The blockchain is impenetrable, so the network assembles and executes smart contracts-applications that are, in principle, free from any intervention or censorship from outside parties. Smart contracts are self-executing, like an automat or vending machine that executes the provisions of the contract digitally, considerably lowering the risk of fraud. The product is transmitted or made available to the customer whenever specific requirements, such as the transfer of money, are demonstrated to have been satisfied.

In contrast to the internet, Ethereum stores all of these agreements and the transaction's data in independent blockchain ledgers rather than a centralized warehouse or servers making it less vulnerable to data breaches. The user is in control of their data. Ethereum is the second largest coin in crypto and has the highest market cap after Bitcoin (BT).

1.8 Significance of study

Ethereum is not just another blockchain. It is also a network of computers and a programming language that enable the development of blockchain-compatible apps. Ether serves as the foundation for these apps and is also used as cash on digital exchanges. Ethereum is a decentralized platform, meaning that no one entity oversees or controls the processes on the blockchain. A well-known network, it seeks to "codify, decentralize, secure, and exchange just about anything" in a wholesome environment.

Scalable, programmable, secure, and decentralized form all the features of Ethereum. It is the blockchain (BC) of choice for businesses and developers who are building technologies on top of it to transform several sectors and how we live our everyday lives.

1.9 Organization of Thesis

After Chapter 1 i.e. Introduction, the rest of the thesis is organized as follows: Chapter 2 provides the detail of the literature review and the challenges obtained. The proposed research methodology is presented in detail in Chapter 3. Chapter 4 provides a description of the implementation and an analysis of the results obtained. Finally, the conclusion and future work is provided in Chapter 5 which is followed by a list of references.

In this work, we have used SMA (Simple Moving Average) and EMA (exponential moving average) for the past trend analysis of Ethereum market prices from January 2015 to January 2021. Afterward, we applied linear regression (LR), decision tree (DT), and a few statistical models to make the comparison of accuracy. The proposed thesis work thus provides a brief explanation of Ethereum trend analysis and gives us an idea of future prices.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

A number of buyers and companies have shown a great interest in the usage of Cryptocurrency to allow digital transactions more reliably. A literature review is an in-depth analysis of past studies which is crucial to get an idea about the previous trends in the market. This chapter discusses the market trends based on the previous data related to Cryptocurrency adoption, the related work that has been already performed in the concerning field, the outcomes achieved from the analysis, and many more concerning the current trend of the market. Various studies from the state-of-the-art literature are securitized in detail to gather the related information covering different aspects. The analysis of data obtained after performing a literature review guided this work about the aspects that need to be covered and may be proved to be beneficial for the community as a whole.

2.2 Related Work

Different research studies performed in-depth surveys, and reviews and applied various models in the field of crypto currency adoption. Khedr et al. (Khedr *et al.*, 2021) presented a detailed survey to determine the extent of work performed related to Cryptocurrency using various statistical and machine learning (ML) methods. Different machine learning algorithms such as Regression, linear regression (LR), Support vector machine (SVM), Artificial neural networks (ANN), deep learning (DL), etc. seemed to be used very frequently in trend analysis of market price. The traditional methods are however easy to implement and generalize but the use of the ML platform has automated the prediction producing the best results in terms of accuracy. The proposed study presented a comprehensive survey of previous studies from the year 2010 to 2020. The gaps highlighted in this work can be proved helpful for the research community to focus on for further research.

Senthuran et al. (Senthuran and Halgamuge, 2020) proposed a study to analyze the prediction accuracies of both Ethereum and Bitcoin based on the historical data i.e. involving blockchain with Cryptocurrency data. A deep learning (DL) platform was utilized to perform such analysis and the data, as well as market price for both, were retrieved online. The results obtained demonstrated high accuracy for both currencies using blockchain data together. Further, the comparison among Ethereum and Bitcoin currencies showed the highest accuracy of prediction for Ethereum and less error rate. The employed DL approach for such analysis seemed to be very effective and provided accurate price prediction for both the currencies which may be beneficial for the financial market.

In a study by Ammer et al. (Ammer, 2022), an algorithm is presented known as long short-term memory (LSTM) to determine the price trends in four currencies out of which one of them was Ethereum. Various evaluation performance measures were adopted including Mean square error (MSE), root mean square error (RMSE), and normalized root mean square error (NRMSE) to check the performance of the LSTM model. The framework of the used model is provided in Figure 2.1. The findings achieved revealed the high efficacy of the LSTM model in predicting all of the Cryptocurrencies thus providing promising results. The correlation between the prediction and target values was accessed using the Pearson correlation metric. The performance values attained using the LSTM model were compared with those of the previous studies. Therefore, the implication of the LSTM model for the purpose showed the highest accuracy with less error. The limitation of this study was the consideration of a single LSTM model without using and comparing other models to check the prediction performance.

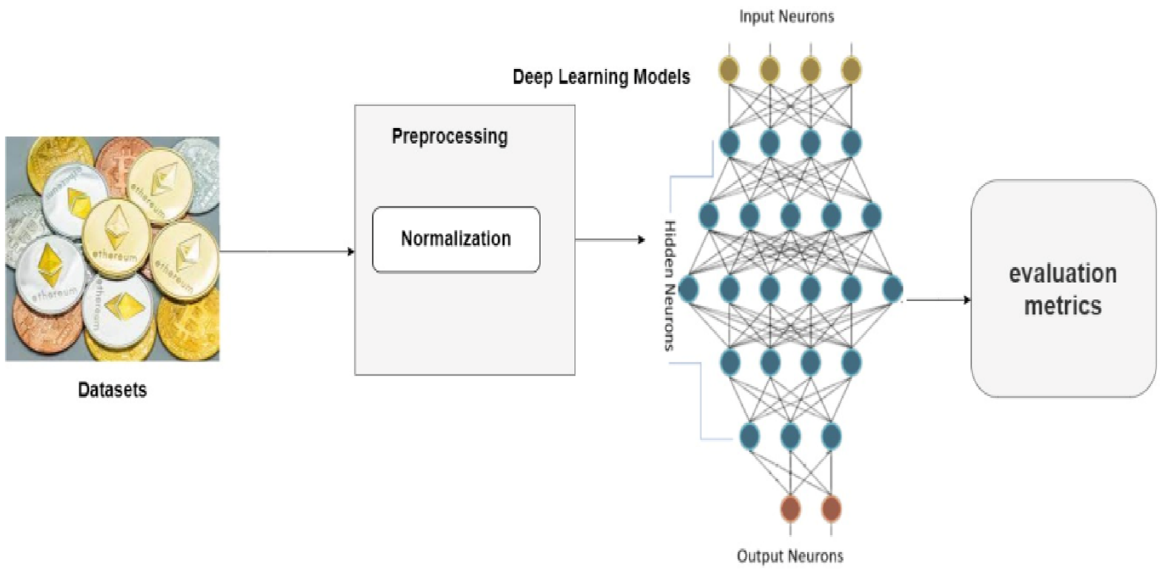


Figure 2.1. Proposed framework used by the study by Ammer et al.(Ammer, 2022)

Kristian et al. (Kristian, Adzikri and Rizkinia, 2021)presented a study to predict Ethereum price by applying two machine learning (ML) algorithms i.e. k- Nearest Neighbors (KNN) and multiple polynomial regression (MPR). Various independent variables such as gold price, oil price, and Ethereum volume were utilized for experimentation purposes. The data was gathered from the year April 2017 to 2021. Data preprocessing was performed considering different scenarios and evaluation parameters involving MSE and MAE were used for performance evaluation. The results obtained revealed the high prediction performance on applying KNN in comparison to MPR thus highlighting the significance of KNN in such analysis.

Oliveira et al. (Oliveira *et al.*, 2021) proposed a study to investigate a key factor of Ethereum. The transaction confirmation or failure was taken as the prime aspect to be determined based on the features of Ethereum. To perform such experimentation, the balance set of data involving both confirmed and failed transactions were considered and analyzed using machine learning methods. F1-score and receiver operating curve (ROC) were used to evaluate the performance of the models. The classification transaction results obtained showed 0.67 for the F1-score and 0.87 for the ROC values revealing the considered model's efficiency. Further, a Gas feature of Ethereum is analyzed to be more relevant in such prediction estimation.

In a research work by Kumar et al. (Kumar and Rath, 2020), deep learning techniques were adopted to analyze the price trends for Ethereum with particular regard to time series trends. Multilayer perceptrons (MLP) and LSTM were implemented for to purpose. The experimentation data was calculated each day, hour, and minute wise and then these techniques were applied. An online data repository namely CoinDesk was preferred to perform such analysis using the DL paradigm. The results achieved demonstrated high accuracy with a low error rate in the prediction of price trends for Ethereum.

Derbentsev et al. (Derbentsev, Matviychuk and Soloviev, 2020) aimed to solve the issue of short-term time series of Cryptocurrency by applying machine learning algorithms. Three currencies i.e. Bitcoin, Ethereum, and Ripple were considered involving their 90-day time horizon. Machine learning techniques such as the Binary Autoregressive Tree model (BART), Neural Networks, and an ensemble of Random Forest (RF) were used for the analysis. The used models showed significance in analyzing the trends automatically.

Hamayel et al. (Hamayel and Owda, 2021) proposed a study to analyze and predict the price of three Cryptocurrencies namely Bitcoin, Litecoin, and Ethereum. The mean absolute percentage error value was used as the performance evaluation metric. Three machine learning models i.e. LSTM, bi-directional LSTM (bi-LSTM), and Gated recurrent unit (GRU) were considered by the study to perform the estimation of price. The analysis of results yielded the highest accuracy for the GRU model followed by LSTM. Bi-LSTM has shown the lowest prediction accuracy in such prediction. Thus, the utilization of these models can help investors to estimate the sale of Cryptocurrency.

A study was presented by Valencia et al. (Valencia, Gómez-Espinosa and Valdés-Aguirre, 2019) in which machine learning techniques were utilized to predict the price trend of four types of Cryptocurrencies including Ethereum, Bitcoin, Ripple, and Litecoin. Three machine learning algorithms namely ANN, SVM, and Random forest (RF) were used for the implementation and the experimentation was performed on the social media data. The platforms such as Twitter were scrutinized and the data was gathered which acts as the input to the models. The results achieved demonstrate the relevance of sentiment analysis and machine learning platforms in the prediction of Crypto currency markets price. Also, out of all the models used, ANN seemed to have gained the highest accuracy. Poongodi et al. (2020) [14] employed two machine learning algorithms in this study i.e. SVM and LR involving time series comprised of daily closing prices of Ethereum. To predict the price of Ethereum, distinct window sizes were considered utilizing the concept of filters having varying weight coefficients. A cross-validation method was applied for the building of the model with high performance in the training phase not dependent on the dataset. The results obtained after the implementation of LR and SVM revealed a higher performance with SVM achieving an accuracy rate of 96% in comparison to the LR technique which attained an accuracy rate of 85%. Further, the results with SVM can be enhanced i.e. more accuracy can be achieved by adding more features that can be given as input to SVM.

Kim et al. (Kim, Bock and Lee, 2019) proposed a study that aims to estimate the uncle block occurrence to resolve the problem of blockchain. New attributes were practiced and an ensemble learning technique was employed to make an accurate prediction of the occurrence of uncle block. Three ensemble techniques such as voting, bagging, and stacking was applied for the prediction purposes. Blockchain information and Bitcoin were used for the data analysis. The results achieved showed that high prediction results were obtained with voting and stacking techniques of the ensemble method when utilized with Ethereum Blockchain data. Therefore, the used technique proved to be successful in making an accurate prediction for the occurrence of Blockchain. In a study by Saad et al. (Saad and Mohaisen, 2018), two Cryptocurrencies were studied and explored namely Bitcoin and Ethereum. Further, the study examined the features that cause price hikes in both currencies. The data that significantly impacts the price of these Cryptocurrencies are gathered and the user, as well as network activity, is also analyzed. Using such information, prime features were identified as relevant to determine the demand and supply factors in Cryptocurrencies. Further, machine learning models were experimented with to estimate the price of Bitcoin. The results obtained on two datasets have shown the best accuracy i.e. 99% for correctly predicting prices for both Bitcoin and Ethereum.

Politis et al. (Politis, Doka and Koziris, 2021) proposed a methodology for constructing deep learning algorithms to estimate the price of Cryptocurrencies and predict the Ethereum price. The study overcomes the limitations of statistical methods that are not fit to capture Cryptocurrency complexity. The uniqueness of deep learning models of not requiring human intervention makes the prediction process easier and the applied model yielded accuracy of up to 84.2% which may be beneficial for researchers in predicting short-time and long-time price forecasts. Zoumpekas et al. (Zoumpekas, Houstis and Vavalis, 2020) proposed a study to perform a data analysis of the market of Cryptocurrency. The study made use of statistical methods and machine learning techniques to predict the variations in the price and to make inferences. Further, deep learning models were utilized to estimate the price of Ethereum in short term. The data was gathered from Poloniex and Convolution Neural Network (CNN) was implemented with four Recurrent Neural Networks (RNN) including GRU, LSTM, Bi-LSTM, and stacked LSTM (s-LSTM). The models used provided promising results for such prediction analysis.

Figure 2.2 provided below represents the Ethereum price prediction data for the period 2017 to 2022 (*Cryptocurrency (Ethereum) Price Prediction Using Python*, no date). Similarly, Figures 2.3, 2.4, and 2.5 depict the price prediction of Ethereum for the years 2021, 2022-2023, and 2024-2025 and beyond (*Ethereum (ETH) Price Prediction 2022, 2023, 2025-2030 | PrimeXBT*, no date).

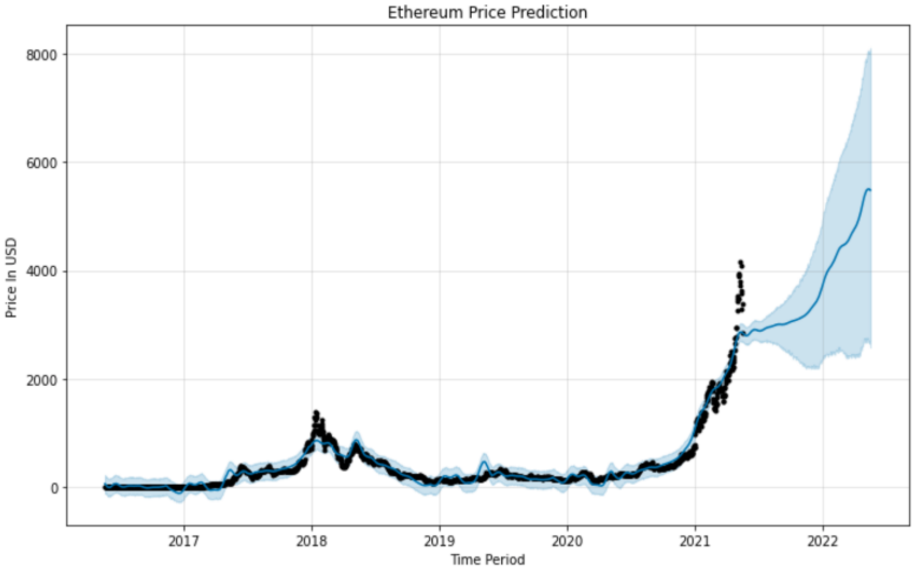


Figure 2.2. Price prediction of Ethereum for 2017-2022

The data used in below figure is tabulated in the form of a table i.e. Table 2.1 that indicated the low and high forecast prices for Ethereum. By analyzing the data in the table, traders can be benefited to estimate the rising and fall in price for short and long periods.

Table 2.1 Data representing Price Forecast for Ethereum (high/maximum and low/minimum) for years 2021, 2022, 2023, and 2024-2025 (*Ethereum (ETH) Price Prediction 2022, 2023, 2025-2030 | PrimeXBT*, no date).

Year	**Potential High**	**Potential Low**
2021	$14000	$1440
2022	$10000	$1440
2023	$7200	$2600
2024-2025	$41000	$4500

In 2021, a high of about $4,400 is reached as high for Ethereum so far, which caused a 60% crash as a result altcoin dropping below $2,000. For Ethereum, the potential high can be observed as much as $14,000 per Ethereum.

Figure 2.3. Price prediction of Ethereum for 2021

After attaining $14,000 per Ethereum, it comes down in the years 2022-2023 and enters into another bear market. This resulted in a fall in the current consolidation range of Ethereum. This fall was a clear indication to keep a proper check on the built quality of Ethereum a retesting should be performed on it for better outcomes.

Figure 2.4. Price prediction of Ethereum for 2022-2023

After a decline in the range of Ethereum, it is estimated that it will again start increasing in the upcoming years to build a much better position to enter the next rally. This will set new standards for future perspectives.

Figure 2.5. Price prediction of Ethereum for 2024-2025 and beyond

2.3 Challenges in Previous Literature

The exploration of literature comes up with certain challenges related to Ethereum which need to be focused on for more effective price prediction analysis as:

- Previous research studies have tried to make models which are based on time series only.
- Lack of research performed using robust models having less variance and may produce good results.
- GRU and LSTM seem to be used more frequently but have certain pitfalls including low convergence rate, less learning efficiency, more training time, overfitting issues, and many more.
- Very little research is directed towards performing experimentation based on regression analysis which can be effective in the prediction of Ethereum price trends.
- Lack of comparison of multiple models to each other based on performance evaluation metrics.

2.4 Summary

This chapter explores the related research work performed on Ethereum to determine its price prediction trends in the market. Several statistical and machine learning methods were identified and analyzed to achieve the best results. The existing literature revealed that for trend prediction, models based on GRU and LSTM have been utilized the most. Among machine learning algorithms, models such as ANN, SVM, LR, RF, DL, and BART have been practiced to make the Ethereum market price prediction more accurate. Various performance evaluation measures were used including RMSE, MSE, NRMSE, MAE, F1, and ROC to evaluate the efficacy of different models.

The next method explains the proposed research methodology used to achieve the designed objectives.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

Research methodology plays an important role in the successful accomplishment of a task. It comprises several important steps starting from data collection to prediction results. The literature shows that there are numbers of prediction and classification methods that can be used to achieve high accuracy. This work applied three methods namely linear regression (LR) (Cycles, 1989), Decision trees (DT) (*1.10. Decision Trees — scikit-learn 1.1.2 documentation*, no date), and exponential smoothing (ES) (*Exponential Smoothing Methods for Time Series Forecasting*, no date) to make the price prediction of Ethereum. Based on a vast amount of comment data, regression analysis (RA) is a numerical approach for constructing a regression connection between the dependent and self-determining elements. In other words, regression analysis is the combination or set of statistical methods which are employed to estimate the relationship between one dependent and one or more independent variables. Linear regression has been widely used in previous studies due to its simpler implementation. Linear regression is also easier to interpret and possess simple and efficient training. One of its unique features is that it is specially designed to be used for making trend analysis and estimation of predictor's strength. Decision tree is another machine learning technique that performs prediction and classification firstly by building the model. The process starts from the root node and continues to the leaf nodes of the tree. The exponential window function is a general method for smoothing time series data known as exponential smoothing. In contrast to the ordinary moving average, which weights previous data equally, exponential functions use weights that decrease exponentially with time.

The first blockchain platform to enable digital conventions was Ethereum. When a digital contract is run on the blockchain, it transforms into a self-contained mainframe programme that runs when certain criteria are met. Blockchain-based digital contracts allow code to be written without wasting time, restrictions, scams, or the involvement of third parties. In this thesis work, each of the models is applied one by one to determine its accuracy in the prediction of Ethereum prices correctly. The concept of each of the approaches i.e. linear regression, decision tree, and exponential smoothing is incorporated with predicting the price trend of Ethereum to make more efficient estimations and analysis with high accuracy. At last, the results of these methods are compared to each other to determine the best among them depending on certain parameters such as root mean square error (RMSE) (Chai and Draxler, 2014).

