University of Mumbai

Bitcoin Price Prediction Using Machine Learning

Submitted at the end of semester VII in partial fulfillment of requirements

For the degree of

Bachelor of Technology

by

BABITA RATUDI
Roll No: 1813090

MUDASSIR KHATRI
Roll No: 1814032

ONKAR SANAP
Roll No: 1814035

YASHVI VORA
Roll No: 1814121

Guide & Co-Guide

PROF. NILKAMAL MORE & PROF. RUCHIRA JADHAV

Department of Electronics and Telecommunication Engineering
K. J. Somaiya College of Engineering, Mumbai-77
(Autonomous College Affiliated to University of Mumbai)

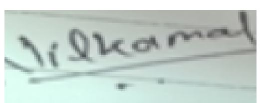
Batch 2018 -2022

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Certificate

This is to certify that the dissertation report entitled **Bitcoin Price Prediction Using Machine Learning** submitted by Babita Ratudi at the end of semester VII of LY B. Tech is a bona fide record for partial fulfillment of requirements for the degree of Bachelors in Technology in Electronics and Telecommunication Engineering of University of Mumbai.



Guide   Co-Guide

_____________________   __________________________

Principal   Head of the Department

Date: 23rd Dec, 2021

Place: Mumbai-77

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Certificate of Approval of Examiners

We certify that this dissertation report entitled **Bitcoin Price Prediction Using Machine Learning** is a bonafide record of project work done by Babita Ratudi,Yashvi Vora,Onkar Sanap and Mudassir Khatri during semester VII.

This project work is submitted at the end of semester VII in partial fulfillment of requirements for the degree of Bachelors in Technology in Electronics and Telecommunication Engineering of University of Mumbai.

____

Internal Examiners

____

External/Internal Examiners

Date: 23rd Dec, 2021

Place: Mumbai-77

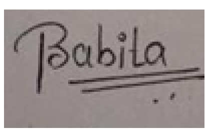
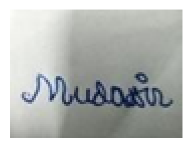
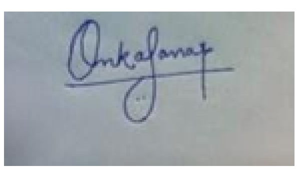
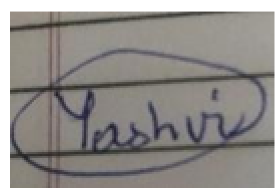
K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

DECLARATION

We declare that this written report submission represents the work done based on our and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in my submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

 **Signature of the Student** **Roll No.-1813090**	 **Signature of the Student** **Roll No.-1814032**
 **Signature of the Student** **Roll No.-1814035**	 **Signature of the Student** **Roll No.-1814121**

Date: 23rd Dec, 2021

Place: Mumbai-77

Abstract

The purpose of this study is to find out with what accuracy the direction of the price of Bitcoin can be predicted using machine learning methods. This is basically a time series prediction problem. While much research exists surrounding the use of different machine learning.

Techniques for time series prediction, research in this area relating specifically to Bitcoin is lacking. In addition, Bitcoin as a currency is in a transient stage and as a result is considerably more volatile than other currencies such as the USD. Interestingly, it is the top performing currency four out of the last five years. Thus, its prediction offers great potential and this provides motivation for research in the area.

Finally, in analyzing the chosen dependent variables, each variable's importance is assessed using a random forest algorithm. In addition, the ability to predict the direction of the price of an asset such as Bitcoin offers the opportunity for profit to be made by trading the asset.

Contents

Chapter 1

Introduction

1.1 Background

Bitcoin is a digital cryptocurrency and payment system that is entirely decentralized, meaning it is based on peer-to-peer transactions with no bureaucratic oversight. Transactions and liquidity within the network are instead based on cryptography. The system first emerged formally in 2009 and is currently a thriving open-source community and payment network. Based on the uniqueness of Bitcoin's payment protocol and its growing adoption, the Bitcoin ecosystem is gaining lots of attention from businesses, consumers, and investors alike. Namely, for the ecosystem to thrive, we need to replicate financial services and products that currently exist in our traditional, fiat currency world and make them available and custom-tailored to Bitcoin, as well as other emerging cryptocurrencies. .

1.2 Motivation

Bitcoin: Bitcoin is a crypto currency which is used worldwide for digital payment or simply for investment purposes. Bitcoin is decentralized i.e. it is not owned by anyone. Transactions made by Bitcoins are easy as they are not tied to any country. Investment can be done through various marketplaces known as "bitcoin exchanges". These allow people to sell/buy Bitcoins using different currencies. The largest Bitcoin exchange is Mt Gox.

Price Prediction: The Bitcoin market's financial analog is, of course, a stock market. To maximize financial reward, the field of stock market prediction has grown over the past decades, and has more recently exploded with the advent of high-frequency, low-latency trading hardware coupled with robust machine learning algorithms. Thus, it makes sense that this prediction methodology is replicated in the world of Bitcoin, as the network gains greater liquidity and more people develop an interest in investing profitably in the system. To do so, we feel it is necessary to leverage machine learning technology to predict the price of Bitcoin

1.3 <u>Project Scope</u>

- Today Bitcoin is a secure transaction system that has a valuable impact on capital. They are awarded under a restriction in which customers offer their computer authority to register and list trades with the bitcoins. The purchase and sale of Bitcoins in different currencies is carried out in an alternative workplace where "purchase" or "sell" requests are placed in the ordered e-book. "Buy" or "bid" offers to talk about the purpose of purchasing certain Bitcoins measures at a few costs while "provide" or "ask" offers to talk about the expectation of providing certain Bitcoins measures at a certain cost. The change is ordered through the coordination of pricing requests from the arrangement of e-books to a valid exchange between customers and suppliers.
- The new corporation agreement between Starbucks, Microsoft, and ICE (International Exchange) is one of the good indications that retailers are prepared to accept cryptocurrency as a payment method. This allows Starbucks customers to pay for coffee by Bitcoins. We can see that the cryptocurrency market will thrive in the near future.
- Bitcoin is the new and most popular virtual currency,while the security and its volatility rate are debatable. This study makes it functional for the peer-to-peer transaction of bitcoin through the network and the blockchain technology.conduct the empirical study that compares the Bayesian neural network with other linear and non-linear benchmark models on modeling and predicting the Bitcoin process
- Bitcoins are stored in a digital wallet which is basically like a virtual bank account. The record of all the transactions, the timestamp data is stored in a place called Blockchain. Each record in a blockchain is called a block. Each block contains a pointer to a previous block of data. The data on blockchain is encrypted. During transactions the user’s name is not revealed, but only their wallet ID is made public.The number of studies is steadily increasing as the popularity of Bitcoin is surging.

1.4 Brief description of project undertaken

Our main goal is to build a web page that would predict bitcoin prices. This will be achieved using machine learning algorithms.The first step towards Bitcoin prediction is database collection. For our paper we have collected databases from the following sources: A Kaggle Kaggle, a subsidiary of Google LLC, is an online community of data scientists and machine learning practitioners. Kaggle allows users to find and publish data sets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers, and enter competitions to solve data science challenges.The next step is database normalization. We basically perform this step to achieve consistency i.e. reduce or eliminate duplicate data, insignificant points and other redundancies.Then we used Arima based model And Random Forest algorithm for prediction.

Tools used in the project :

- AdobeXD - UI Designing
- HTML, CSS, JS,JQuery,Bootstrap - Frontend
- Flask - Backend
- TensorFlow, Sci-kit Learn - ML Libraries
- Google Colab,Jupyter Notebook, VS Code - IDE
- GitHub - Version Control

Chapter 2

Literature Survey

The first paper surveys research work applying artificial intelligence and machine learning techniques in the field of cryptocurrencies. Analyzing cryptocurrencies is considered a relatively recent domain that became active in the last decade.

In the second paper, we attempt to predict the Bitcoin price accurately taking into consideration various parameters that affect the Bitcoin value. For the first phase of our survey, we aim to understand and identify daily trends in the Bitcoin market while gaining insight into optimal features surrounding Bitcoin price. For the second phase of our survey, using the available information, we will predict the sign of the daily price change with highest possible accuracy.

In the third paper, we discuss the method of Bayesian regression and its efficacy for predicting price variation of Bitcoin, a recently popularized virtual, cryptographic currency. Bayesian regression refers to utilizing empirical data as a proxy to perform Bayesian inference. In this paper, instead they utilize a model for predicting real-valued quantity, the price of Bitcoin. Based on this price prediction method, they devise a simple strategy for trading Bitcoin. The strategy is able to nearly double the investment in less than 60 Days period when run against real data trace.

In the Fourth paper, we conduct an empirical study that compares the Bayesian neural network with other linear and non-linear benchmark models on modeling and predicting the Bitcoin process. It shows that BNN performs well in predicting Bitcoin price time series and explaining the high volatility of the recent Bitcoin price.

In the Fifth paper, we predict the Bitcoin price accurately taking into consideration various parameters that affect the Bitcoin value. For the first phase of our survey, we aim to understand and identify daily trends in the Bitcoin market while gaining insight into optimal features surrounding Bitcoin price. For the second phase of our survey, using the available information, we will predict the sign of the daily price change with highest possible accuracy.

Chapter 3

Product Design

3.1 Objectives :

1] The main aim is to predict Bitcoin Price using different Machine learning algorithms. It must calculate the estimated price of bitcoin based on the historical data.

2] To predict bitcoin price with maximum efficiency using LSTM and ARIMA.

3] To compare between ARIMA and LSTM to find which is the most efficient algorithm for predicting bitcoin price.

4] Make a Full-stack website widely used for real-time data display and predictions of the profits and prices of cryptocurrencies using machine learning.

5] To ensure less risk and more profit for investor.