3.2 Proposed Algorithm

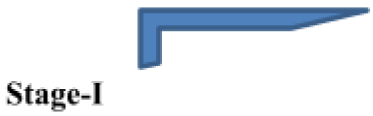
An algorithm is the sequence of steps that are followed to implement a strategy from beginning to end successfully. To achieve the purpose of predicting the price trend of Ethereum using various machine learning approaches, an algorithm is designed which is followed in further work and is presented below:

Algorithm 3.1

1. *Acquire the data from online repository i.e. Ethereum yahoo finance api that contains historical price data for past years.*
2. *Perform preprocessing of data by removing null values and scaling of data.*
3. *Perform data visualization to represent data in graphs using different measures such as arithmetic moving average (AMA) and weighted moving average (EWMA).*
4. *Perform modelling of data by applying machine learning models including Decision tree (DT), linear regression (LR) and Exponential smoothing (ES).*
5. *Train each predictor one by one for the preprocessed data values using 70% data.*
6. *Test each method or predictor with the remaining data.*
7. *Perform evaluation of performance for all the methods applied and compare the results with them based on performance evaluation metrics.*
8. *Validate the efficacy of the model comparison with state-of-art studies.*

3.3 Proposed Methodology

In this research work, the attention is directed towards the analysis of trends and estimation of Ethereum prices with the help of a linear regression model. The proposed framework or methodology followed for such analysis is comprised of several stages i.e. (1) Dataset Collection (2) Data Preprocessing (3) Data Visualization (4) Modeling (5) Output, as shown in Figure 3.1 **Stage-I** includes the collection of relevant data related to financial markets which will be further used for the experimentation. In **Stage II**, data is preprocessed to make it clean to perform the operations efficiently. **Stage III** involves data visualization to analyze the data in a more detailed manner. In **Stage-IV**, modeling is performed by employing machine learning techniques and lastly, **Stage-V** gives the output predicting the price trend of Ethereum. A detailed explanation of all these stages is provided as under:



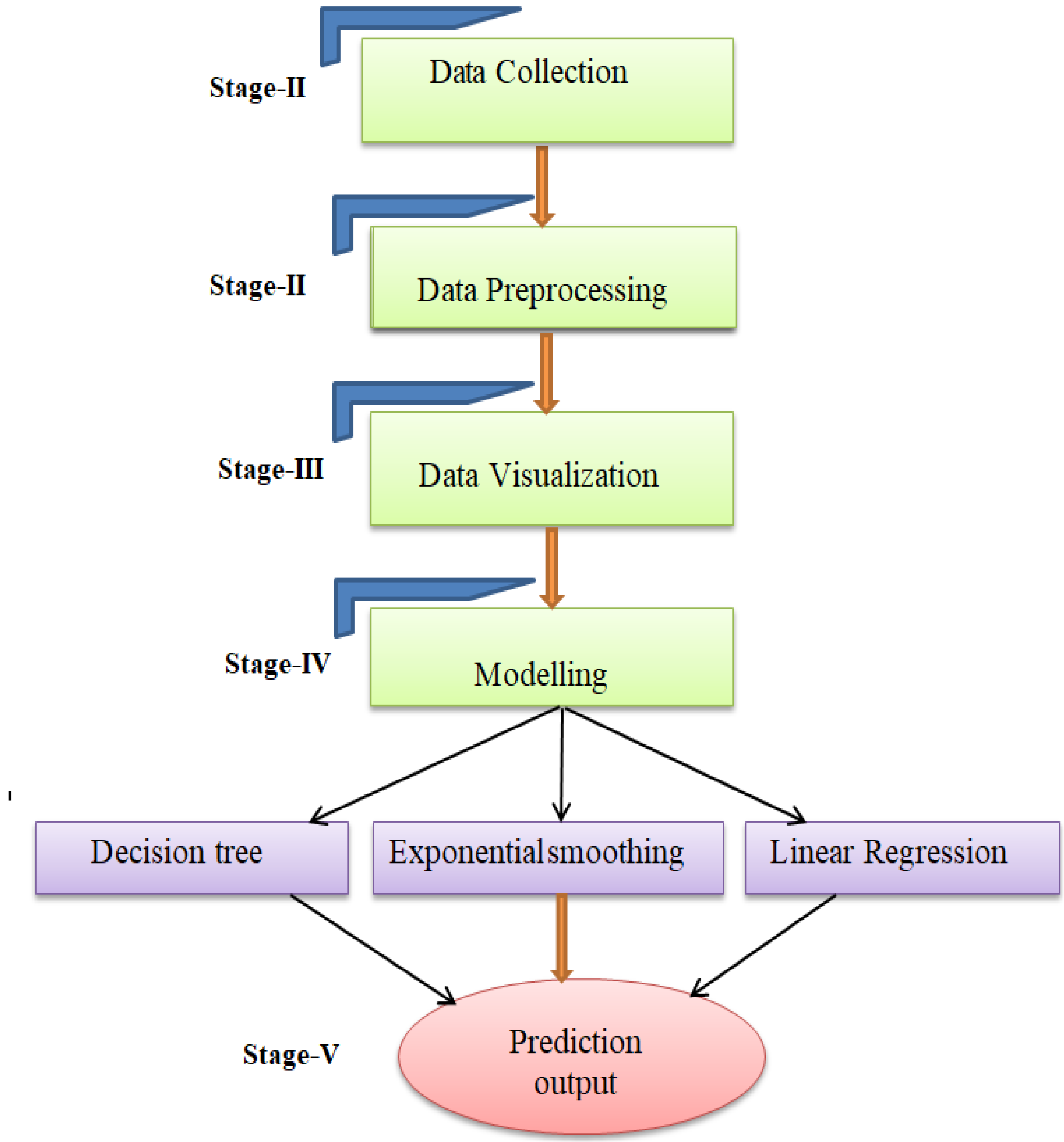


Figure 3.1. Proposed Methodology

3.3.1 Stage-I: Dataset Collection

Dataset plays a crucial role in any of the implementation process. It is the backbone of a model based on which experimentations are performed. The data may be primary or secondary. Primary data refers to the data which is collected by an individual himself and can be considered as the real-time data. This data includes personal interviews, questionnare, and various surveys which are gathered by personally visiting various companies and field work. The primary data is available in raw form which needs to be process further to extract meaningful information from it. This includes a long collection procedure but is very expensive and is in accordance with the individual's specific requirements. On the other hand, Secondary data refers to the data which has already been collected by an individual and then used by the others. In other words, this type of data is collected by someone else earlier and is already in processed form in most of the cases. This data can be considered as the past or old data. This data can be gathered from government official websites, various publications, research articles, government files and records. This data can be collected in a very short span as it is already available in online platforms. The secondary data is very economical in comparison to primary data and may not be in accordance with the individual's specific requirements. Primary data can result in more accuracy than the secondary data as it is collected as per proper requirements. One of the advantages of the secondary data is that it is the benchmark data and has more reliability. When we collect the primary data, it can be considered accurate unless and until it has been validated and is published on some benchmark online platform. In this research, we have utilized the secondary data for the implication purposes as a large amount of financial data is required in this work. Therefore, to perform implementation, we have used yahoo finance (YF) API to get the dataset. This data used the date-time index to get the data between the years of 2015 to year 2021. So, a huge amount of financial data related to the price of Ethereum is present in this dataset. Some of the examples of data from the entire dataset is given in the Appendix section. The **Yahoo Finance API** is used for obtaining real-time and historical data for many different financial markets and products, as displayed on **Yahoo Finance** (*Ethereum Data | Kaggle*, no date). This data contains 1150 rows of data points and different columns namely high, low, open, close, volume, and adj close. The data is available online at the Kaggle repository and provides historical information for daily price rates of Ethereum. The data in the dataset starts on the 7th of August 2015. The currency is taken in Us Dollars (USD). This data is also available in other platforms other than kaggle including CoinmarketCap, etc.

3.3.2 Data Preprocessing

Preprocessing is a crucial stage in which various operations such as noise filtering, resizing, rescaling, cropping, etc. are performed to enhance the quality of the data. The purpose of preprocessing is to transform the raw into processed form to attain more accuracy and increase the performance of the model. Preprocessing is the form of data preparation which is most widely used in the process of data mining and data analysis in various fields. In this work, the null values are checked. If these values were present in the data, they are removed using the dropna function (dropna ()) (*Pandas DataFrame dropna() Method*, no date). The main aim of this function is used to eliminate the rows that involve null values. Thus, the null values are replaced with the mean of the data set in Numpy library. Apart from that, scaling technique is applied to scale the data using a standard scaler. The Standard Scaler follows Standard Normal Distribution (SND) (*The Standard Normal Distribution | Examples, Explanations, Uses*, no date). Therefore, it makes mean = 0 and scales the data to have variance =1. In other words, this function discards the mean and each variable is scaled to the unit variance. After performing the above preprocessing operations, the entire dataset is divided in train and test. Also, all the columns except close for training and testing purposes were dropped. The preprocessing operations performed are depicted in Figure 3.2.

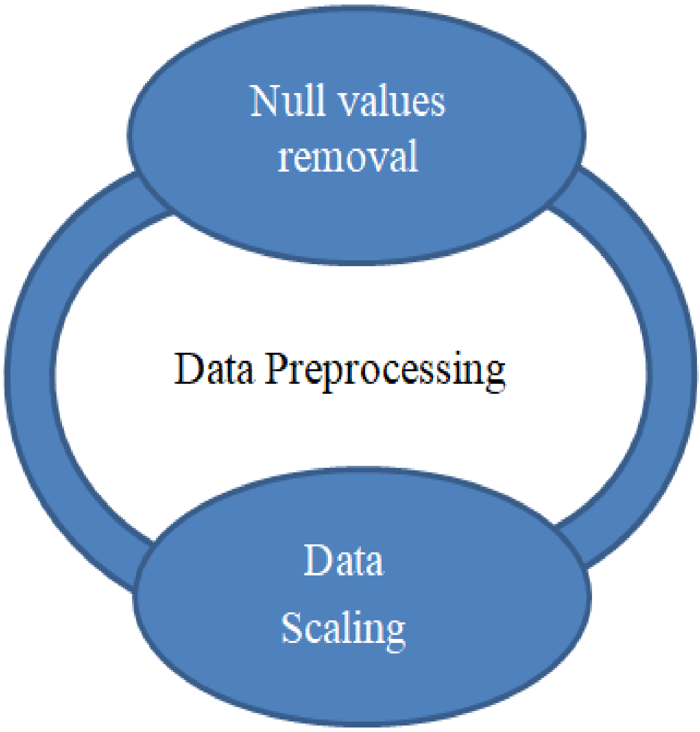


Figure 3.2. Data preprocessing operations used

3.3.3 Data Visualization

When we represent the data with the help of some graphics such as graphs, bars, pie-chart, plots, etc. to make its interpretation easier, it is known as data visualization. In simple words, data visualization is simply the displaying of data and information in graphical form. The representation of the information in such a way allows for analyzing the data in a more detailed and easier way. The basic idea of different structures that can be utilized to represent the data is provided in Figure 3.3 (Margret Anouncia, Gohel and Vairamuthu, 2020).

Figure 3.3. An example of data visualization structures

To check and analyze the trend of the open and closed prices of Ethereum data, this work plotted some graphs. One month and two weeks' simple moving average was used to analyze the data more precisely and in more detail. An arithmetic moving average (AMA) which is known as a simple moving average (SMA) (Tutorial *et al.*, no date) is created by averaging recent prices and dividing the result by the number of periods used in the calculation. Therefore, this measure makes the computations for the arithmetic mean for a given set of prices over a particular number of days. To represent or explain a time series, a quantitative or statistical measure known as the exponentially weighted moving average (EWMA) (Sukparungsee, Areepong and Taboran, 2020) is taken into consideration. The EWMA has several applications in banking, with technical analysis and volatility modeling being the two most common ones.

The data presented in the above tables i.e. Table 3.1 is analyzed graphically and presented in the form of graphs i.e. Figure 3.4 and Figure 3.5 for better understanding.

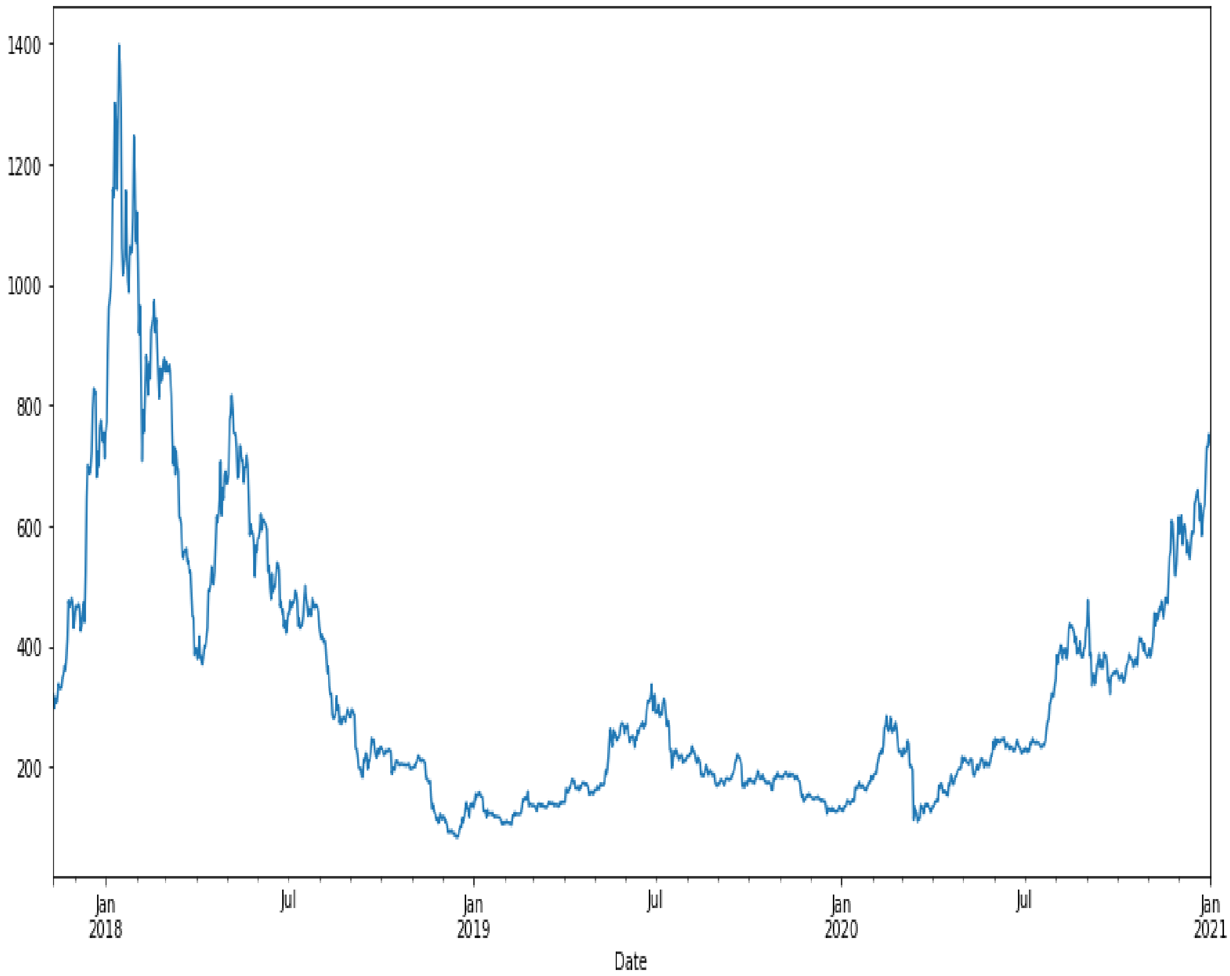


Figure 3.4. Representation of data from 2018 to 2021 for ‘open’ attribute graphically

Therefore, a two-week Exponential Weighted Moving Average (EWMA) is determined and plotted with the help of a normal curve for better understanding and interpretation. In this type of average, the most recent data points are allocated with the highest weight as well as significance. This average is also named as Exponential Moving Average (EMA).

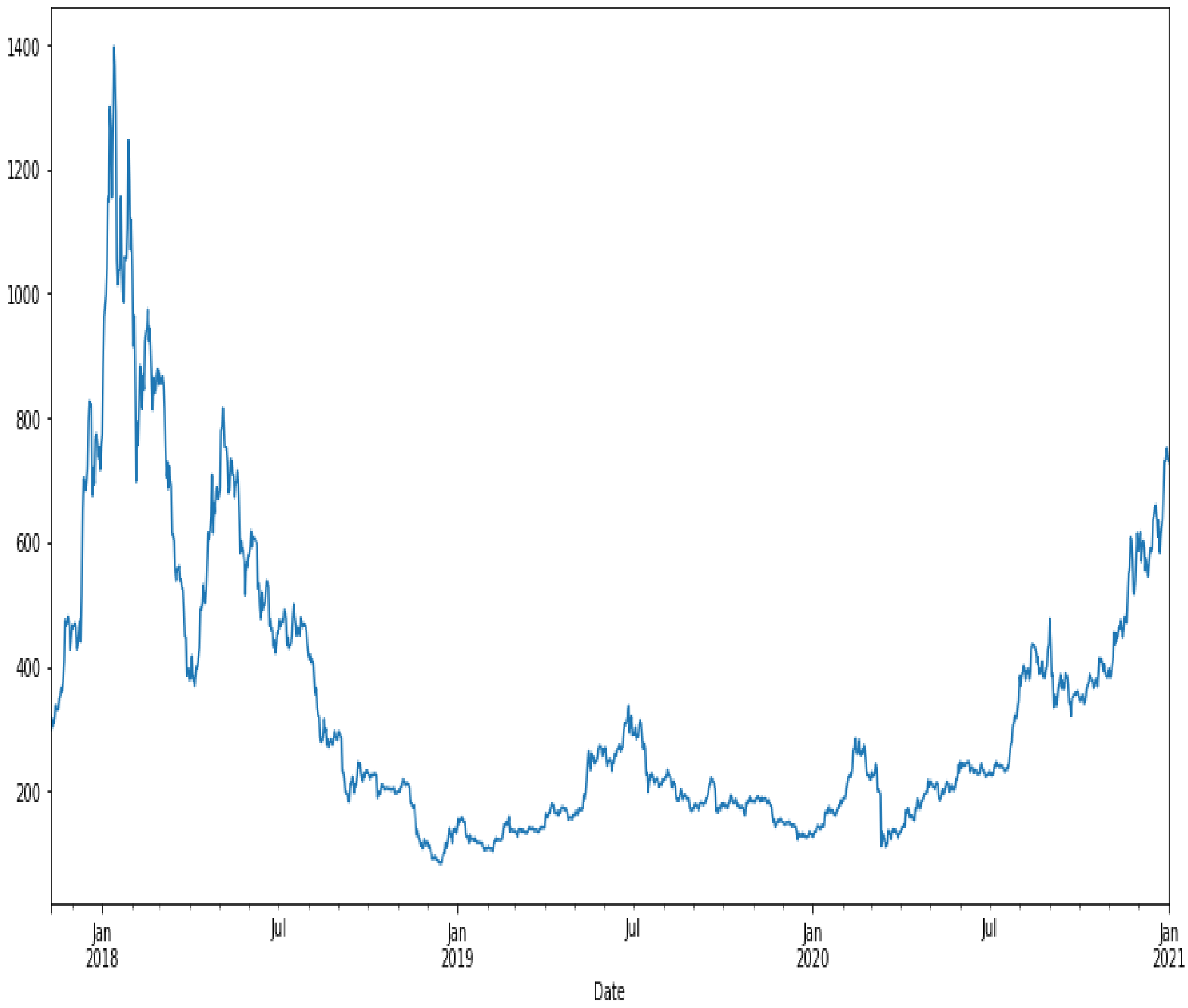


Figure 3.5. Representation of data from 2018 to 2021 for 'close' attribute graphically

3.3.4 Modeling

Once the process of data preprocessing and data visualization has been completed, the modeling step is performed. This phase involves the building of the models based on certain rules and strategies. A data model refers to the abstract version of a model that allows further creation of conceptual models to establish the relationship among data points. The modeling should be performed in such a way that all the selected parameters show proper suitability and achieve the highest prediction results. There are three models which are used in this work to perform the prediction of price trend analysis for Ethereum as described below:

3.3.4.1 Decision tree (DT)

Decision tree (DT) (*Decision Tree Algorithm, Explained - KDnuggets*, no date) (in Figure 3.6) is the most widely used supervised learning technique that can serve both the purposes of classification and regression. The basic motive of a decision tree is to predict the target variable's value based on the simple decision rules extracted from the related features set. A decision tree employs a tree-like model to represent options and their potential results, including several variables and chance event outcomes. In a decision tree, the internal node is the depiction of the applied test and the tree branch represents the output of the test performed. A decision trees comprises a tree flow chart-like structure that can able to handle both categorical as well as numerical data. Therefore, it consists of leaf and decision nodes as depicted with the help of Figure 3.6. To completely understand the concept of decision trees, some terminologies are used:

- **Terminology**

1) **Node:** Every object in the tree represents a node. It indicates the whole population which is further split into two or many similar sets.
2) **Parent Node:** The parent node is the node that is split into sub-nodes.
3) **Child Node:-** The child of the parent node represents the child node.
4) **Leaf Node:-** Also known as the terminal node and can't get split.
5) **Branch:-** It is the subtree of the entire decision tree.
6) **Root:-** The topmost node of the decision tree is the root node.

- **Assumptions while creating a Decision Tree**

While using Decision trees, assumptions should be followed (*Understanding Decision Tree, Algorithm, Drawbacks and Advantages. | by Sagar Rawale | Medium*, no date) :

1) At the start, the entire training set is treated as the root.
2) The values of features are supposed to be categorical. In the case of continuous values, discretization is done before constructing the model
3) The attribute values are used to recursively distribute the records.
4) To decide the order in which nodes should be placed in the tree is done according to some statistical method.

- **Working of Decision Trees**

Decision Trees (*Decision Tree Algorithm, Explained - KDnuggets*, no date) are one of the important methods in machine learning that works on linear as well as non-linear data. These algorithms work according to these algorithms works according to the rules made on data. The accuracy of the decision trees heavily depends on the decision to split the tree i.e. correctly deciding the number of splits. This strategy is different for regression as well as classification trees. Therefore, multiple algorithms are utilized to divide a node into two or more sub-nodes.

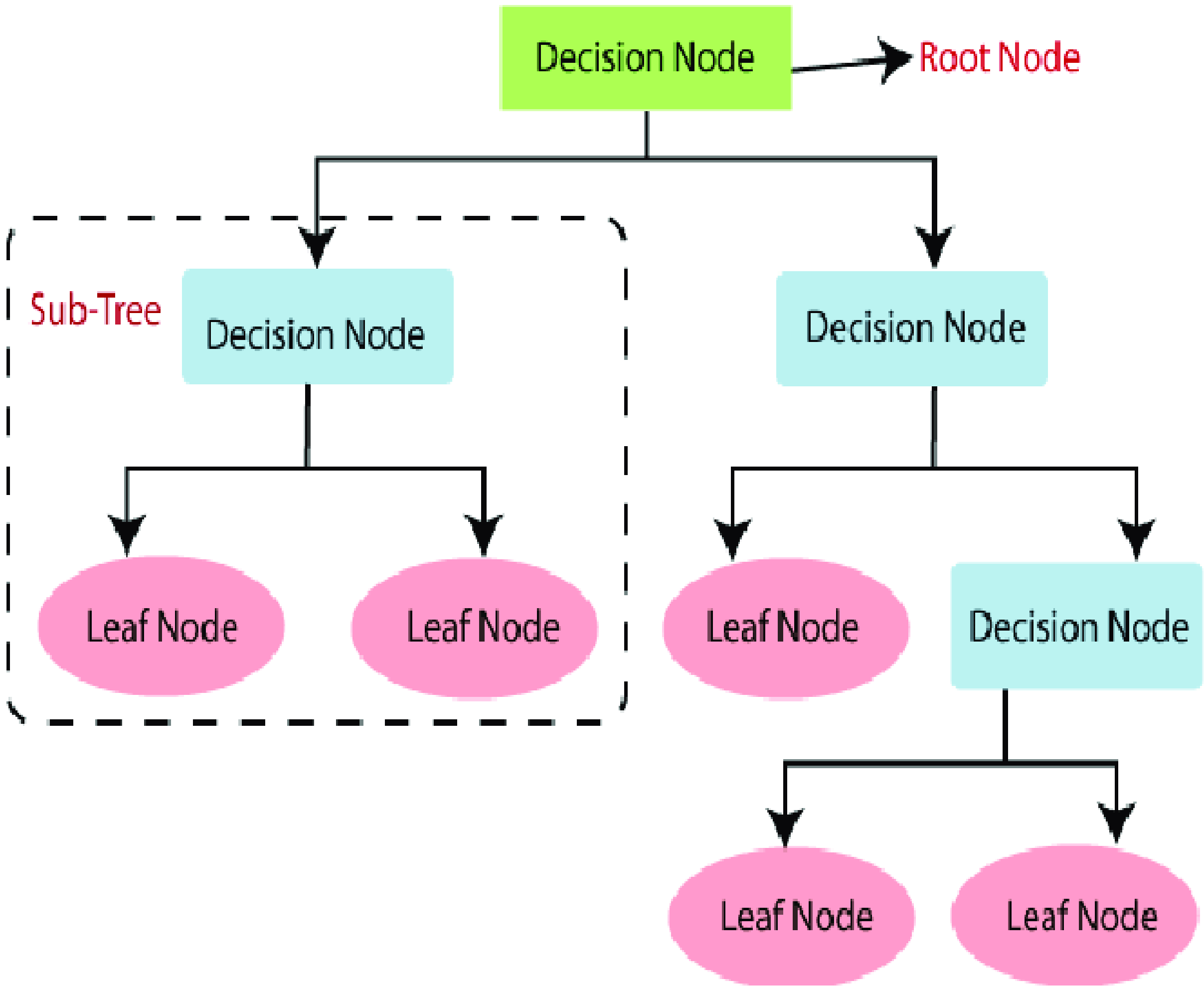


Figure 3.6. Decision tree flowchart structure (*Structure of the decision tree (DT) algorithm. | Download Scientific Diagram*, no date)

The nodes are partitioned by the decision trees on all the variables which are available and the division that yields identical sub-nodes is chosen and selected.

Initially, the algorithm starts from the decision's tree root node to predict the class in a given dataset. Then the values of the root are compared with the record attribute present in the real dataset. Depending on this comparison, the jump is made to the other branch to start with the next node.

The value of the attribute is re-compared with the value of other sub-nodes for the next node to move in a further direction. This procedure will continue till the algorithm reaches the destination node of the tree i.e. leaf node. The entire process can be easily understood with the help of the following algorithm.

1) Start with the root node of the tree 'R'. This node includes the entire dataset.
2) Determine the attribute which is best among all in the dataset with the help of some

attribute selection measure (ASM).

3) Split the root node 'R' into subsets 'S' which contain all the possible values for the attributes that are best.
4) The node i.e. decision tree node which has the best attribute is generated.
5) By utilizing the subsets of the entire dataset as constructed in step no.3, generate new decision trees recursively.
6) Continue with this procedure till where the further split of a node is not possible and this node will be the final node i.e. leaf node.

To select the attribute in step 2, there exist several methods but the most widely used methods include information gain and the Gini index. These measures tell the number of nodes in which the split should be done.

Information Gain (IG):

- IG (depicted in Figure 3.7) refers to the measurement of entropy deviation which is performed after the split of the entire dataset is made depending on some attribute.
- The calculation of the amount of information reflected by a feature about a class is the main purpose of IG.
- Based on the values obtain for IG, the node is divided and the decision tree is constructed.
- The decision tree techniques always keep on trying to increase the IG value and the node which attains the maximum/highest value of IG is split first. IG can be easily computed using the formula provided below

$$\text{Information Gain (IG)} = \text{Entropy(E)} - [(\text{weighted average}) * \text{Entropy(feature)} \quad (3.1)$$

Where entropy is the measurement of the amount of impurity or randomness present in a given attribute from the dataset and can be calculated by the formula below

$$\text{Entropy (E)} = -S(y)\log2\ S(y) - S(n)\log2\ S(n) \quad (3.2)$$

Here E denotes the total sample number, S(y) and S(n) represent the probability of yes and no.

Gini Index (GI):

- While building a decision tree in the CART (Classification and Regression Tree), GI is applied to determine the purity or impurity.
- The attribute which has low GI value is given the more preference in comparison to the attribute with a high value of GI.

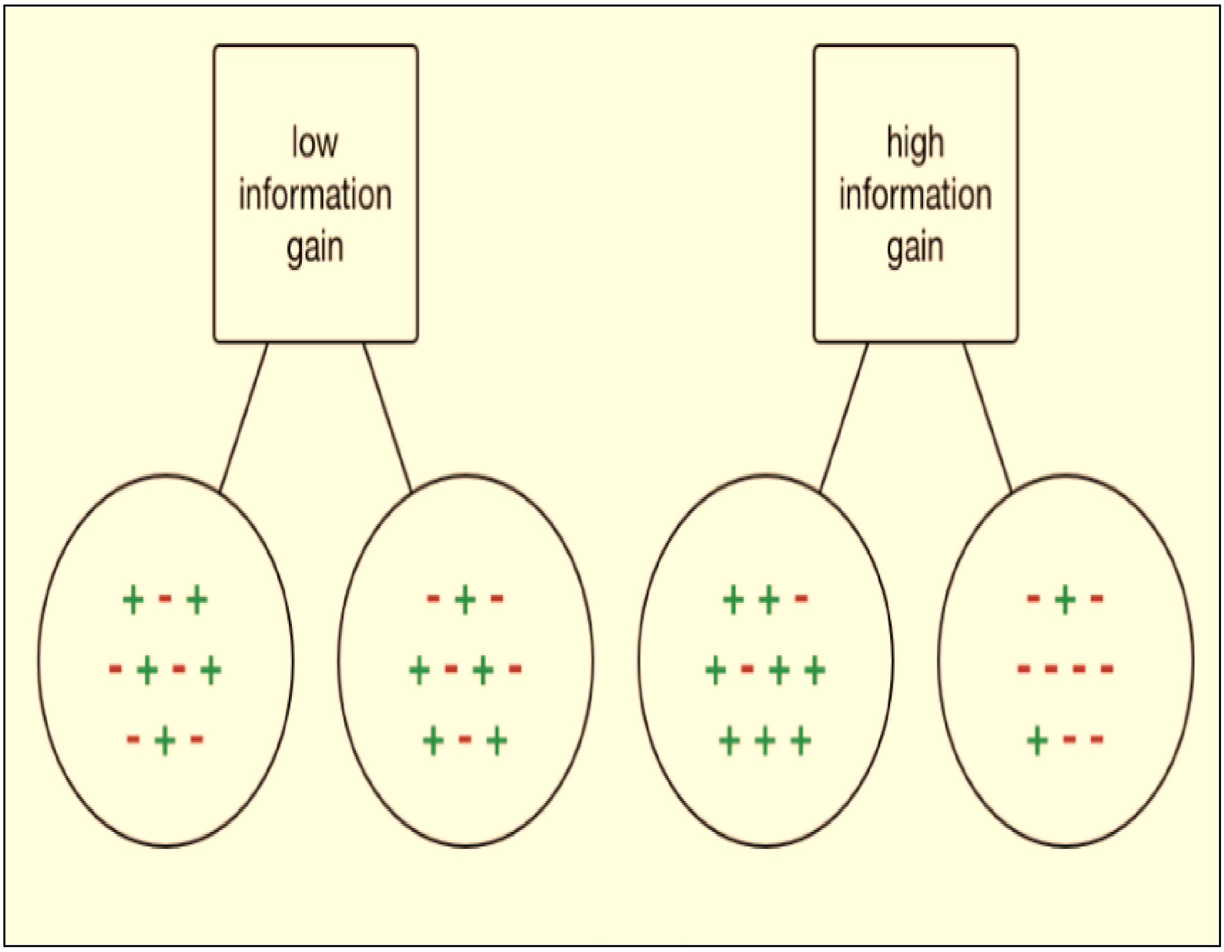


Figure 3.7. Information Gain Depiction (*Decision Tree Algorithm, Explained - KDnuggets*, no date)

- This measure is used by the CART algorithm and can able to create only binary splits.
- The higher value of GI denotes high inequality or low homogeneity.
- This measure can be computed using the formula below.

$$\text{Gini Index (GI)} = 1 - \sum_i S_i^2 \tag{3.3}$$

- **Example of Decision Trees**

Let us consider that you go on shopping and you have several choices in the mall. For a particular shirt, you want to decide whether to purchase it or not. Therefore, to tackle this issue, the process starts at the root node (Price attribute using ASM) by the decision tree. This root node will be re-splitted into the next node on which a decision needs to be made (Cloth type) and one node i.e. leaf node depending on the corresponding labels. Further, the next is split into a decision node (Color needed) and a leaf node. Finally, there occurs a final split of the decision node into two leaf nodes i.e. (Purchased and Not-purchased). The whole scenario can be understood in Figure 3.8 provided as under. Therefore, in decision trees, at every node, the tree can be split into further sub-nodes according to different decisions made, and at the end, it returns the optimal attribute value.

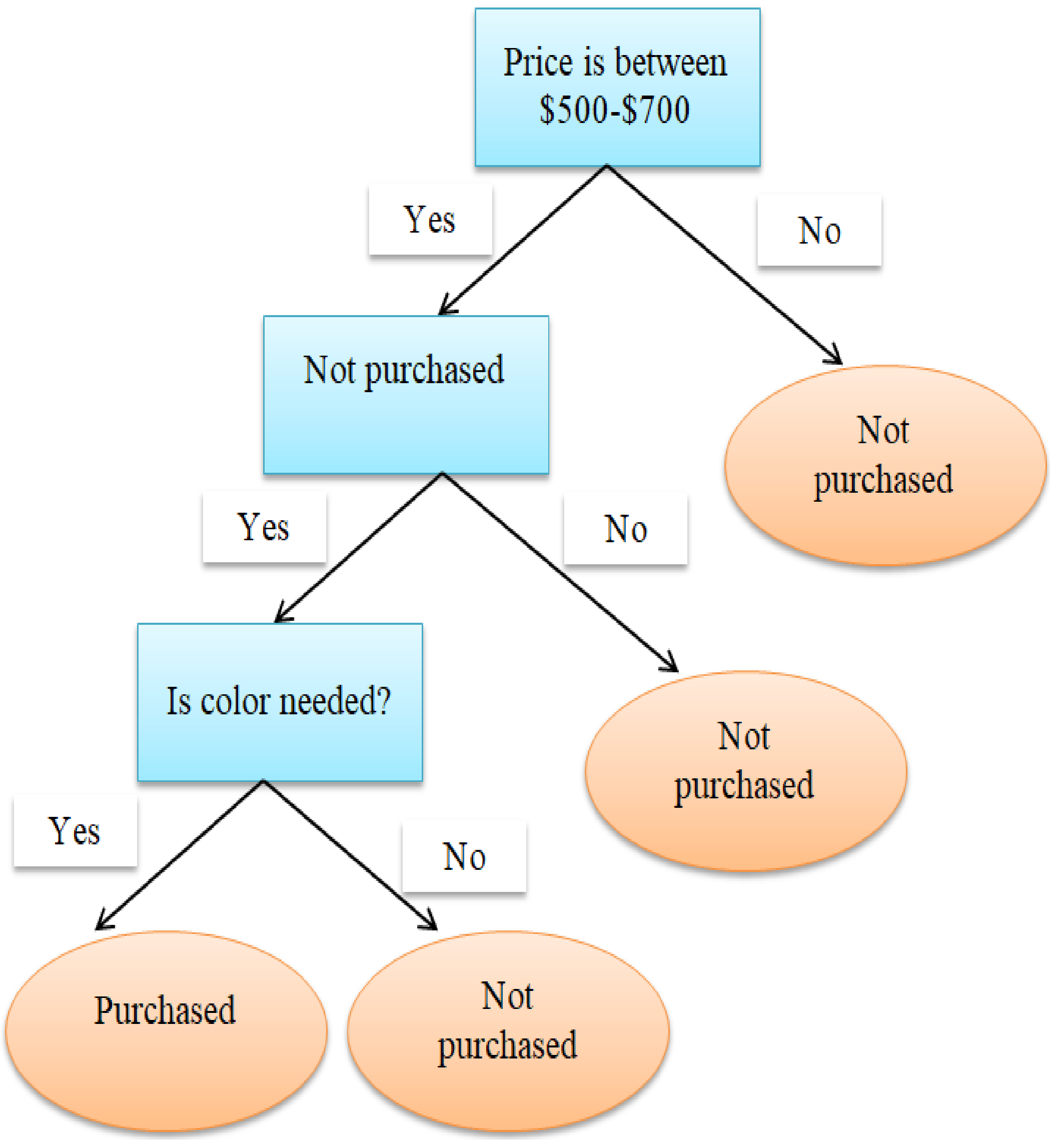


Figure 3.8. An example of decision trees

- **Advantages of Decision Trees** (*1.10. Decision Trees — scikit-learn 1.1.2 documentation*, no date)

1) Easy to understand, interpret and visualize the trees.

2) It does not require much data processing and preparation

3) Ability to handle categorical as well as numerical data.

4) The problems having multiple outputs can be easily resolved.

5) In case the assumptions are not properly followed by the real model, also performs well.

6) Decision trees proved to be highly significant in tackling decision-based problems.
7) Missing values can be effectively handled by decision trees.
8) In comparison to other techniques such as Random forests, decision trees possess a fast training procedure.
9) The need for feature scaling is not the issue in decision trees.
10) The decision tree employs the white box model which makes the model easier to be observed.
11) It is easy to apply statistical tests with decision trees to validate their significance as well as reliability.
12) Decision trees can be combined to make an ensemble of trees and achieve better performance.

- **Disadvantages of Decision Trees**

1) These are known as weak learners as a small amount of change in training data can very much alter the performance of the model.
2) Overfitting is the main problem that can generate faulty results.
3) A very small amount of added noise can change the results of decision tress drastically.
4) Sometimes, these trees generate more complex hence and make the generalization and interpretation very difficult.
5) In decision trees, some of the concepts are very much hard to learn and apply.
6) Decision trees are not fit for the applications where large datasets are involved as they made the tree complex.
7) If there occurs domination of some classes, the learners of the decision tree generate biased trees.
8) Decision trees are not stable and allow large deviations among results that can lead to wrong predictions and more classification errors.

3.3.4.2 Linear regression (LR)

When modelling the relationship between a scalar answer and one or more explanatory variables in statistics, linear regression is a linear method. There are different types of regression analysis including linear, logistic, polynomial, stepwise, ridge, lasso, and elastic net (Priya, 2021). Each of these models has certain applicability characteristics and can be used under different problem scenarios. Out of all the types, linear regression is the most simplest and popular one which is further divided into two subtypes i.e. simple linear regression and multiple linear regression. In simple linear regression, there is only one independent variable while in multiple regression; there are two or more independent variables. In other words, one of its forms is known as univariate regression analysis when the fundamental connection under investigation only includes the reliant elements and one self-determining element i.e. have one x variable and also one y variable. Another form of linear regression analysis is known as multiple regression analysis when the fundamental connection under investigation involves the reliant element and two or more self-determining elements i.e. have one y variable and many x variables (Hanley, 2016). In nutshell, when there is only one variable that is helpful for prediction, simple linear regression is employed; when there are several explanatory factors, multiple linear regressions are utilized. Let us explain the concept of simple and multiple regressions one by one with the help of figures and equations.

- **Simple Linear Regression (SLR)**

Consider the data points that belong to some dataset. In this type of regression, the independent variable is drawn on the x-axis while the dependent variable is drawn on the y-axis as plotted in Figure 3.9.

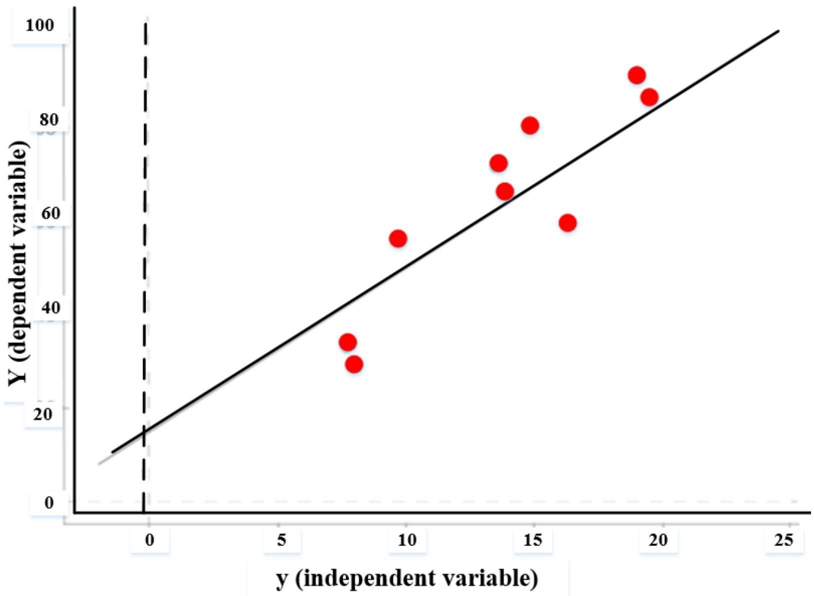
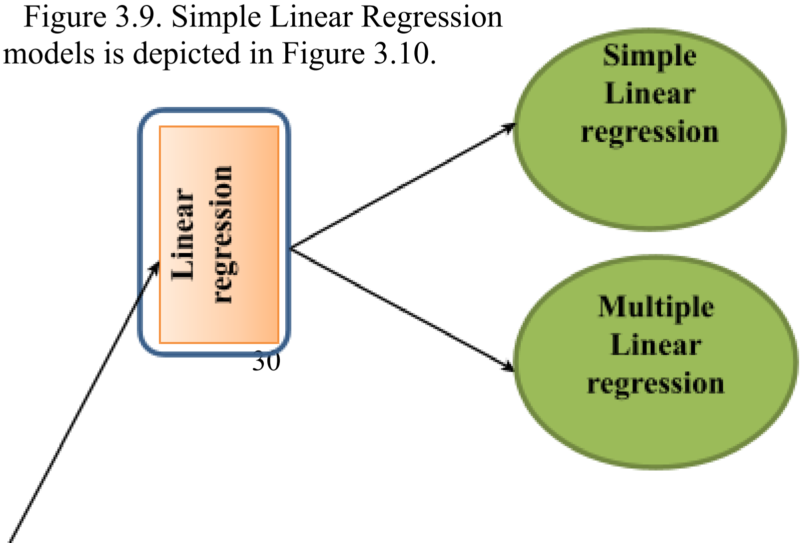


Figure 3.9. Simple Linear Regression

Various types of regression models is depicted in Figure 3.10.



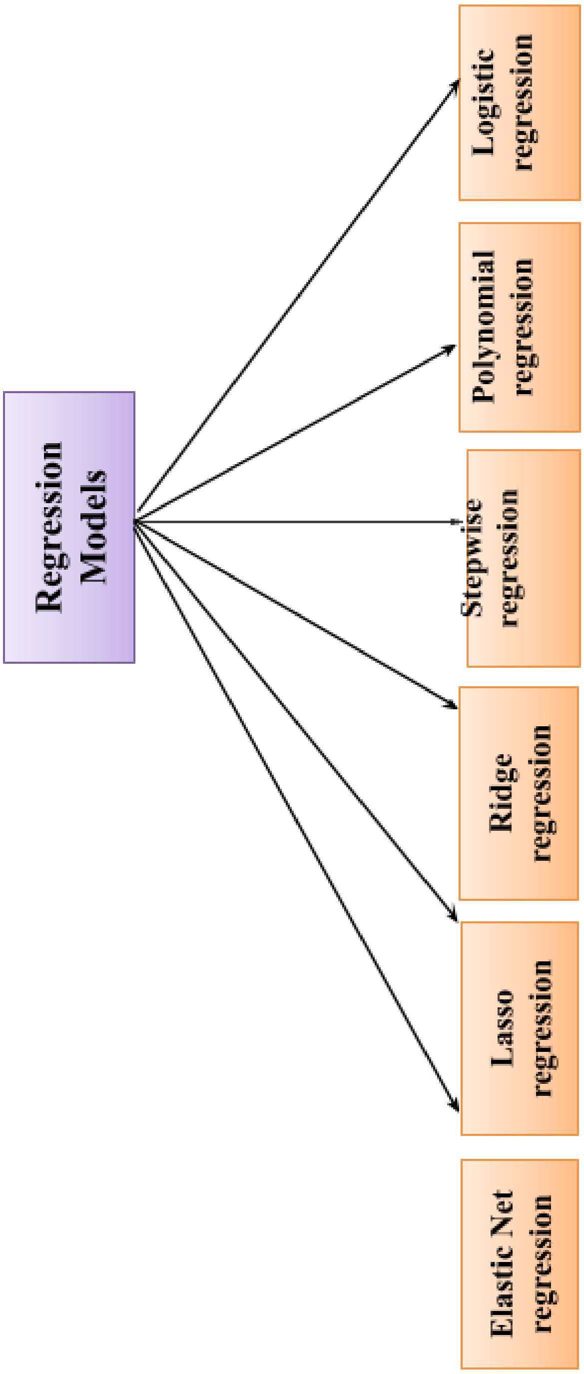


Figure 3.10. Different Types of regression Models (Priya, 2021)

The straight line is drawn in Figure 3.9. represents the relation between dependent and independent variables and known as the line of best fit. When we attain the line of best fit, the model starts prediction. The equation denoting the simple linear regression is given as under

$$z = a_0 + a_1 y \tag{3.4}$$

where a_0 denotes the intercept and a_1 is the slope, and x and y represent independent and dependent variables.

- **Multiple Linear Regression (MLR)**

The basic difference between linear and multiple regressions as given in Figure 3.11 (Shalabh, 2008) is that it involves more than one independent variable. The equation for such type of regression can be given in (3.5)

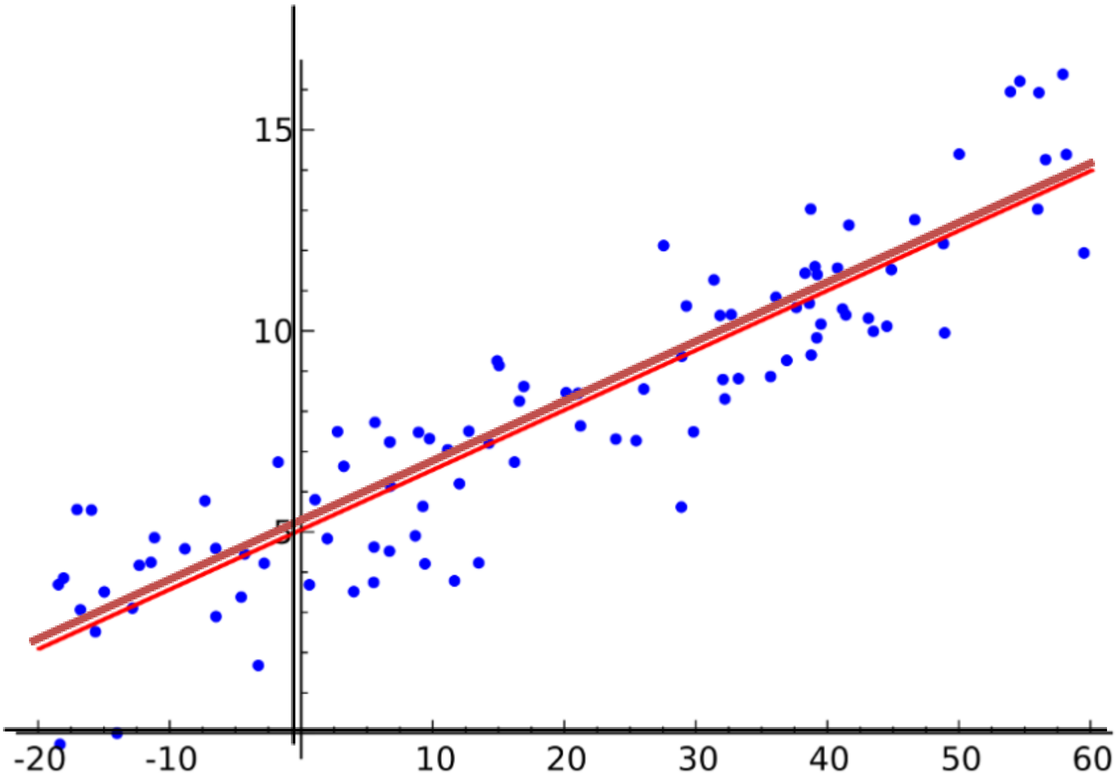


Figure 3.11. Multi Linear Regression

$$z = a_0 + a_1 y_1 + a_2 y_2 + a_3 y_3 + \dots\dots\dots + a_n y_n \tag{3.5}$$

- **Assumptions in Linear Regression**

1) **Linearity:** defines that both dependent and independent variables should be linearly related to each other (Figure 3.12) (Statistics Solutions, 2014).

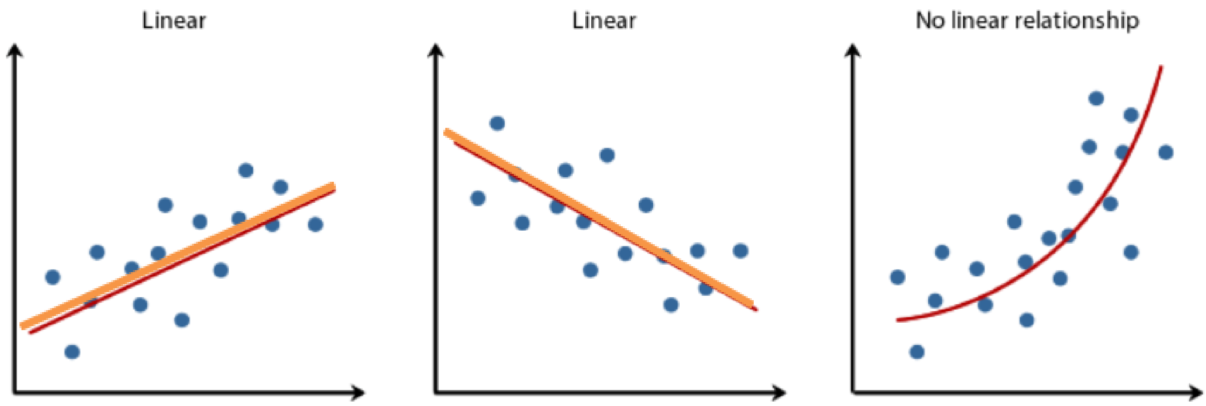


Figure 3.12. Representing linear and non-linear relationships(Statistics Solutions, 2014)

2) **Normality:** states that both x and y variables should follow a normal distribution as shown in Figure 3.13.

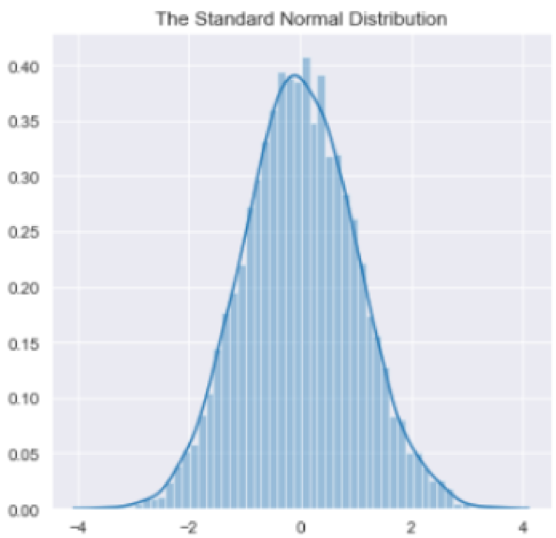


Figure 3.13. Representing a normal distribution

3) **Homoscedasticity:** states that there will be homogeneity in all the residuals across the regression line. All should increase, decrease or remain constant.

4) **Independence property:** All the variables should be independent of each other i.e. there should be no correlation.

5) The error terms should be **normally distributed** and **independent** of each other.

3.3.4.2 Exponential smoothing (ES)

Exponential smoothing (Ostertagia and Ostertag, 2011) is an important technique that is used to smooth the time-series data by applying an exponential window function. This method estimates according to the prior assumption made by the user. To the current observations, the highest weight is allocated, then the proceeding observations are allocated the least weight and earlier observations are allocated with less weight and the process continues. In other words, the newest observation gets the largest weight and the oldest one gets the smallest weight in this process. In the past years, ES has been used as the most popular method for forecasting purposes. Mathematically, this method can be defined as under

$$e_t = \alpha y_t + (1 - \alpha)e_{t-1} = e_{t-1} + \alpha(y_t - e_{t-1}) \qquad (3.6)$$

where e_t denotes the statistic which is smoothed, e_{t-1} is the statistic that is previously smoothed, α represents the smoothing factor and t is the period. The highest value of this smoothing factor results in a reduced smoothing level.

- **Types of Exponential Smoothing**

There are three types of (*Exponential Smoothing Methods for Time Series Forecasting*, no date) Exponential Smoothing which are briefly given one by one as under

1) **Simple or Single Exponential Smoothing (SES):** This method is mostly utilized for data having no trend as well as no seasonal pattern. It makes use of weighted moving averages having decreased weights in an exponential manner (equation 3.6).

2) **Double Exponential Smoothing (DES):** This method is also referred to as Holt's trend. Many researchers also named it exponential smoothing in the second order. This method is utilized mostly when the data possess a linear trend and no seasonal pattern.

3) **Triple Exponential Smoothing (TES):** When the application of smoothing is performed three times, it is known as Triple Exponential Smoothing. This method is utilized mostly when the time-series data possess both linear trends as well as seasonal patterns. The triple Exponential Smoothing method is also named as Holt-Winters exponential smoothing.

- **Example of Exponential Smoothing**

Let us consider the sale of a book in an exhibition for ten years as provided in Table 3.1 and we have to calculate simple exponential smoothing with α =0.3. So, this method can be solved using the formula provided in equation 3.6 (*Exponential Smoothing Methods for Time Series Forecasting*, no date).

Therefore, once all the machine learning methods are applied, they are comparatively analyzed based on certain performance evaluation measures. The output was generated as the predicted values that were further utilized to make the Ethereum price trend analysis.

Table 3.1. Sales data for 10 years and the respective solution using SES (*What is Exponential Smoothing ? - 3 Types of Exponential Smoothing | Analytics Steps*, no date)

Month	Sales
January	30
February	25
March	35
April	25
May	20
June	30
July	35
August	40
September	30
October	45

…….

3.3.5 Summary

This chapter discusses the detailed research plan which is followed to achieve the purpose of predicting trends in Ethereum prices. The methodology designed or proposed consists of five main stages i.e. Dataset collection, Data Preprocessing, Data Visualization, Modelling, and Output. Each stage is explained in a detailed manner to get an insight into the strategy and techniques which are used in this thesis work. Firstly, data is collected from the online yahoo repository, and then data is preprocessed to make it more meaningful for performing further operations. Afterward, data visualization techniques are applied to display the gathered data graphically and modeling is performed. Three machine learning and data forecast techniques are applied namely Decision trees, Linear regression, and Exponential smoothing. The output produced by each method is then comparatively analyzed to determine the best among them using root mean square error.

Month	Sales	Exponential smooth α =0.3
January	30	30.00
February	25	30.00
March	35	28.50
April	25	30.45
May	20	14.1
June	30	15.87
July	35	20.109
August	40	24.5763
September	30	29.20341
October	45	29.442387
November	–	34.1096709

The next chapter explains the implementation and the analysis results obtained for predicting the price trends of Ethereum.

CHAPTER 4

IMPLEMENTATION AND RESULTS

4.1 Introduction

As earlier discussed, in this thesis work, we have implemented three methods i.e. Decision trees, linear regression, and Exponential Smoothing. The purpose was to predict the closing price of Ethereum using these machine learning algorithms. The procedure begins with dataset collection to generate the output and derive the analysis for the market closing price of Ethereum. This chapter discusses the evaluation metrics and methods used to evaluate the efficacy of the models in terms of error. The details regarding the tools, platform, libraries, etc. used for the implementation of the three learning approaches are briefly defined in this chapter. The training and testing data are discussed and the results obtained are presented with the help of graphs. The efficiency of the three models is estimated by comparatively analyzing them as well as with the previous studies.

4.2 Implementation Details

4.2.1 Tools

For the implementation of different techniques, the Python platform is used (*Top 10 Python Libraries - InterviewBit*, no date). Python is a general-purpose, high-level, interpreted programming language. Code readability is prioritized in its design philosophy, which employs heavy indentation. One of the important benefits of using this language is that it is simple to use and very easy to implement as it contains various in-built libraries and functions. Due to its application in various areas, this language has become the most popular language among academicians and researchers. Therefore, it comprises various libraries having different functions. The collection of modules is present in python's library which includes huge snips of code which can be utilized again and again in various programs. To implement a method or something, there is no need the write the identical code several times. These libraries have a significant role in fields like machine learning, data science, data mining, data visualization, etc. Some of the important python libraries used in this work are discussed under (*Libraries in Python - GeeksforGeeks*, no date):

4.2.1.1 Pandas

Pandas (Tutorials Point, 2015) are a Python-based open source data analysis and manipulation tool that is quick, strong, adaptable, and simple to use. It is one of the most popular libraries among data scientists. This library provides high-level data structure with greater flexibility and various crucial tools. The use of this library helps in easy analysis of data, manipulation, and making data cleaner. Various operations can be performed using the Panda library such as iteration, sorting, indexing, data conversions, etc. The following data can be used using this library as

- Dataset's data
- Time series data
- Labeled data of rows and columns matrix
- Unlabeled data
- Statistical data

4.2.1.2 NumPy

The meaning of name NumPy refers to the Numerical python (Shell, 2012). It is a python library that supports faster processing of larger multi-dimensional arrays and matrices, along with a huge collection of mathematical functions especially from the section of linear algebra to apply to any arrays or vectors. With machine learning applications, this library is mostly preferred. It comprises built-in functions of mathematics which make it easier to use for performing computations. The other libraries of python like TensorFlow (Tutorials Point, 2015) also make use of this library to perform various tasks on tensors internally. Some of the features of NumPy are as:

- User-friendly and very interactive library.
- Allows easy implementation of complex mathematical functions and equations.
- Easy to understand and code.
- Array interface facility.

4.2.1.3 Matplotlib

Matplotlib (Data, 2022) is a graph plotting library of Python and its extension NumPy. For integrating charts into programs utilizing all-purpose Graphical user interface (GUI) toolkits like Tkinter, wxPython, Qt, or GTK, it offers an object-oriented API. These libraries plot numerical data, therefore, mostly used for data analysis tasks. Matplotlib is open source and capable of producing high-definition figures such as graphs, histograms, etc.

4.2.1.4 Scikit-Learn

For Python, Scikit (Wang and Lu, 2018) is a free and open-source machine learning package. Among the clustering, regression, and classification methods it provides support to various algorithms such as support vector machines, random forests, gradient boosting, k-means, etc. It is also designed to operate with Python's scientific and numerical libraries, NumPy and SciPy. This library is most popular for providing music-related good suggestions on Spotify.

4.2 Evaluation Metrics

Performance evaluation metrics play a vital role to evaluate the efficacy of a model in correctly completing a classification or regression task. These measures simply tell how well the model has performed in terms of various metrics. In this work, the problem was related to time series data for which the regression approaches need to be utilized. Therefore, one of the most important measures i.e. Root mean square error (RMSE) (scikit-learn developers, 2017) was used as the loss function to determine the performance of three machine learning algorithms concerning error rate. The model which will give less RMSE value is supposed to be performed well and vice-versa. The standard deviation of the predicted errors is referred to RMSE. In other words, it is the difference between the model's predicted values and the original values. Mathematically, RMSE can be given as under:

$$\text{RMSE} = \sqrt{\frac{\sum_{tm=1}^{Tm} (\mathrm{y}_{1,tm} - y_{2,tm})^2}{Tm}} \tag{4.1}$$

Here, in the above equation, $y_{1,tm}$ and $y_{2,tm}$ are the two-time series. T_m represents the time.

4.3 Results

The results are generated and analyzed for three models i.e. Decision trees, Exponential smoothing, and linear regression one by one. The entire data is split into testing and training. For training purposes, we have kept 1120 data points and 30 points are kept for testing purposes. This subsection provides the output obtained for each model and makes analysis based on the results obtained. All the code snippets are provided in the Appendix section.

- **Linear Regression Results**

Firstly linear regression is applied to make the predictions regarding the closing price of Ethereum. The data from the dataset used as discussed in the previous chapter is fed as the input to the classifier or predictor to predict the price. Table 4.1 shows the testing data and the last column represents the predictions made by the model on the closing price of Ethereum.

Based on the testing performed on the data, as shown above, the graphs are plotted for test and train operation among the original closing price values and the predicted one, presented in Figures 4.1 and 4.2. In Figure 4.1, the blue line indicates the closed price and the orange line denotes the predicted price. The data in the figure reveals that the predicted price is far from the closing price and large error. In nut nutshell, it indicates a bad prediction. Similarly, In Figure 4.2, which represents training data, has shown that the predicted price results are not closer to the original closing price as the line doesn't fit properly, Therefore, in the case of linear regression, the results obtained were not satisfactory as it included more number of wrong predictions that can lead to large error.

Table 4.1. Test predictions made by the model on closing price of Ethereum

Date	High	Low	Open	Close	Volume	Adj Close	preds
2020-12-03	622.452698	588.346375	598.459229	616.708740	16146190946	616.708740	515.089681
2020-12-04	618.983154	569.283508	616.722778	569.354187	16337589997	569.354187	514.752287
2020-12-05	596.595459	563.106628	569.347656	596.595459	13498010566	596.595459	514.414894
2020-12-06	606.791931	584.411743	596.568665	601.908997	11290893016	601.908997	514.077501
2020-12-07	602.917908	585.428650	601.797119	591.843384	10720480962	591.843384	513.740107
2020-12-08	594.751587	552.469238	591.900818	554.827759	14398919320	554.827759	513.402714
2020-12-09	577.288391	532.998413	554.792908	573.479126	15855915840	573.479126	513.065321
2020-12-10	574.600159	549.784058	573.504028	559.678528	11672582040	559.678528	512.727927
2020-12-11	560.376709	537.811646	559.679199	545.797363	11098819124	545.797363	512.390534
2020-12-12	573.339417	545.245605	545.578552	568.567322	8534557897	568.567322	512.053141
2020-12-13	593.781250	564.565979	568.609863	589.663208	9070377862	589.663208	511.715747
2020-12-14	590.492981	577.118408	589.782471	586.011169	8125837102	586.011169	511.378354
2020-12-15	596.247742	580.628784	586.021790	589.355591	9326645840	589.355591	511.040961
2020-12-16	636.640320	582.039124	589.378662	636.181824	15817248373	636.181824	510.703567

2020-12-17	673.834229	628.749390	636.154175	642.868958	25479532147	642.868958	510.366174
2020-12-18	662.699097	632.356079	642.916992	654.811951	15756303983	654.811951	510.028781
2020-12-19	668.769592	646.616211	654.624207	659.297913	12830893778	659.297913	509.691387
2020-12-20	659.923706	625.014465	659.185059	638.290833	13375855442	638.290833	509.353994
2020-12-21	646.846558	600.836060	638.315186	609.817871	14419493621	609.817871	509.016601
2020-12-22	635.076599	589.552002	609.420532	634.854187	14745890080	634.854187	508.679207
2020-12-23	637.122803	560.364258	634.824585	583.714600	15261413038	583.714600	508.341814
2020-12-24	613.815186	568.596375	584.135620	611.607178	14317413703	611.607178	508.004421
2020-12-25	633.061401	605.424438	611.554565	626.410706	13520927700	626.410706	507.667027
2020-12-26	650.721436	617.402100	626.498047	635.835815	14761125202	635.835815	507.329634
2020-12-27	711.393555	628.334961	635.887146	682.642334	26093552821	682.642334	506.992241
2020-12-28	745.877747	683.205811	683.205811	730.397339	24222565862	730.397339	506.654847
2020-12-29	737.952881	692.149414	730.358704	731.520142	18710683199	731.520142	506.317454
2020-12-30	754.303223	720.988892	731.472839	751.618958	17294574210	751.618958	505.980061
2020-12-31	754.299438	726.511902	751.626648	737.803406	13926846861	737.803406	505.642667
2021-01-01	749.201843	719.792236	737.708374	730.367554	13652004358	730.367554	505.305274

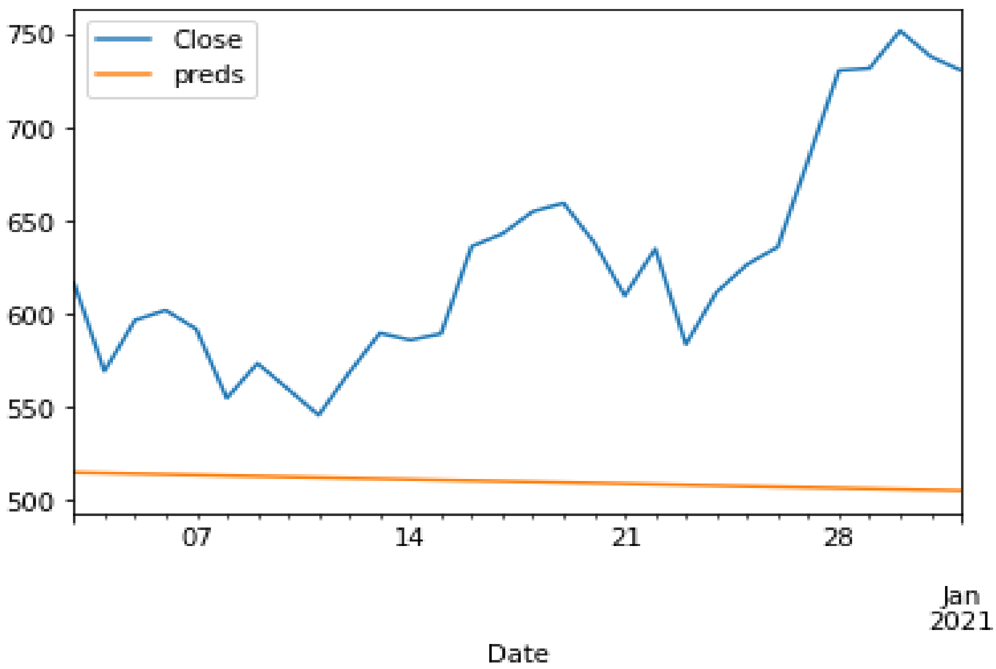


Figure 4.1. Closed and predicted price on test data

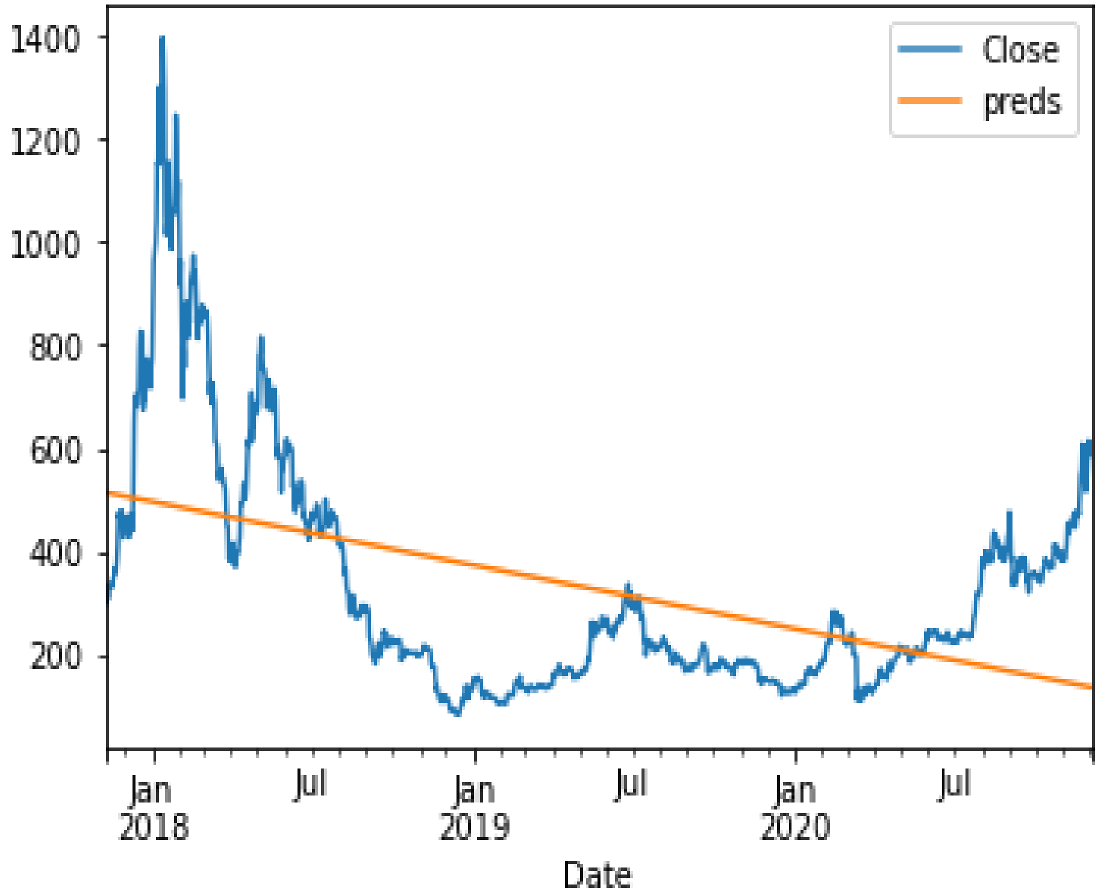


Figure 4.2. Closed and predicted price on train data

Further, the analysis for total price prediction is made as shown in Figures 4.3 and 4.4. using SMA and EWMA. These are the two types of averaging methods which are used in this work and are explained in the previous chapter. In Figure 4.3, we have taken the mean of the previous 1-month data and 2-week data (SMA). The results are plotted for closing price and price for 1 month-SMA, 2 week-SMA, and closing price of Ethereum. The output reflects the close association of all these prices indicating good prediction using LR. The lines for all the prices i.e orange, blue and green seemed to be close to each other.

Similarly, Figure 4.4 shows the plot between the closing price and the price computed using EWMA. It is analyzed that there is close relation between two prices as both the price values overlapped with each- other. the predicted output using EWMA in the LR method is somewhat similar to the closing price values of Ethereum. The lines for all the prices i.e orange and blue seemed to be close to each other which is an indication of good prediction. The data was considered from the year Jan 2018 to Jan 2021.

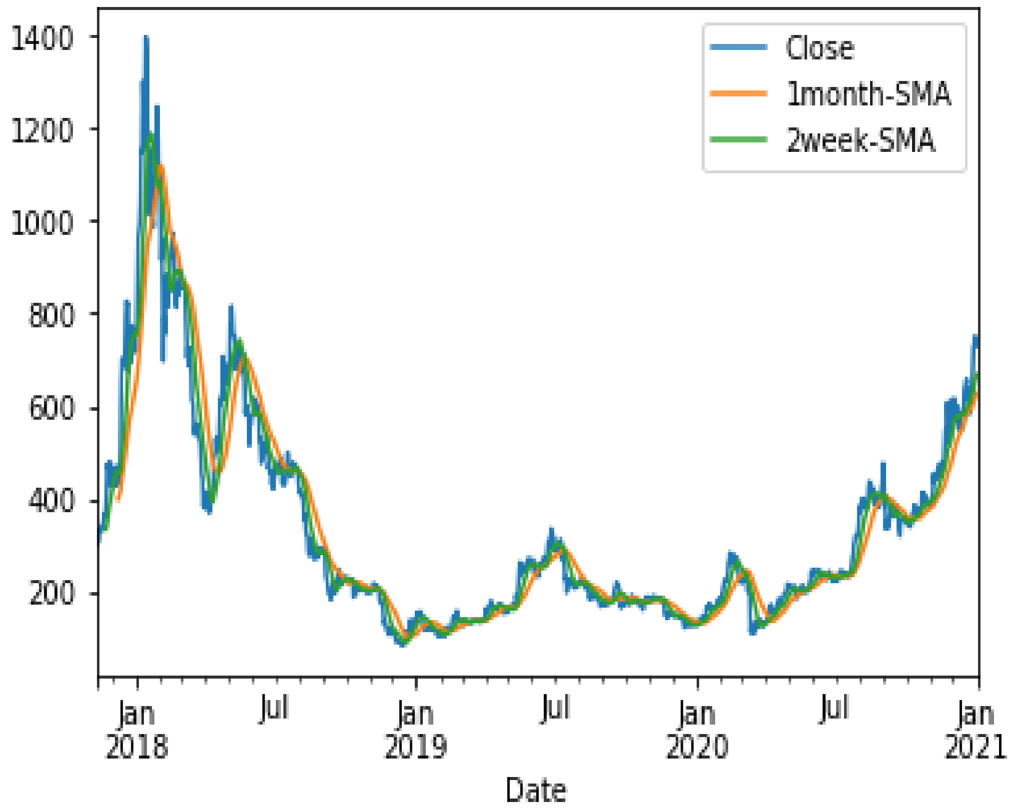


Figure 4.3. Prices: Closing, 1 month-SMA, 2 week-SMA of Ethereum

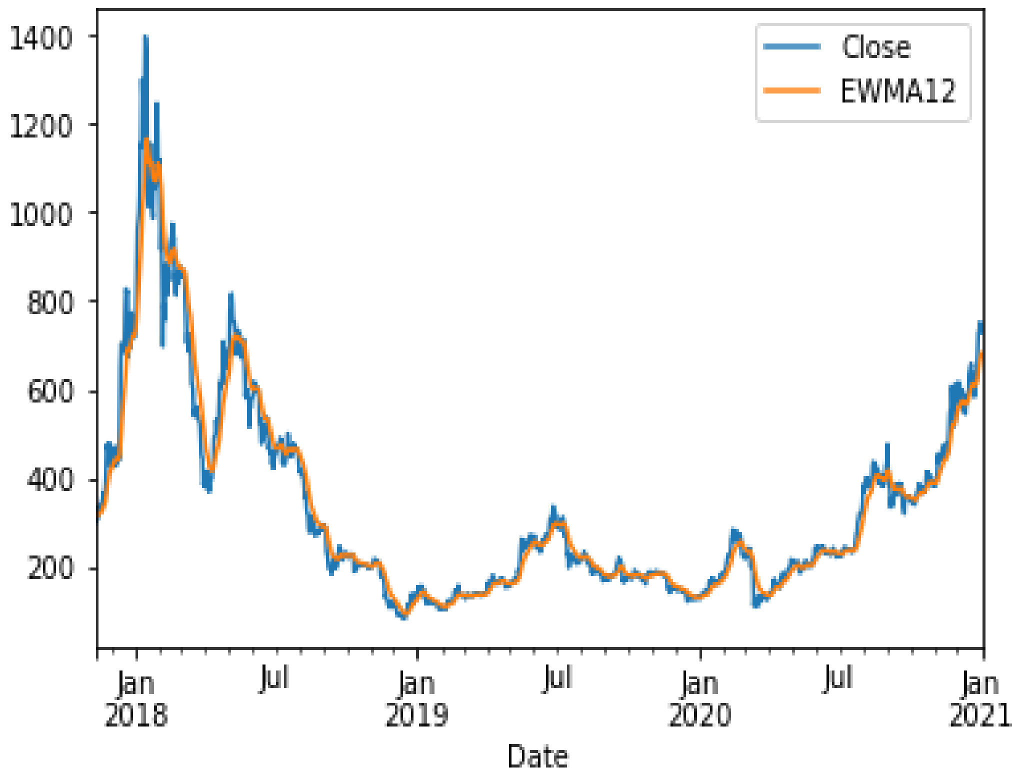


Figure 4.4. Plot between closing price and the price computed using EWMA

- **Exponential Smoothing Results**

After linear regression, Exponential smoothing is employed to make the predictions concerning the closing price of Ethereum. A snippet of code applying the same is shown in Appendix section. Table 4.2 denotes the test data prediction values of EWM and SES for the closing price of Ethereum. Table 4.3 presents the data from the entire dataset used for performing testing and the prediction results. Similarly, the model is evaluated against the test set and the results obtained are shown in Table 4.4.

Table 4.2. Test data prediction values of EWM and SES

Date	High	Low	Open	Close	Volume	Adj Close	EWMA14	SES14
2017-11-09	329.451996	307.056000	308.644989	320.884003	893249984	320.884003	320.884003	320.884003
2017-11-10	324.717987	294.541992	320.670990	299.252991	885985984	299.252991	317.999868	317.999868
2017-11-11	319.453003	298.191986	298.585999	314.681000	842300992	314.681000	317.557352	317.557352
2017-11-12	319.153015	298.513000	314.690002	307.907990	1613479936	307.907990	316.270770	316.270770
2017-11-13	328.415009	307.024994	307.024994	316.716003	1041889984	316.716003	316.330135	316.330135

Table 4.3. Ethereum Price Prediction results on test set using ES

Date	High	Low	Open	Close	Volume	Adj Close	preds
2020-12-03	622.452698	588.346375	598.459229	616.708740	16146190946	616.708740	515.089681
2020-12-04	618.983154	569.283508	616.722778	569.354187	16337589997	569.354187	514.752287
2020-12-05	596.595459	563.106628	569.347656	596.595459	13498010566	596.595459	514.414894

2020-12-06	606.791931	584.411743	596.568665	601.908997	11290893016	601.908997	514.077501
2020-12-07	602.917908	585.428650	601.797119	591.843384	10720480962	591.843384	513.740107
2020-12-08	594.751587	552.469238	591.900818	554.827759	14398919320	554.827759	513.402714
2020-12-09	577.288391	532.998413	554.792908	573.479126	15855915840	573.479126	513.065321

2020-12-10	574.600159	549.784058	573.504028	559.678528	11672582040	559.678528	512.727927
2020-12-11	560.376709	537.811646	559.679199	545.797363	11098819124	545.797363	512.390534
2020-12-12	573.339417	545.245605	545.578552	568.567322	8534557897	568.567322	512.053141
2020-12-13	593.781250	564.565979	568.609863	589.663208	9070377862	589.663208	511.715747
2020-12-14	590.492981	577.118408	589.782471	586.011169	8125837102	586.011169	511.378354
2020-12-15	596.247742	580.628784	586.021790	589.355591	9326645840	589.355591	511.040961
2020-12-16	636.640320	582.039124	589.378662	636.181824	15817248373	636.181824	510.703567
2020-12-17	673.834229	628.749390	636.154175	642.868958	25479532147	642.868958	510.366174
2020-12-18	662.699097	632.356079	642.916992	654.811951	15756303983	654.811951	510.028781
2020-12-19	668.769592	646.616211	654.624207	659.297913	12830893778	659.297913	509.691387
2020-12-20	659.923706	625.014465	659.185059	638.290833	13375855442	638.290833	509.353994
2020-12-21	646.846558	600.836060	638.315186	609.817871	14419493621	609.817871	509.016601
2020-12-22	635.076599	589.552002	609.420532	634.854187	14745890080	634.854187	508.679207
2020-12-23	637.122803	560.364258	634.824585	583.714600	15261413038	583.714600	508.341814
2020-12-24	613.815186	568.596375	584.135620	611.607178	14317413703	611.607178	508.004421
2020-12-25	633.061401	605.424438	611.554565	626.410706	13520927700	626.410706	507.667027
2020-12-26	650.721436	617.402100	626.498047	635.835815	14761125202	635.835815	507.329634
2020-12-27	711.393555	628.334961	635.887146	682.642334	26093552821	682.642334	506.992241
2020-12-28	745.877747	683.205811	683.205811	730.397339	24222565862	730.397339	506.654847
2020-12-29	737.952881	692.149414	730.358704	731.520142	18710683199	731.520142	506.317454
2020-12-30	754.303223	720.988892	731.472839	751.618958	17294574210	751.618958	505.980061
2020-12-31	754.299438	726.511902	751.626648	737.803406	13926846861	737.803406	505.642667
2021-01-01	749.201843	719.792236	737.708374	730.367554	13652004358	730.367554	505.305274

Table 4.4. Prediction results against test set using ES

2020-12-03	601.177362
2020-12-04	607.256754
2020-12-05	610.700403
2020-12-06	612.262239
2020-12-07	622.475890
2020-12-08	627.561684
2020-12-09	628.231427
2020-12-10	629.562081
2020-12-11	633.839776
2020-12-12	645.502148
2020-12-13	656.185533
2020-12-14	670.686811
2020-12-15	672.621902
2020-12-16	686.265110
2020-12-17	689.459548
2020-12-18	696.431691
2020-12-19	700.381035
2020-12-20	702.172225
2020-12-21	713.885739
2020-12-22	719.718375
2020-12-23	720.486470
2020-12-24	722.012529
2020-12-25	726.918398
2020-12-26	740.293376
2020-12-27	752.545603
2020-12-28	769.176376
2020-12-29	771.395634
2020-12-30	787.042331
2020-12-31	790.705868
2021-01-01	798.701862

The code used to visually produce the plots as shown in Figure 4.5 for analyzing the split of training and testing data for the last 30 days is presented in the Appendix section of this thesis. Here, the blue line denotes training, and the orange line indicates testing data. It can be analyzed from the graphical plots that both the lines are monotonically increasing which means that the same price trends are occurring. Similarly, the graph in Figure 4.6 represents the split of the entire dataset into training, testing, and prediction (green line). Therefore, in this figure, one more data i.e. the prediction price data is added to represent the split.

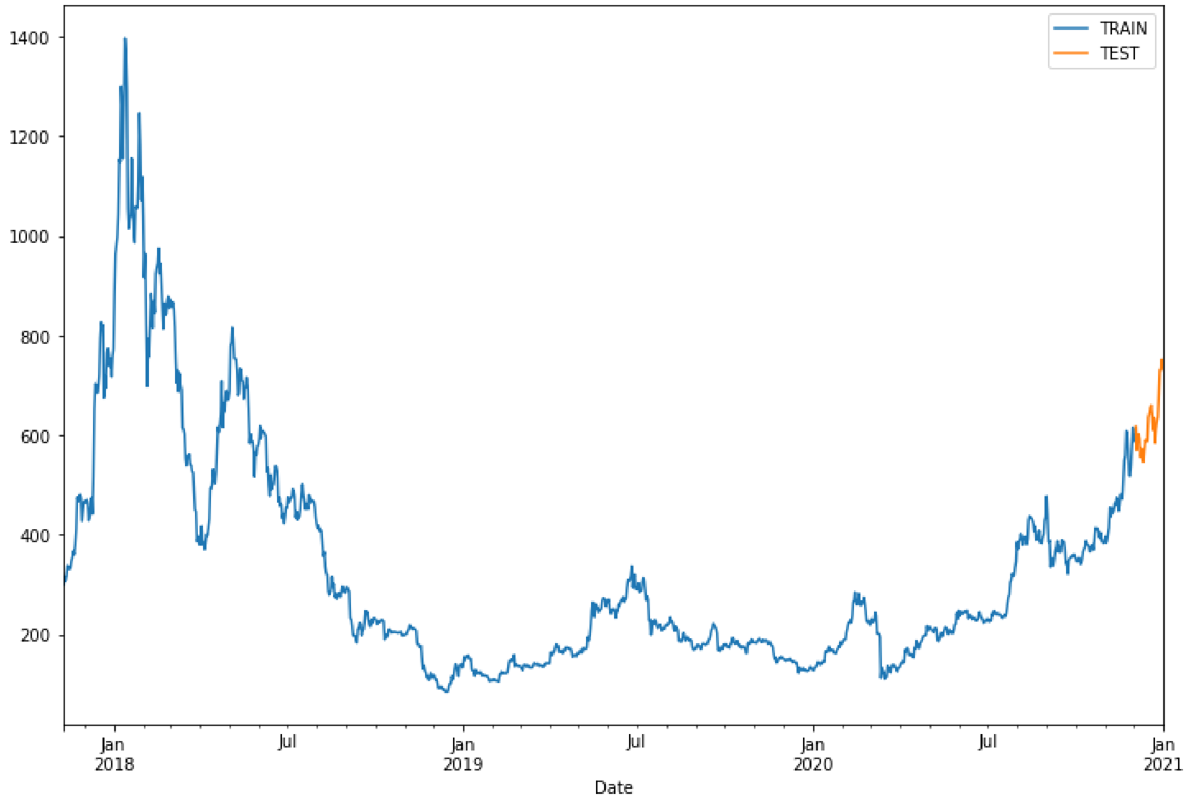


Figure 4.5. Split of training and testing data for the last 30 days

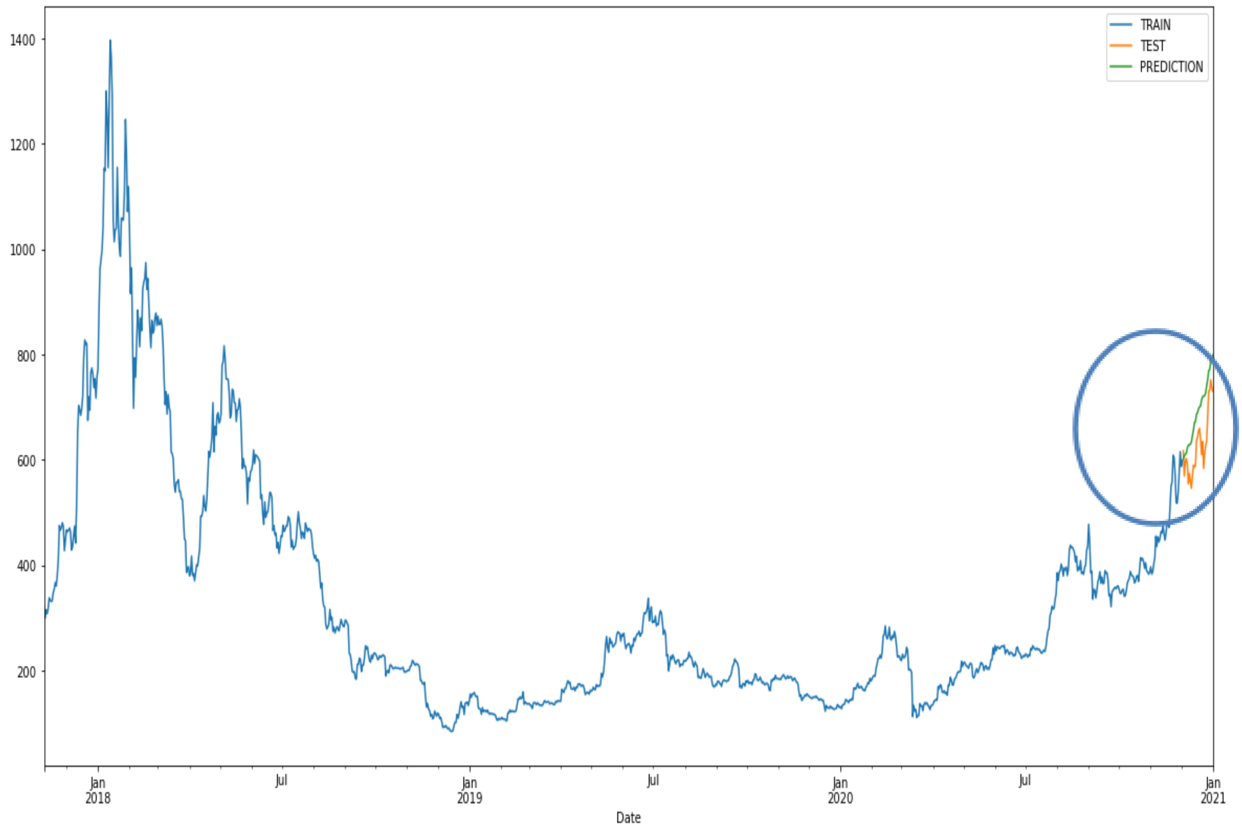


Figure 4.6. Split of entire dataset into training, testing and prediction

The figure below i.e Figure 4.7 is the elaborated view of the circled potion of Figure 4.6 for better understanding and visualization of the splitted data.

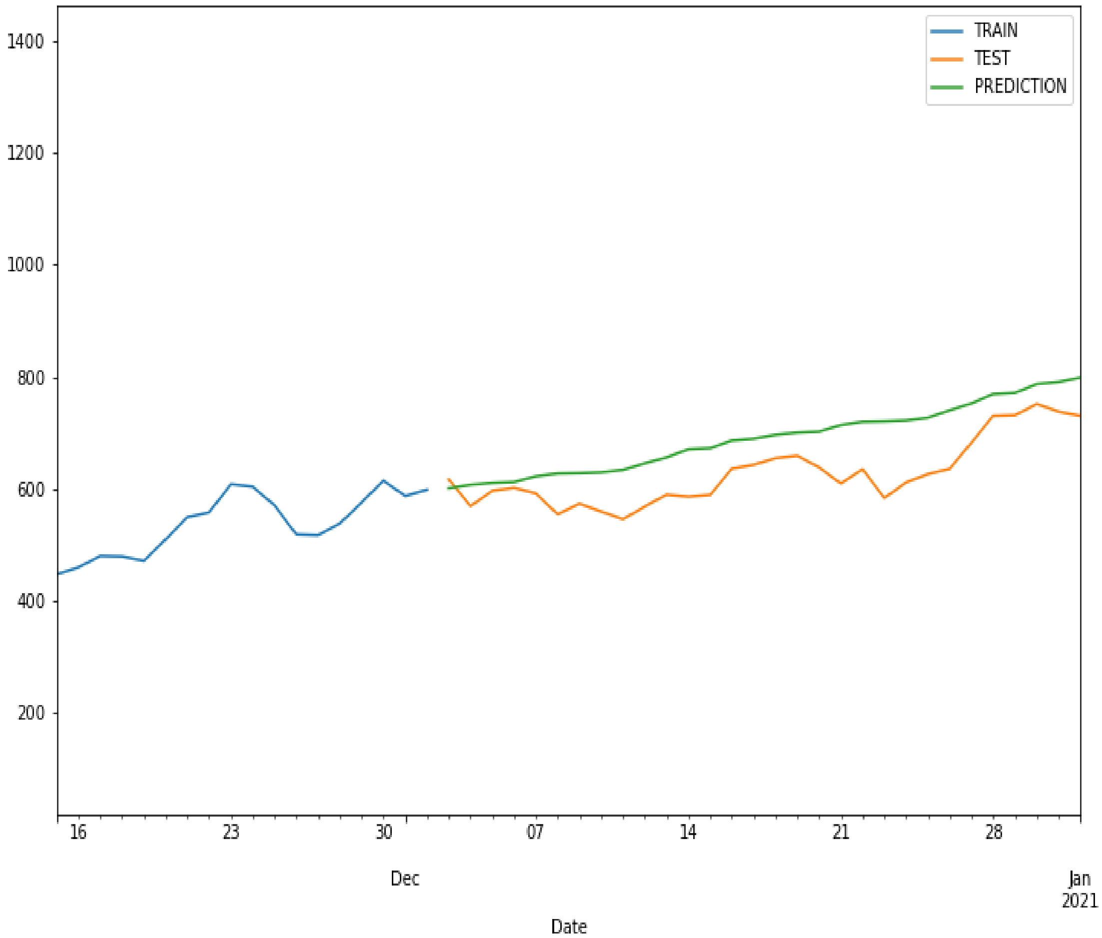


Figure 4.7. Elaborated view of the circled potion of Figure 4.6

Table 4.5. Prediction values of EWM and SES for closing price of Ethereum on training data

Date	High	Low	Open	Close	Volume	Adj Close	EWMA14	SES14
2017-11-09	329.451996	307.056000	308.644989	320.884003	893249984	320.884003	320.884003	320.884003
2017-11-10	324.717987	294.541992	320.670990	299.252991	885985984	299.252991	317.999868	317.999868
2017-11-11	319.453003	298.191986	298.585999	314.681000	842300992	314.681000	317.557352	317.557352
2017-11-12	319.153015	298.513000	314.690002	307.907990	1613479936	307.907990	316.270770	316.270770
2017-11-13	328.415009	307.024994	307.024994	316.716003	1041889984	316.716003	316.330135	316.330135
...	...	...	...	...	...	...	...	...

...	...	...	...	...	...	...	...	...
2020-11-28	548.044861	508.125366	517.597351	538.229797	14770243833	538.229797	523.795478	523.795478
2020-11-29	576.602417	531.987549	538.264587	575.758057	15017517758	575.758057	530.723822	530.723822
2020-11-30	615.240540	571.537781	575.757080	614.842529	20276867833	614.842529	541.939650	541.939650
2020-12-01	635.160583	571.753967	615.070312	587.324158	27178964465	587.324158	547.990917	547.990917
2020-12-02	604.022461	578.741028	587.261597	598.352356	16883292129	598.352356	554.705776	554.705776

1120 rows × 8 columns

The table above i.e. Table 4.5 denotes prediction values of EWM and SES for the closing price of Ethereum on training data. The graph obtained in Figure 4.11 reflects that with the increasing closing market price of Ethereum, the model's predicted price is also increasing for all data. Therefore, the analysis results using ES demonstrated a good prediction output for Ethereum as the predicted line properly fits the original line with a minimum error rate.

- **Decision Trees Results**

In the last, after the implication of linear regression and exponential smoothing, the third model namely the decision tree is employed to make the predictions for the closing price of Ethereum. A snippet of code applying the same for the test data is given in Appendix and the respective prediction output is presented in Table 4.6.

Table 4.6. Prediction output using DT for Ethereum price

```
array([[320.88400269, 320.88400269, 320.88400269],
       [299.25299072, 317.99986776, 317.99986776],
       [314.68099976, 317.55735202, 317.55735202],
       [307.9079895 , 316.27077035, 316.27077035],
       [316.71600342, 316.33013476, 316.33013476],
       [337.63101196, 319.17025172, 319.17025172],
       [333.35699463, 321.06181744, 321.06181744],
       [330.92401123, 322.37677662, 322.37677662],
       [332.39401245, 323.71240806, 323.71240806],
       [347.61199951, 326.89902025, 326.89902025],
       [354.38598633, 330.56394906, 330.56394906],
       [366.73001099, 335.38609065, 335.38609065],
       [360.40100098, 338.72141203, 338.72141203],
       [380.65200806, 344.31215817, 344.31215817],
       [410.16598511, 353.09266843, 353.09266843],
       [474.91101074, 369.33511407, 369.33511407],
       [466.27600098, 382.26056566, 382.26056566],
       [471.32998657, 394.13648844, 394.13648844],  ……..
```

```
[480.35501099, 405.63229145, 405.63229145],
[472.90200806, 414.601587  , 414.601587  ],
[427.52301025, 416.32444343, 416.32444343],
[447.11401367, 420.42971946, 420.42971946],
[466.54000854, 426.57775801, 426.57775801],
[463.44900513, 431.49392429, 431.49392429],
[465.85299683, 436.07513396, 436.07513396],
[470.20401001, 440.62565077, 440.62565077],
[463.28100586, 443.64636478, 443.64636478],
[428.5880127 , 441.6385845 , 441.6385845 ],
[434.4079895 , 440.67450517, 440.67450517],
[456.03100586, 442.72203859, 442.72203859]])
```

The data obtained in the above table is plotted in the form of a graph as given in Figure 4.8. The analysis of test data among predicted and the closing price of Ethereum has shown no relationship to each other. Both the blue and orange lines demonstrated dissimilar trends. One showed an increase in the price and the other reflected decrease in the closing price. Therefore, no correction has been found and the model provided a bad prediction output concerning the closing market price of Ethereum.

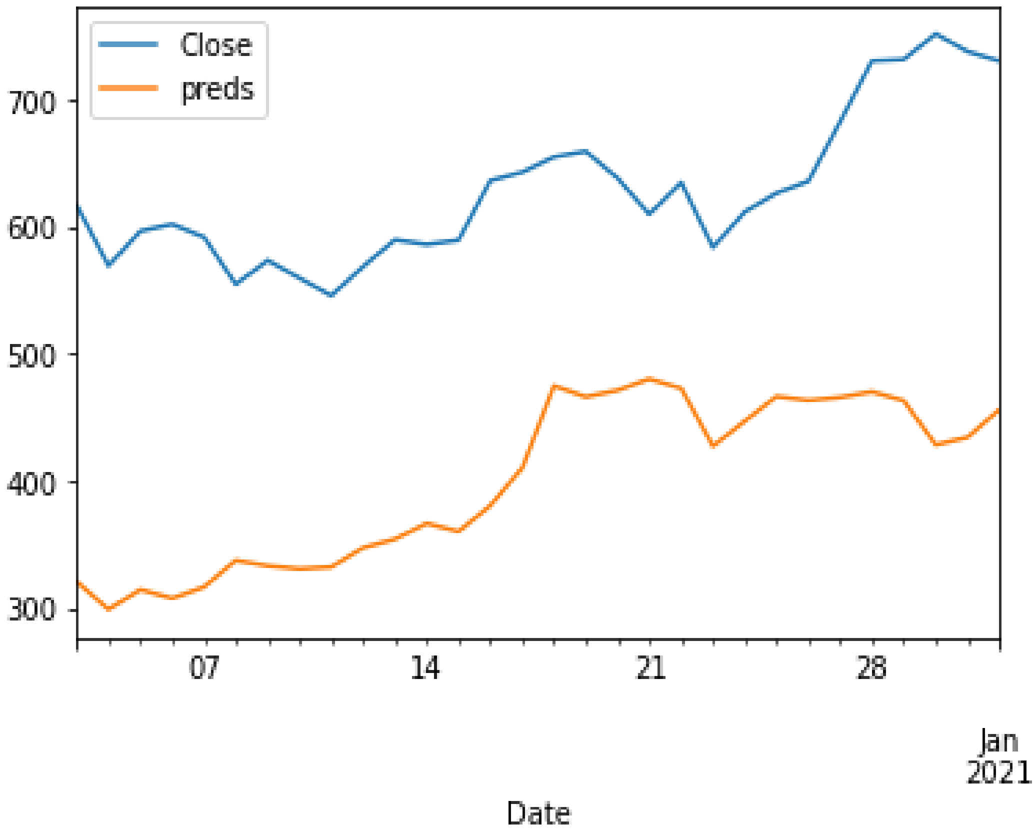


Figure 4.8. Close and predict plot on test data using DT

Further, the analysis of training data among predicted and the closing price of Ethereum has shown the serious issue shown in Figure 4.9. The training data (depicted in the orange line) for the predicted price seemed to overfit the original closing price. Overfitting is the issue in which the model is not able to make accurate predictions. In this problem, the model is only capable of making correct predictions for the data only on it is trained and for the unknown data, the model doesn't give good prediction results. Therefore, in our work, DT gets pruned to an overfitting problem that resulted in declined output with more error rate.

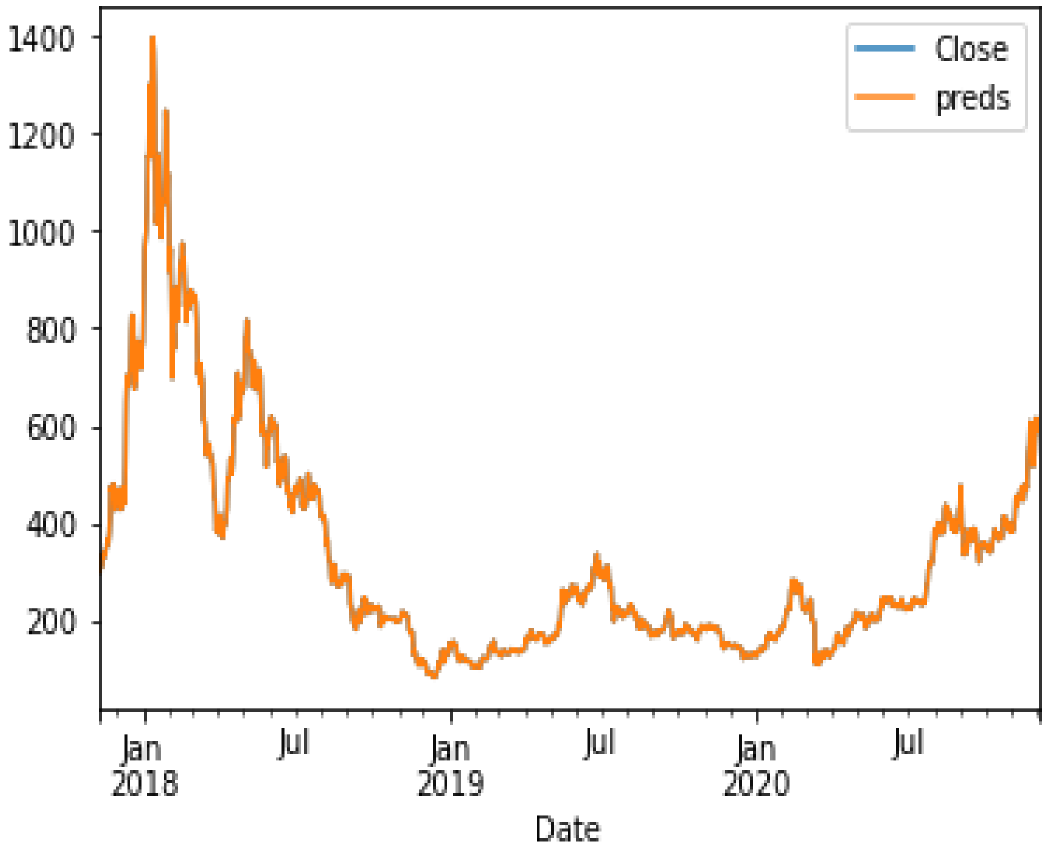


Figure 4.9. Close and predict plot on test data using DT

The table above i.e. Table 4.7 represents the test data for original closing price values and the predictions obtained after applying DT model.

4.5 Comparative Analysis of Models

Once the prediction results for all the models are obtained, they were graphically analyzed. To pick up the best performing machine learning algorithm among all the applied models, these are comparatively evaluated using an important parameter i.e. root mean square error (RMSE). The error for each of the models is determined the analysis of the results is made. The results obtained reveal that for the decision tree, Exponential smoothing, and linear regression, RMSE

Table 4.7. Test data for closing price values and the predictions using DT

Date	Close	preds
2020-12-03	616.708740	320.884003
2020-12-04	569.354187	299.252991
2020-12-05	596.595459	314.681000
2020-12-06	601.908997	307.907990
2020-12-07	591.843384	316.716003
2020-12-08	554.827759	337.631012
2020-12-09	573.479126	333.356995
2020-12-10	559.678528	330.924011
2020-12-11	545.797363	332.394012
2020-12-12	568.567322	347.612000
2020-12-13	589.663208	354.385986
2020-12-14	586.011169	366.730011
2020-12-15	589.355591	360.401001
2020-12-16	636.181824	380.652008
2020-12-17	642.868958	410.165985
2020-12-18	654.811951	474.911011

2020-12-19	659.297913	466.276001
2020-12-20	638.290833	471.329987
2020-12-21	609.817871	480.355011
2020-12-22	634.854187	472.902008
2020-12-23	583.714600	427.523010
2020-12-24	611.607178	447.114014
2020-12-25	626.410706	466.540009
2020-12-26	635.835815	463.449005
2020-12-27	682.642334	465.852997
2020-12-28	730.397339	470.204010
2020-12-29	731.520142	463.281006
2020-12-30	751.618958	428.588013
2020-12-31	737.803406	434.407990
2021-01-01	730.367554	456.031006

is found to be 15.098, 7.947, and 10.856 respectively as tabulated in Table 4.8. For easy visualization, these results are also shown with the help of Figure 4.10. Therefore, the highest error seemed to have occurred when using decision trees followed by linear regression. The lowest error is achieved in the case of Exponential smoothing which makes it the best and the most effective model for making price predictions of Ethereum with the minimum amount of error.

Table 4.8. Comparative results of RMSE for three machine learning models

DecisonTree_RMSE	ExponentialSmoothing_RMSE	LinearRegression_RMSE
15.098976	7.947508	10.8565

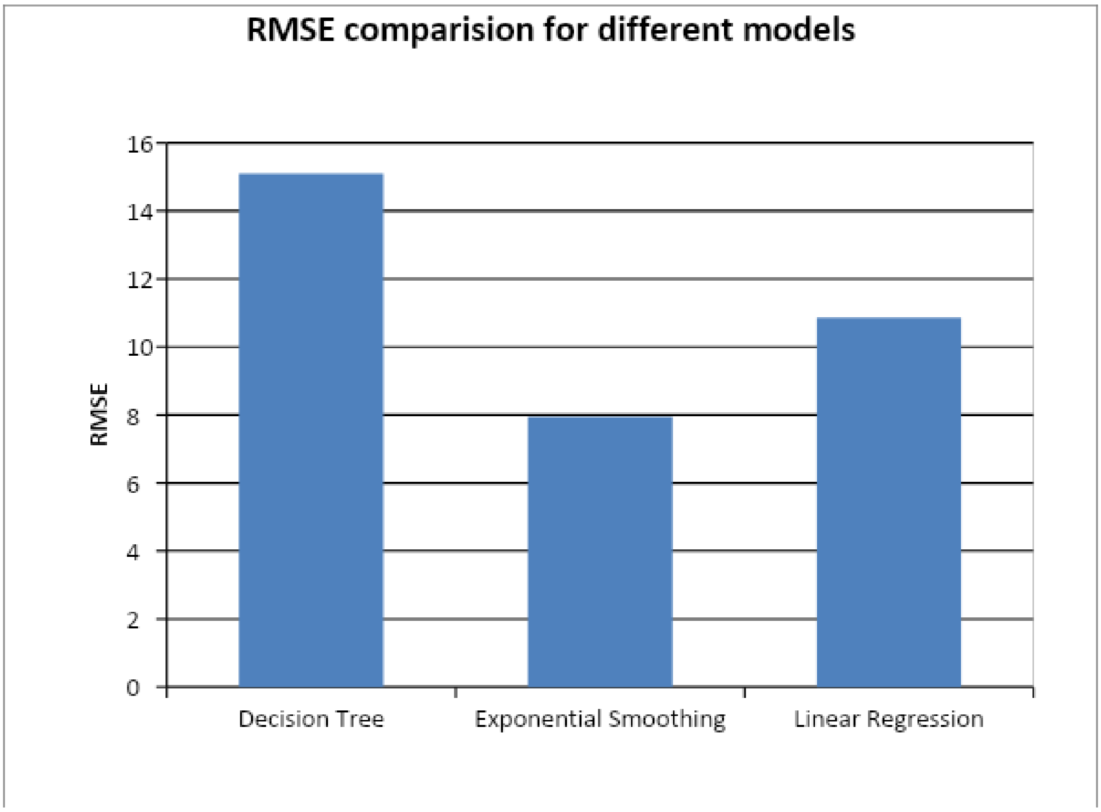


Figure 4.10 Comparative graphical analysis of three machine learning models based on RMSE

4.6 Summary

This chapter briefly provides information about the platform and the tools used for the implementation of machine learning models. The performance evaluation measure used to evaluate the performance of three models i.e. Decision trees, linear regression, and Exponential smoothing is discussed. Further, the implementation results for the prediction of trends in the closing price of Ethereum using each algorithm are discussed in detail one by one for better understanding. All the results on both test and train data are explained graphically with the help of plots for easy visualization and interpretation to make an effective analysis. At last, the performance of all three models is evaluated using RMSE which indicates Exponential smoothing s the best technique to achieve the purpose of this thesis work with minimum error in comparison to decision trees and linear regression.

The next chapter provides the conclusion and future recommendations to carry out this research further.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

In recent years, the usage of Cryptocurrency has increased to a larger extent. This platform has attracted a wide range of buyers, companies and merchants to enable payment options in the form of digital currency worldwide. The previous and related data obtained reveals that the acceptance of Cryptocurrencies has skyrocketed. A number of examples are available which is the clear indication of the replacement of the physical currency mechanism with the advanced platform i.e. Cryptocurrency. One of the examples can be seen in Microsoft platform where Cryptocurrency has been adopted for the payment purposes. A patent has been filled by the Visa to enable payments using a Cryptocurrency system with the help of wallets or Visa cards. People are drawn to Cryptocurrencies because of their invisibility and anonymity. The purpose of Cryptocurrency research is to gain a clear picture of the market and determine its value using the acquired data trends.

Cryptocurrencies have been populated among the traders, stakeholder and various companies. To develop decentralized applications with more effectiveness, Ethereum has come forward as the most widely used and popular Cryptocurrency platform. It was invented by Vitalik Buterin in 2013. The concept of blockchain is incorporated in the Ethereum and permits the safe transfer of crucial transactions without utilizing a password. This technology is growing as the greatest invention which has huge hype among developers to create and implicate many decentralized apps.

Cryptocurrencies, no doubt, have become one of the important sources of investors but correctly determining the price of Cryptocurrency such as Ethereum is a very complex task. The price of Ethereum is dependent on a number of aspects including trends in market, high and low drop value, etc. therefore they possess high amounts of volatility and huge fluctuations. In order to tackle the issue of accurately predicting the market trend and price of Ethereum, there is the need for some automated approaches such as a machine learning platform.

In recent years, the machine learning paradigm has been widely utilized for the purpose of predicting future price trends of Ethereum. On the basis of recent information, these algorithms can make predictions for the future forecast trends. Therefore, moving towards this advanced platform of prediction, this thesis work has applied three approaches to attain the purpose of predicting the closing price of Ethereum with high efficacy and low error.

In **Chapter 1**, the brief history of development of Cryptocurrency specially focusing on Ethereum is discussed and the points are highlighted differentiating Ethereum and Bitcoin. Based on the past data, a problem statement is designed and research gaps are identified and discussed. On the basis of the gaps identified, some of the objectives are proposed upon which the entire thesis work is performed to cover the unexplored areas in this field. Next, the scope and significance of the study is given to show the relevance of this type of study. Atlast, the structure of the entire thesis is given.

Chapter 2 discussed the state-of-the-art literature to know the extent of research work already performed in this area related to predicting the price trends of Ethereum. This chapter explored various statistical and machine learning techniques which are analyzed properly to be used in attaining the best results. The analysis of existing research demonstrated the high use of The GRU, LSTM, ANN, SVM, LR, RF, DL, BART based techniques and models to perform the prediction of Ethereum market price prediction. RMSE, MSE, NRMSE, MAE, F1, ROC are the most preferred performance evaluation metrics.

In **Chapter 3**, the detailed research methodology and the plan used for the prediction of trends in Ethereum prices are elaborated. The methodology is designed which comprises different stages including Dataset Acquisition Data Preprocessing, Data Visualization, Modelling and Prediction Outcome. Each stage is discussed in depth to properly show the details of the methods utilized in this work. Initially, data is gathered from yahoo repository which is open source and then data is preprocessed to convert it into from raw to relevant form. Then, data visualization techniques are employed to provide graphical visualization of data by the help of plots and modeling is done after that. Three machine learning techniques i.e. Decision trees, linear regression and Exponential smoothing are applied for prediction of Ethereum price. At last, the comparison of results achieved by all the methods is done using root mean square error to determine the best one.

In **Chapter 4,** the details about the tools used for the implementation is provided. The performance measure considered and applied to test the performance of three models i.e. Decision trees, linear regression and Exponential smoothing is briefly discussed. The results obtained using three models on test and train data are presented graphically by using graphs and plots for easy interpretation of results. At last, the performance of all the applied techniques is evaluated with the help of RMSE. The output obtained shown the highest performance using Exponential smoothing with minimum error as compared to other two methods. Thus the amalgamation of Ethereum price prediction with Exponential smoothing can provide good estimations with high reliability.

Chapter 5 discusses the conclusion of this thesis work summarizing the outcome acheived in every chapter one by one. Further, the future work is provided in the form of points which need to be focused for further research to cover the uncovered aspects.

After the completion of all the chapters, references to various studies chosen from the literature are provided which may be helpful for the researchers to perform further research related to prediction of price trends of Ethereum by the help of machine learning approaches.

5.2 Future Recommendations

Following aspects can be considered in future to precede the research in this field for prediction the price of Ethereum more efficiently, also shown in Figure 5.1:

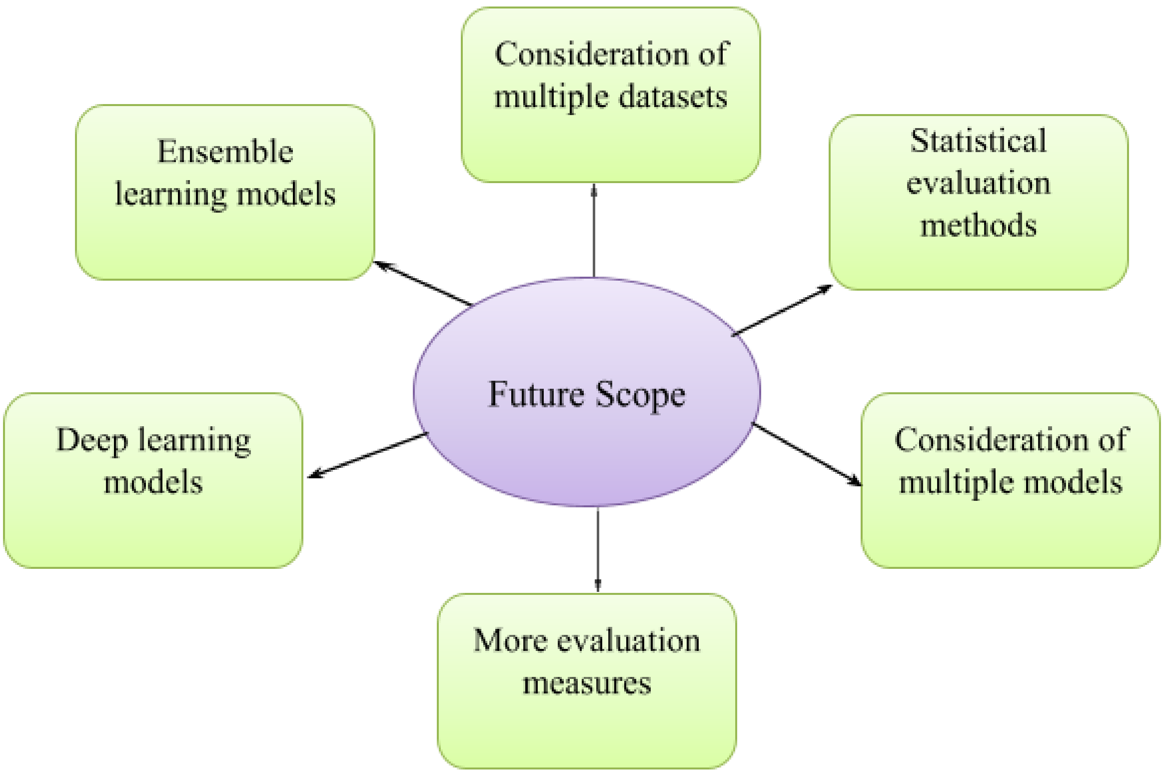


Figure 5.1. Future Recommendations

- Deep learning networks such as CNN, RNN can be used as they require no human intervention.
- Multiple machine learning algorithms can be combined together to produce more accurate results such as ensemble learning platforms.
- Despite considering a single dataset, evaluations can be extended for more datasets also.
- Results of the models can be evaluated using statistical validations to prove their relevance such as Friedman's rank test (FRT), Cross-validations (CV) etc.
- Further comparison can be made by implementing all the models used in previous studies.
- Other performance evaluation measures such as accuracy, sensitivity, specificity, etc. can be considered for more accurate predictions.

REFERENCES

1.10. Decision Trees — scikit-learn 1.1.2 documentation (no date). Available at: https://scikit-learn.org/stable/modules/tree.html (Accessed: 29 August 2022).

Ammer, M.A. (2022) 'Deep Learning Algorithm to Predict Cryptocurrency Fluctuation Prices : Increasing Investment Awareness', pp. 1–22.

An Introduction to Ethereum and Smart Contracts: an Authentication Solution (no date). Available at: https://auth0.com/blog/an-introduction-to-ethereum-and-smart-contracts-part-3/ (Accessed: 29 August 2022).

Berentsen, A. and Wiedmer, J. (2018) 'The Price of Cryptocurrencies: An Empirical Analysis'.

Chai, T. and Draxler, R.R. (2014) 'Root mean square error (RMSE) or mean absolute error (MAE)? -Arguments against avoiding RMSE in the literature', *Geoscientific Model Development*, 7(3), pp. 1247–1250. Available at: https://doi.org/10.5194/gmd-7-1247-2014.

Cioffi, R. *et al.* (2020) 'Artificial intelligence and machine learning applications in smart production: Progress, trends, and directions', *Sustainability (Switzerland)*, 12(2). Available at: https://doi.org/10.3390/su12020492.

Cryptocurrency (Ethereum) Price Prediction Using Python (no date). Available at: https://www.godatainsights.com/post/cryptocurrency-ethereum-price-prediction-using-python (Accessed: 29 August 2022).

Cycles, S. (1989) 'Chapter 9 Chapter 9', *Cycle*, 1897(Figure 1), pp. 44–45. Available at: https://doi.org/10.1007/0-387-25465-X.

Data, P. (2022) 'About the Tutorial Copyright & Disclaimer'.

Decision Tree Algorithm, Explained - KDnuggets (no date). Available at: https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html (Accessed: 29 August 2022).

Derbentsev, V., Matviychuk, A. and Soloviev, V.N. (2020) 'Forecasting of Cryptocurrency Prices Using Machine Learning', *Advanced Studies of Financial Technologies and Cryptocurrency Markets*, (Ml), pp. 211–231. Available at: https://doi.org/10.1007/978-981-15-4498-9_12.

Ethereum (ETH) Price Prediction 2022, 2023, 2025-2030 | PrimeXBT (no date). Available at: https://primexbt.com/for-traders/ethereum-price-prediction-forecast/ (Accessed: 29 August 2022).

Ethereum Data | Kaggle (no date). Available at: https://www.kaggle.com/datasets/varpit94/ethereum-data (Accessed: 29 August 2022).

Exponential Smoothing Methods for Time Series Forecasting (no date). Available at: https://www.encora.com/insights/exponential-smoothing-methods-for-time-series-forecasting (Accessed: 29 August 2022).

Grinberg, R. (2012) 'Bitcoin: An Innovative Alternative Digital Currency Recommended Citation Bitcoin: An Innovative Alternative Digital Currency', *Hastings Science and Technology Law Journal*, 4(1), p. 159.

Hamayel, M.J. and Owda, A.Y. (2021) 'A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms', *Ai*, 2(4), pp. 477–496. Available at: https://doi.org/10.3390/ai2040030.

Hanley, J.A. (2016) 'Simple and multiple linear regression: sample size considerations', *Journal of Clinical Epidemiology*, 79(May), pp. 112–119. Available at: https://doi.org/10.1016/j.jclinepi.2016.05.014.

Jani, S. (2017) 'An overview of ethereum & its comparison with bitcoin', *International Journal of Scientific & Engineering Research*, 10(8), pp. 1–5. Available at: http://www.ijser.org.

Khedr, A.M. *et al.* (2021) 'Cryptocurrency price prediction using traditional statistical and machine-learning techniques: A survey', *Intelligent Systems in Accounting, Finance and Management*, 28(1), pp. 3–34. Available at: https://doi.org/10.1002/ISAF.1488.

Kim, H.-M., Bock, G.-W. and Lee, G. (2019) 'Predicting Ethereum Prices using Machine Learning and Block Chain Information', *undefined* [Preprint]. Available at: https://doi.org/10.14400/JDC.2020.18.11.129.

Kristian, N., Adzikri, F. and Rizkinia, M. (2021) 'Ethereum Price Prediction Comparison Using k-NN and Multiple Polynomial Regression', *17th International Conference on Quality in Research, QIR 2021: International Symposium on Electrical and Computer Engineering*, pp. 141–146. Available at: https://doi.org/10.1109/QIR54354.2021.9716169.

Kumar, D. and Rath, S.K. (2020) 'Predicting the Trends of Price for Ethereum Using Deep Learning Techniques', *Advances in Intelligent Systems and Computing*, 1056, pp. 103–114. Available at: https://doi.org/10.1007/978-981-15-0199-9_9.

Libraries in Python - GeeksforGeeks (no date). Available at: https://www.geeksforgeeks.org/libraries-in-python/ (Accessed: 29 August 2022).

Margret Anouncia, S., Gohel, H.A. and Vairamuthu, S. (2020) 'Data visualization: Trends and challenges toward multidisciplinary perception', *Data Visualization: Trends and Challenges*

Toward Multidisciplinary Perception, (June), pp. 1–179. Available at: https://doi.org/10.1007/978-981-15-2282-6.

Oliveira, V.C. *et al.* (2021) 'Analyzing Transaction Confirmation in Ethereum Using Machine Learning Techniques', *Performance Evaluation Review*, 48(4), pp. 12–15. Available at: https://doi.org/10.1145/3466826.3466832.

Ostertagova, E. and Ostertag, O. (2011) 'The Simple Exponential Smoothing Model', *Modelling of Mechanical and Mechatronic systems*, 1(September 2011), pp. 380–384. Available at: http://www.researchgate.net/publication/256088917_THE_SIMPLE_EXPONENTIAL_SMOOTHING_MODEL.

Pandas DataFrame dropna() Method (no date). Available at: https://www.w3schools.com/python/pandas/ref_df_dropna.asp (Accessed: 29 August 2022).

Politis, A., Doka, K. and Koziris, N. (2021) 'Ether price prediction using advanced deep learning models', *IEEE International Conference on Blockchain and Cryptocurrency, ICBC 2021*, pp. 0–2. Available at: https://doi.org/10.1109/ICBC51069.2021.9461061.

Priya, K.S. (2021) 'Linear Regression Algorithm in Machine Learning through MATLAB', *International Journal for Research in Applied Science and Engineering Technology*, 9(12), pp. 989–995. Available at: https://doi.org/10.22214/ijraset.2021.39410.

Saad, M. and Mohaisen, A. (2018) 'Towards characterizing blockchain-based cryptocurrencies for highly-accurate predictions', *INFOCOM 2018 - IEEE Conference on Computer Communications Workshops*, X(X), pp. 704–709. Available at: https://doi.org/10.1109/INFCOMW.2018.8406859.

scikit-learn developers (2017) 'scikit-learn user guide Release 0.18.2 scikit-learn developers', pp. 1–2050. Available at: https://scikit-learn.org/0.18/_downloads/scikit-learn-docs.pdf.

Senthuran, G. and Halgamuge, M.N. (2020) 'Prediction of Cryptocurrency Market Price Using Deep Learning and Blockchain Information', *Essentials of Blockchain Technology*, pp. 349–364. Available at: https://doi.org/10.1201/9780429674457-15.

Shalabh, I.K. (2008) 'Chapter 3 Multiple Linear Regression Model', pp. 1–41.

Shell, S. (2012) 'An introduction to Numpy and Scipy', pp. 1–24.

Statistics Solutions (2014) 'Assumptions of Linear Regression', *Statistics Solutions*, pp. 1–7. Available at: https://www.statisticssolutions.com/assumptions-of-linear-regression/.

Structure of the decision tree (DT) algorithm. | Download Scientific Diagram (no date).

Available at: https://www.researchgate.net/figure/Structure-of-the-decision-tree-DT-algorithm_fig1_353409254 (Accessed: 29 August 2022).

Sukparungsee, S., Areepong, Y. and Taboran, R. (2020) 'Exponentially weighted moving average—Moving average charts for monitoring the process mean', *PLoS ONE*, 15(2). Available at: https://doi.org/10.1371/journal.pone.0228208.

The Standard Normal Distribution | Examples, Explanations, Uses (no date). Available at: https://www.scribbr.com/statistics/standard-normal-distribution/ (Accessed: 29 August 2022).

Top 10 Python Libraries - InterviewBit (no date). Available at: https://www.interviewbit.com/blog/python-libraries/ (Accessed: 29 August 2022).

Tutorial, S.M.A.T. *et al.* (no date) 'Simple Moving Average (SMA) Strategies [Video] The SMA – Not Always So Simple'.

Tutorials Point (2015) 'About the Tutorial Disclaimer & Copyright', *Organizational Behavior*, pp. 1–45.

Understanding Decision Tree, Algorithm, Drawbacks and Advantages. | by Sagar Rawale | Medium (no date). Available at: https://medium.com/@sagar.rawale3/understanding-decision-tree-algorithm-drawbacks-and-advantages-4486efa6b8c3 (Accessed: 29 August 2022).

Valencia, F., Gómez-Espinosa, A. and Valdés-Aguirre, B. (2019) 'Price movement prediction of cryptocurrencies using sentiment analysis and machine learning', *Entropy*, 21(6). Available at: https://doi.org/10.3390/e21060589.

Vujičić, D., Jagodić, D. and Randić, S. (2018) 'Blockchain technology, bitcoin, and Ethereum: A brief overview', *2018 17th International Symposium on INFOTEH-JAHORINA, INFOTEH 2018 - Proceedings*, 2018-Janua(March), pp. 1–6. Available at: https://doi.org/10.1109/INFOTEH.2018.8345547.

Wang, W. and Lu, Y. (2018) 'Analysis of the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) in Assessing Rounding Model', *IOP Conference Series: Materials Science and Engineering*, 324(1). Available at: https://doi.org/10.1088/1757-899X/324/1/012049.

What is Exponential Smoothing ? - 3 Types of Exponential Smoothing | Analytics Steps (no date). Available at: https://www.analyticssteps.com/blogs/tutorial-exponential-smoothing-and-its-types (Accessed: 29 August 2022).

Zoumpekas, T., Houstis, E. and Vavalis, M. (2020) ‘ETH analysis and predictions utilizing deep learning’, *Expert Systems with Applications*, 162. Available at: https://doi.org/10.1016/J.ESWA.2020.113866.

APPENDIX

Appendix-1:- Dataset Details

This subsection provides the details regarding the dataset used in this research work. Some of the example data for each year is presented as in the following:

- **For year 2017**

Date	Open	High	Low	Close	Adj Close	Volume
11/9/2017	308.645	329.452	307.056	320.884	320.884	893249984
11/10/2017	320.671	324.718	294.542	299.253	299.253	885985984
11/11/2017	298.586	319.453	298.192	314.681	314.681	842300992
11/12/2017	314.69	319.153	298.513	307.908	307.908	1613479936
11/13/2017	307.025	328.415	307.025	316.716	316.716	1041889984
11/14/2017	316.763	340.177	316.763	337.631	337.631	1069680000
11/15/2017	337.964	340.912	329.813	333.357	333.357	722665984
11/16/2017	333.443	336.159	323.606	330.924	330.924	797254016
11/17/2017	330.167	334.964	327.523	332.394	332.394	621732992
11/18/2017	331.98	349.616	327.687	347.612	347.612	649638976
11/19/2017	347.401	371.291	344.74	354.386	354.386	1181529984
11/20/2017	354.094	372.137	353.289	366.73	366.73	807027008
11/21/2017	367.443	372.47	350.693	360.401	360.401	949912000
11/22/2017	360.312	381.42	360.147	380.652	380.652	800819008
11/23/2017	381.439	425.548	376.088	410.166	410.166	1845680000
11/24/2017	412.501	480.973	402.758	474.911	474.911	2292829952
11/25/2017	475.676	485.192	461.053	466.276	466.276	1422080000
11/26/2017	465.974	472.723	451.606	471.33	471.33	1197779968
11/27/2017	471.531	493.405	468.485	480.355	480.355	1396480000
11/28/2017	480.518	482.48	466.347	472.902	472.902	1346499968
11/29/2017	473.281	522.307	425.071	427.523	427.523	2675940096
11/30/2017	431.215	465.497	401.243	447.114	447.114	1903040000
12/1/2017	445.209	472.609	428.312	466.54	466.54	1247879936
12/2/2017	466.851	476.239	456.653	463.449	463.449	943649984
12/3/2017	463.705	482.814	451.852	465.853	465.853	990556992
12/4/2017	466.054	474.777	453.312	470.204	470.204	1005550016
12/5/2017	470.294	473.558	457.66	463.281	463.281	1216720000
12/6/2017	462.604	462.708	420.21	428.588	428.588	1998259968
12/7/2017	426.369	441.397	414.411	434.408	434.408	2129570048
12/8/2017	434.989	466.062	422.367	456.031	456.031	2336379904
12/9/2017	457.344	504.147	456.253	473.502	473.502	2003849984
12/10/2017	472.789	472.789	429.514	441.721	441.721	1404179968

……..

- **For year 2018**

5/1/2018	670.463	674.403	637.54	673.613	673.613	2678960128
5/2/2018	674.075	688.842	667.42	687.149	687.149	2822269952
5/3/2018	686.591	784.341	686.591	779.543	779.543	4210939904
5/4/2018	776.775	803.746	762.632	785.624	785.624	3533410048
5/5/2018	784.583	827.455	784.237	816.12	816.12	3035040000
5/6/2018	816.088	835.057	764.883	792.311	792.311	3105570048
5/7/2018	793.339	795.758	710.178	753.725	753.725	4316120064
5/8/2018	755.009	774.249	728.129	752.857	752.857	2920489984
5/9/2018	752.902	759.529	718.472	752.275	752.275	2877870080
5/10/2018	752.579	766.748	726.664	727.277	727.277	2748950016
5/11/2018	727.013	736.977	669.825	679.586	679.586	3290080000
5/12/2018	679.877	691.411	644.066	686.048	686.048	2668480000
5/13/2018	687.175	741.312	675.319	733.496	733.496	2362500096
5/14/2018	732.733	742.17	695.792	730.549	730.549	3005110016
5/15/2018	731.143	739.052	700.995	708.871	708.871	2523069952
5/16/2018	708.087	710.2	682.541	707.05	707.05	2476130048
5/17/2018	708.718	718.833	668.834	672.657	672.657	2350619904
5/18/2018	672.102	695.031	663.809	694.367	694.367	2305740032
5/19/2018	695.072	715.578	686.791	696.53	696.53	2021549952
5/20/2018	697.923	723.753	692.669	715.369	715.369	2156910080
5/21/2018	717.193	719.278	692.494	699.222	699.222	2005170048
5/22/2018	700.178	700.976	644.026	647.741	647.741	2230469888
5/23/2018	646.67	651.636	572.952	583.585	583.585	2995429888
5/24/2018	584.536	610.818	557.206	601.755	601.755	2791099904
5/25/2018	602.14	617.186	575.624	586.734	586.734	2110919936
5/26/2018	587.426	606.175	583.512	587.28	587.28	1694300032
5/27/2018	588.52	590.328	562.866	572.668	572.668	1788790016
5/28/2018	573.045	576.049	512.552	516.036	516.036	2356900096
5/29/2018	516.148	572.264	516.148	565.388	565.388	2330820096
5/30/2018	566.83	583.136	545.431	559.59	559.59	2053970048
5/31/2018	558.497	585.538	557.066	577.645	577.645	1985040000
6/1/2018	578.672	589.093	567.665	580.043	580.043	1945890048
6/2/2018	580.429	597.077	577.322	591.808	591.808	1880390016
6/3/2018	591.259	624.513	591.259	618.329	618.329	1832550016
6/4/2018	619.437	623.429	583.747	592.985	592.985	1903430016
6/5/2018	593.406	611.33	580.982	609.303	609.303	1844269952
6/6/2018	610.262	611.643	596.396	607.124	607.124	1756530048
6/7/2018	607.687	616.144	601.693	605.187	605.187	1880140032
6/8/2018	605.443	608.811	595.593	601.077	601.077	1637779968
6/9/2018	600.905	608.583	597.562	597.562	597.562	1519309952
6/10/2018	594.345	594.345	511.889	526.479	526.479	2234880000
6/11/2018	524.857	536.858	515.269	533.284	533.284	1982119936
6/12/2018	532.71	538.955	491.235	496.843	496.843	1932760064
6/13/2018	498.018	501.905	459.004	477.494	477.494	2080130048
6/14/2018	478.381	523.545	467.466	519.742	519.742	2458650112
6/15/2018	520.48	521.305	487.469	491.004	491.004	1808269952

…….

- **For year 2019**

10/6/2019	176.3644	177.3645	171.2998	173.0591	173.0591	5852890674
10/7/2019	172.9403	182.3564	171.5278	181.1863	181.1863	7844316834
10/8/2019	181.1134	184.36	179.1334	182.0216	182.0216	7466282780
10/9/2019	182.0363	194.3005	180.6684	193.2933	193.2933	9088122101
10/10/2019	193.1911	193.8969	188.3065	191.6597	191.6597	8375913276
10/11/2019	191.8011	195.3188	181.6618	182.5697	182.5697	9128522970
10/12/2019	182.534	186.3049	179.9848	180.8266	180.8266	7494328840
10/13/2019	180.8613	185.0759	180.3172	182.0752	182.0752	6733182273
10/14/2019	182.0285	187.3036	181.6623	186.9609	186.9609	7276520699
10/15/2019	186.9771	187.7599	179.4628	181.4061	181.4061	7731456579
10/16/2019	181.3437	181.6686	174.0642	176.0135	176.0135	7691244590
10/17/2019	175.9069	178.8996	174.5658	178.0284	178.0284	6737237423
10/18/2019	177.9886	178.1798	171.0051	173.6213	173.6213	7566257807
10/19/2019	173.6494	175.6082	172.2155	172.913	172.913	6551453871
10/20/2019	172.9748	176.7187	171.2107	175.5344	175.5344	6801091120
10/21/2019	175.5248	177.7417	173.2878	174.921	174.921	6815820627
10/22/2019	174.9051	175.65	172.2662	172.3009	172.3009	6990951966
10/23/2019	172.2628	172.4206	157.4634	162.4028	162.4028	9624925919
10/24/2019	162.5141	164.1476	160.888	162.1685	162.1685	7300917537
10/25/2019	162.1897	183.0001	161.966	181.5232	181.5232	10358594018
10/26/2019	181.6671	195.9425	176.1338	179.8355	179.8355	13831784986
10/27/2019	179.9326	188.1553	177.6824	184.2422	184.2422	10815941952
10/28/2019	184.3975	187.8832	180.2496	182.6628	182.6628	10406734124
10/29/2019	182.6704	191.8468	182.3644	190.3426	190.3426	10622761958
10/30/2019	190.3364	191.1119	181.5464	184.6922	184.6922	10484902804
10/31/2019	184.7976	188.7513	180.0684	183.9669	183.9669	9607939606
11/1/2019	183.8037	185.0597	181.0945	183.9699	183.9699	9145611130
11/2/2019	184.0182	185.709	182.7977	183.9257	183.9257	8087991830
11/3/2019	183.9948	185.0242	179.8181	182.425	182.425	8760247744
11/4/2019	182.319	188.0229	181.8216	186.3552	186.3552	10551917945
11/5/2019	186.3093	191.2578	184.332	189.3042	189.3042	10024177342
11/6/2019	189.1135	193.5473	188.6677	191.5938	191.5938	10156458684
11/7/2019	191.5043	191.8611	186.567	187.9765	187.9765	9081247799
11/8/2019	187.9243	188.816	182.5414	184.2115	184.2115	9176780911
11/9/2019	184.3117	185.8385	183.9022	185.0287	185.0287	7277418704
11/10/2019	184.9434	190.9206	184.6868	189.4773	189.4773	8752784211
11/11/2019	189.5029	190.3384	185.0863	185.4896	185.4896	7877424106
11/12/2019	185.5648	187.948	184.1911	186.8434	186.8434	7792186666
11/13/2019	186.9424	189.3319	185.9379	188.2587	188.2587	7343173596
11/14/2019	188.2725	188.6346	184.9923	185.9996	185.9996	7872664470
11/15/2019	186.0582	186.2518	179.3796	180.5212	180.5212	8815678477

…..

- **For year 2020**

5/3/2020	215.3521	219.2709	208.6924	210.9332	210.9332	20469034664
5/4/2020	210.8909	211.8284	199.0477	208.174	208.174	22602446422
5/5/2020	208.013	211.7786	204.0311	206.7744	206.7744	19004689099
5/6/2020	206.4814	211.5346	204.0409	204.0558	204.0558	20343543799
5/7/2020	203.9129	214.3925	202.0748	212.2894	212.2894	23594744655
5/8/2020	212.1982	216.3277	208.8307	212.9916	212.9916	20445139356
5/9/2020	213.1422	214.7391	209.0715	211.6001	211.6001	18950547549
5/10/2020	211.5522	211.5522	182.7112	188.5996	188.5996	25211575193
5/11/2020	188.6322	191.3623	180.7183	185.9128	185.9128	20054601647
5/12/2020	185.8773	191.6013	185.7018	189.3125	189.3125	15899726284
5/13/2020	189.3741	200.1973	189.1277	199.1933	199.1933	17054662289
5/14/2020	198.8915	204.1176	196.8688	202.9491	202.9491	20150524861
5/15/2020	202.9554	203.5664	193.7557	195.6227	195.6227	16602342092
5/16/2020	195.6134	202.7712	194.5016	200.6771	200.6771	15379081645
5/17/2020	200.6089	209.1609	200.1028	207.1587	207.1587	15470397303
5/18/2020	207.1798	215.9085	207.1091	214.5251	214.5251	17411566928
5/19/2020	214.6049	214.6049	210.1431	213.4511	213.4511	14346192779
5/20/2020	213.4462	214.7168	207.9758	210.0967	210.0967	12730175511
5/21/2020	210.1292	211.6252	193.3464	199.8836	199.8836	13308321229
5/22/2020	199.8371	208.5915	198.0409	207.1692	207.1692	12041592114
5/23/2020	207.1945	210.3865	205.2942	208.6944	208.6944	10665476768
5/24/2020	208.7161	210.5951	202.3703	202.3703	202.3703	11833299572
5/25/2020	201.9827	206.3615	200.6676	205.3197	205.3197	10415044124
5/26/2020	205.2596	205.7525	200.2643	201.9023	201.9023	10159741290
5/27/2020	201.893	208.8634	201.7851	208.8634	208.8634	10631034756
5/28/2020	208.8854	220.2765	206.2427	219.8404	219.8404	12212469604
5/29/2020	219.925	224.2169	218.2381	220.6751	220.6751	12265816557
5/30/2020	220.7172	243.9431	218.7445	242.3456	242.3456	15027397867
5/31/2020	242.3514	244.0453	230.0528	230.9757	230.9757	12234904813
6/1/2020	230.8603	248.2363	230.4881	246.9918	246.9918	13951727936
6/2/2020	246.8282	252.222	233.2253	237.2191	237.2191	13782107567
6/3/2020	237.3952	244.1793	235.4644	244.1793	244.1793	9861760817
6/4/2020	244.1053	245.929	236.7653	244.4264	244.4264	10170414304
6/5/2020	244.3496	247.3295	240.6821	241.222	241.222	9293963914
6/6/2020	241.2014	245.981	239.7245	241.9313	241.9313	8114873845
6/7/2020	241.9081	245.4353	236.3253	245.1673	245.1673	9544883157
6/8/2020	245.1786	246.6442	241.5422	246.3099	246.3099	8076783299
6/9/2020	246.175	248.3424	242.3385	244.9115	244.9115	8446545788
6/10/2020	244.8221	248.6512	242.8197	247.4449	247.4449	8792990206
6/11/2020	247.5485	249.8883	229.943	231.7027	231.7027	12356528860
6/12/2020	231.6255	239.3547	229.6451	237.4932	237.4932	8868955009
6/13/2020	237.5446	239.1931	235.8897	238.9088	238.9088	7141624980
6/14/2020	238.9682	239.1015	232.9582	234.1147	234.1147	7439385176
6/15/2020	234.0583	234.2378	221.2418	229.9289	229.9289	10536099884
6/16/2020	229.7623	236.3943	228.4261	234.4162	234.4162	7965648016

…….

- **For year 2021**

1/1/2021	737.7084	749.2018	719.7922	730.3676	730.3676	13652004358
1/2/2021	730.4026	786.7985	718.1095	774.535	774.535	19740771179
1/3/2021	774.5118	1006.565	771.5616	975.5077	975.5077	45200463368
1/4/2021	977.0588	1153.189	912.3054	1040.233	1040.233	56945985763
1/5/2021	1041.499	1129.371	986.8113	1100.006	1100.006	41535932781
1/6/2021	1101.005	1209.429	1064.233	1207.112	1207.112	44699914188
1/7/2021	1208.078	1282.58	1167.443	1225.678	1225.678	40468027280
1/8/2021	1225.968	1273.828	1076.082	1224.197	1224.197	44334826666
1/9/2021	1223.74	1303.872	1182.27	1281.077	1281.077	33233105361
1/10/2021	1280.871	1347.926	1194.716	1262.247	1262.247	40616938053
1/11/2021	1261.623	1261.623	924.9226	1090.145	1090.145	60733630300
1/12/2021	1088.527	1149.24	1012.764	1043.435	1043.435	37494601692
1/13/2021	1043.741	1134.339	994.5491	1130.739	1130.739	30109792795
1/14/2021	1130.231	1244.163	1093.061	1218.453	1218.453	33410915929
1/15/2021	1221.877	1250.506	1090.721	1171.835	1171.835	35972039310
1/16/2021	1171.443	1290.054	1157.624	1233.538	1233.538	32319240157
1/17/2021	1233.453	1265.645	1174.389	1230.172	1230.172	29258032819
1/18/2021	1230.313	1259.45	1187.311	1257.28	1257.28	25817455560
1/19/2021	1257.435	1432.3	1254.523	1377.296	1377.296	47195935190
1/20/2021	1375.248	1405.744	1243.3	1382.274	1382.274	46784030909
1/21/2021	1382.684	1382.684	1098.476	1121.571	1121.571	45932464754
1/22/2021	1118.889	1271.688	1046.597	1236.512	1236.512	43918338506
1/23/2021	1235.268	1272.151	1200.893	1230.991	1230.991	27253895441
1/24/2021	1231.211	1395.111	1225.274	1391.609	1391.609	36418163554
1/25/2021	1390.64	1467.785	1304.974	1324.415	1324.415	43565777745
1/26/2021	1323.742	1376.085	1253.34	1357.058	1357.058	41572917750
1/27/2021	1358.333	1368.074	1215.311	1253.187	1253.187	39394416990
1/28/2021	1251.28	1356.289	1226.174	1332.492	1332.492	34637234789
1/29/2021	1369.087	1428.981	1292.24	1382.523	1382.523	53611955259
1/30/2021	1382.232	1402.4	1328.529	1376.115	1376.115	30616574234
1/31/2021	1376.824	1378.916	1288.502	1314.986	1314.986	25198853581
2/1/2021	1314.855	1373.846	1274.358	1369.041	1369.041	29210670920
2/2/2021	1369.505	1542.991	1362.771	1515.194	1515.194	45437142801
2/3/2021	1514.77	1660.91	1510.01	1660.91	1660.91	41874566399
2/4/2021	1661.17	1689.187	1561.854	1594.763	1594.763	44396871836
2/5/2021	1594.793	1756.511	1594.793	1718.651	1718.651	40108628454
2/6/2021	1717.797	1738.314	1649.069	1677.847	1677.847	39873420648
2/7/2021	1677.606	1690.037	1501.75	1614.228	1614.228	39889440151
2/8/2021	1613.642	1770.591	1571.58	1746.617	1746.617	48012285956
2/9/2021	1746.926	1815.964	1711.621	1768.035	1768.035	44180727529
2/10/2021	1768.04	1826.697	1686.542	1744.243	1744.243	41916084617

…..

- **For year 2022**

2/1/2022	2687.899	2802.315	2682.622	2792.117	2792.117	13194846235
2/2/2022	2791.959	2802.212	2630.12	2682.854	2682.854	13876301217
2/3/2022	2682.226	2712.483	2587.783	2679.163	2679.163	12755505065
2/4/2022	2681.058	2983.587	2675.444	2983.587	2983.587	18987223729
2/5/2022	2984.446	3054.13	2966.781	3014.648	3014.648	13102093957
2/6/2022	3014.96	3061.261	2965.43	3057.476	3057.476	9466018022
2/7/2022	3057.422	3182.528	3002.927	3142.471	3142.471	15197063785
2/8/2022	3143.009	3219.473	3038.378	3122.609	3122.609	17136080906
2/9/2022	3121.183	3263.156	3063.16	3239.457	3239.457	13951308490
2/10/2022	3240.113	3271.316	3070.378	3077.482	3077.482	18629485080
2/11/2022	3077.413	3127.451	2888.708	2927.384	2927.384	16043881065
2/12/2022	2927.386	2980.079	2870.177	2917.363	2917.363	11254355757
2/13/2022	2916.79	2947.777	2845.398	2883.463	2883.463	9054963563
2/14/2022	2880.188	2957.964	2840.258	2933.479	2933.479	12164552172
2/15/2022	2933.729	3185.521	2917.857	3179.877	3179.877	13921257873
2/16/2022	3180.447	3181.617	3055.168	3127.83	3127.83	12352406833
2/17/2022	3126.858	3154.615	2861.852	2881.482	2881.482	15860206214
2/18/2022	2884.341	2937.309	2761.643	2785.728	2785.728	15748173433
2/19/2022	2784.873	2826.348	2707.378	2763.701	2763.701	9774183169
2/20/2022	2763.757	2763.757	2585.946	2628.648	2628.648	11641437834
2/21/2022	2627.666	2752.458	2568.254	2573.816	2573.816	18646392740
2/22/2022	2572.899	2648.917	2510.679	2639.299	2639.299	16360200507
2/23/2022	2639.447	2741.368	2587.413	2590.36	2590.36	13382637240
2/24/2022	2588.166	2689.048	2308.915	2598.067	2598.067	29312342666
2/25/2022	2598.436	2821.972	2579.208	2764.536	2764.536	17208902048
2/26/2022	2764.99	2849.424	2745.009	2781.112	2781.112	11724648351
2/27/2022	2780.504	2831.125	2581.616	2621.802	2621.802	16150857254
2/28/2022	2621.172	2929.18	2586.388	2919.201	2919.201	19266124733
3/1/2022	2919.776	3029.652	2868.939	2972.485	2972.485	18757425786
3/2/2022	2972.472	3026.868	2919.943	2950.118	2950.118	16636517503
3/3/2022	2950.157	2964.673	2797.319	2834.469	2834.469	13091199728
3/4/2022	2834.987	2835.176	2587.748	2617.156	2617.156	14496939024
3/5/2022	2618.474	2679.103	2596.99	2664.831	2664.831	8072368396
3/6/2022	2664.944	2673.637	2555.037	2555.037	2555.037	8872976607
3/7/2022	2555.298	2639.943	2455.594	2497.771	2497.771	14594098731
3/8/2022	2497.721	2618.166	2489.755	2576.748	2576.748	13922922903
3/9/2022	2577.165	2761.796	2573.655	2729.783	2729.783	14173665398
3/10/2022	2729.116	2729.116	2566.193	2608.049	2608.049	13292477213
3/11/2022	2608.271	2664.559	2534.688	2559.563	2559.563	12382419582
3/12/2022	2559.661	2606.439	2559.127	2574.754	2574.754	6532996574
3/13/2022	2573.488	2594.55	2503.885	2518.945	2518.945	8632000379
3/14/2022	2518.486	2604.034	2505.299	2590.696	2590.696	11244398839
3/15/2022	2590.669	2662.33	2515.766	2620.15	2620.15	12861105614

……

Appendix-2:- Code Snippets

This subsection provides code snippets representing the implementation of three machine learning models and to produce the respective results and graphs.

- **For importing dataset**

```
import pandas as pd
import numpy as np
%matplotlib inline
import matplotlib.pyplot as plt
```

```
start=datetime.datetime(2015,1,1)
end=datetime.datetime(2021,1,1)

df=web.DataReader("ETH-USD",'yahoo',start,end)
```

- **For train and test data split**

```
train = df.iloc[:1120]
test = df.iloc[1120:]
```

```
train.columns
```

```
Index(['High', 'Low', 'Open', 'Close', 'Volume', 'Adj Close'], dtype='object')
```

```
y_train=train.drop(['High', 'Low', 'Open', 'Volume', 'Adj Close'],axis=1).values
X_train=np.arange(0,1120).reshape(-1,1)
```

```
len(X_train)==len(y_train)
```

```
True
```

1) For Linear regression

```
from sklearn.linear_model import LinearRegression
model = LinearRegression()

### Fit/Train the Model on the training data



model.fit(X_train,y_train)
```

- **For prediction on test data**

```
▾ LinearRegression
LinearRegression()
```

```
y_test=test.drop(['High', 'Low', 'Open', 'Volume', 'Adj Close'],axis=1).values
```

```
X_test=np.arange(0,30).reshape(-1,1)
```

```
preds=model.predict(X_test)
```

```
test['preds']=preds
```

```
<ipython-input-11-9d12be3e01f7>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/
  test['preds']=preds
```

```
from sklearn.metrics import mean_absolute_error
```

```
test
```

- **To plot a graph for test data prediction results**

```
linear_regression_test_error=mean_absolute_error(test['Close'],test['preds'])**0.5
```

```
test=test.drop(['High','Low','Open','Volume','Adj Close'],axis=1)
```

```
test.plot()
```

- **For prediction on training data**

```
preds=model.predict(X_train)
train['preds']=preds
```

```
linear_regression_train_error=mean_absolute_error(train['Close'],train['preds'])**0.5
```

```
train['preds']=preds

train=train.drop(['High','Low','Open','Volume','Adj Close'],axis=1)

train.plot()
```

- **To plot price for Closing, 1month- SMA and 2 week- SMA**

```
train = df.iloc[:754]
test = df.iloc[754:]
```

```
df=df.drop(['High','Low','Open','Volume','Adj Close'],axis=1)
```

```
df['1month-SMA'] = df['Close'].rolling(window=30).mean()
df['2week-SMA'] = df['Close'].rolling(window=14).mean()

df.head(15)

df.plot()
```

- **To plot price for Closing, and EWMA12**

```
df['EWMA12'] = df['Close'].ewm(span=14,adjust=False).mean()

df[['Close','EWMA12']].plot()
```

2) For Exponential Smoothing

```
# statsmodels.tsa.holtwinters SimpleExpSmoothing
from statsmodels.tsa.holtwinters import SimpleExpSmoothing

span = 14
alpha = 2/(span+1)

df['EWMA14'] = df['Close'].ewm(alpha=alpha,adjust=False).mean()
df['SES14']=SimpleExpSmoothing(df['Close']).fit(smoothing_level=alpha,optimized=False).fittedvalues.shift(-1)
df.head()
```

- **To make test predictions**

```
from statsmodels.tsa.holtwinters import ExponentialSmoothing

fitted_model = ExponentialSmoothing(train['Close'],trend='mul',seasonal='mul',seasonal_periods=14).fit()

## Evaluating Model against Test Set

# YOU CAN SAFELY IGNORE WARNINGS HERE!
# THIS WILL NOT AFFECT YOUR FORECAST, IT'S JUST SOMETHING STATSMODELS NEEDS TO UPDATE UPON NEXT RELEASE.
test_predictions = fitted_model.forecast(30).rename('HW Forecast')

test_predictions
```

```
test['preds_exp']=test_predictions
```

```
<ipython-input-34-5b427ec20ac7>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stabl
  test['preds_exp']=test_predictions
```

- **To plot test and train data split for closing price**

```
exponential_test_error=mean_absolute_error(test['Close'],test['preds_exp'])**0.5
```

```
train['Close'].plot(legend=True,label='TRAIN')
test['Close'].plot(legend=True,label='TEST',figsize=(12,8));
```

- **To plot test, train and prediction data of closing price**

```
train['Close'].plot(legend=True,label='TRAIN')
test['Close'].plot(legend=True,label='TEST',figsize=(20,10))


test_predictions.plot(legend=True,label='PREDICTION');
```

- **To plot test, train and prediction data of closing price in elaborated view**

```
train['Close'].plot(legend=True,label='TRAIN')
test['Close'].plot(legend=True,label='TEST',figsize=(12,8))
test_predictions.plot(legend=True,label='PREDICTION',xlim=['2020-11-15','2021-01-01']);
```

- **To make predictions on EWMA14 and SES14**

```
train = df.iloc[:1120]
test = df.iloc[1120:]

train.columns

y_train=train.drop(['High', 'Low', 'Open', 'Volume', 'Adj Close'],axis=1).values
X_train=np.arange(0,1120).reshape(-1,1)

len(X_train)==len(y_train)
```

```
model.fit(X_train,y_train)

y_test=test.drop(['High', 'Low', 'Open', 'Volume', 'Adj Close'],axis=1).values

X_test=np.arange(0,30).reshape(-1,1)

preds=model.predict(X_test)

#test['preds']=preds

#test=test.drop(['High','Low','Open','Volume','Adj Close'],axis=1)
```

```
train
```

3) For Decision Trees

```
model = DecisionTreeRegressor()

### Fit/Train the Model on the training data


model.fit(X_train,y_train)
```

```
▾ DecisionTreeRegressor
DecisionTreeRegressor()
```

```
y_train
```

- **To make prediction for test data on closing price of Ethereum**

```
array([[320.88400269, 320.88400269, 320.88400269],
       [299.25299072, 317.99986776, 317.99986776],
       [314.68099976, 317.55735202, 317.55735202],
       ...,
       [614.8425293 , 541.93964974, 541.93964974],
       [587.32415771, 547.99091747, 547.99091747],
       [598.35235596, 554.70577593, 554.70577593]])
```

```
preds=model.predict(X_test)
```

```
(test['Close'].values).shape
```

```
(30,)
```

```
preds[1]
```

```
array([299.25299072, 317.99986776, 317.99986776])
```

```
decision_test_error=mean_absolute_error(test['Close'].values,preds)**0.5
```

```
preds
```

- **To plot predictions and closing price on test data**

```
#y_test=test.drop(['High', 'Low', 'Open', 'Volume', 'Adj Close'],axis=1).values

X_test=np.arange(0,30).reshape(-1,1)

preds=model.predict(X_test)

test['preds']=preds

#test=test.drop(['High','Low','Open','Volume','Adj Close'],axis=1)

test.plot()
```

- **To plot predictions and closing price on train data**

```
X_train=np.arange(0,1120).reshape(-1,1)

preds=model.predict(X_train)

train['preds']=preds

train=train.drop(['High','Low','Open','Volume','Adj Close'],axis=1)

train.plot()
```

- **Code for comparative analysis based on RMSE**

```
data = {'DecisonTree_RMSE':[decision_test_error],
        'ExponentialSmoothing_RMSE':[exponential_test_error],
       'LinearRegression_RMSE':[linear_regression_test_error]}
pf=pd.DataFrame(data)
```