3.2 System diagram :

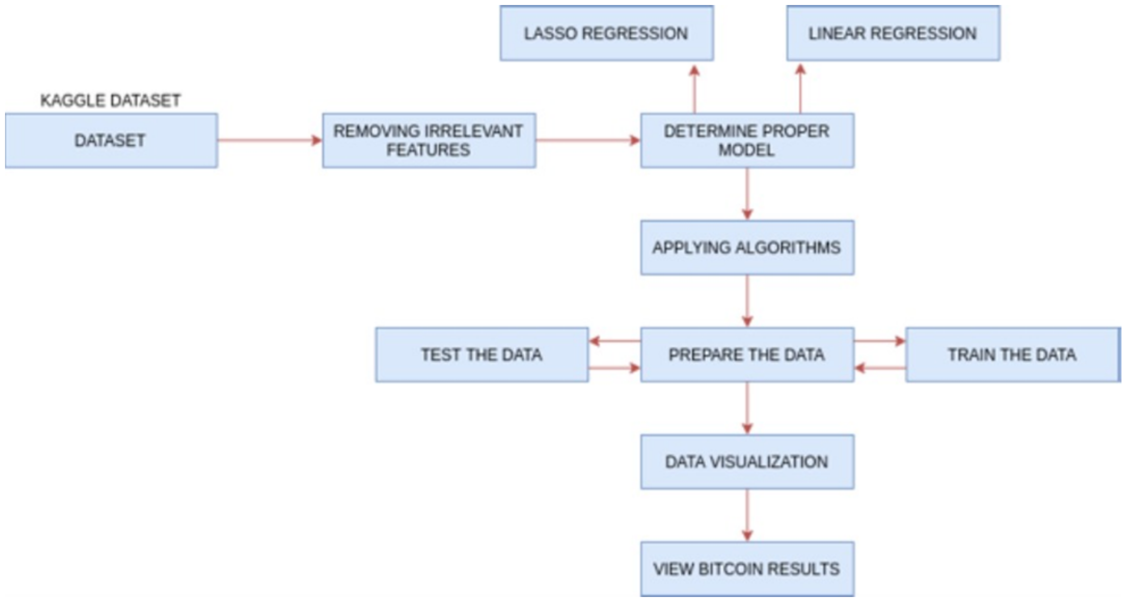


Fig 3.1 BLOCK DIAGRAM

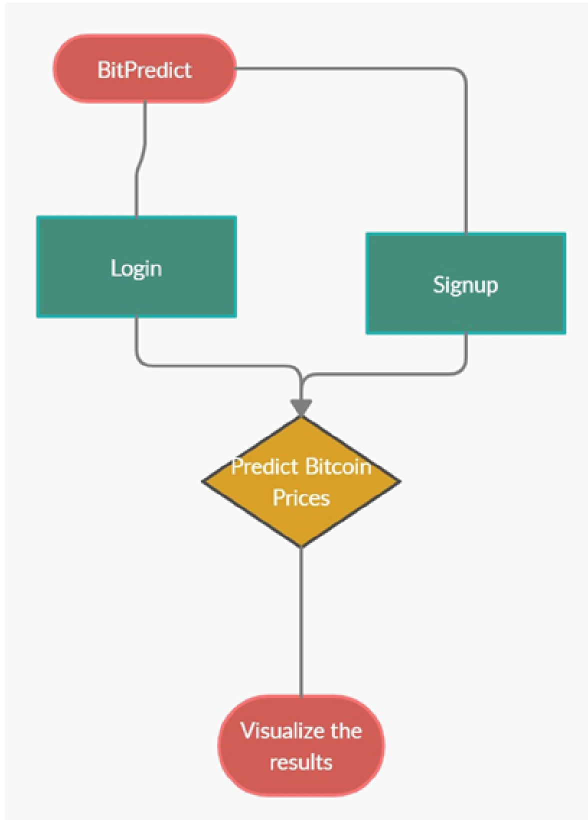


Fig 3.2 FLOWCHART DIAGRAM

3.3 Dataset :

1. We have used the Historical Dataset of Bitcoin Price from 2012-2021.It was downloaded through open source known as Kaggle.
2. The features of this data set are: Timestamp, Open, High, Low, Closed, Volume (BTC), Volume (Currency), and Weighted Price values

```
[ ]  data.info()

     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 4857377 entries, 0 to 4857376
     Data columns (total 8 columns):
      #   Column             Dtype
     ---  ------             -----
      0   Timestamp          int64
      1   Open               float64
      2   High               float64
      3   Low                float64
      4   Close              float64
      5   Volume_(BTC)       float64
      6   Volume_(Currency)  float64
      7   Weighted_Price     float64
     dtypes: float64(7), int64(1)
     memory usage: 296.5 MB
```

	Timestamp	Open	High	Low	Close	Volume_(BTC)	Volume_(Currency)	Weighted_Price
count	4.857377e+06	3.613769e+06	3.613769e+06	3.613769e+06	3.613769e+06	3.613769e+06	3.613769e+06	3.613769e+06
mean	1.471301e+09	6.009024e+03	6.013357e+03	6.004488e+03	6.009014e+03	9.323249e+00	4.176284e+04	6.008935e+03
std	8.428019e+07	8.996247e+03	9.003521e+03	8.988778e+03	8.996360e+03	3.054989e+01	1.518248e+05	8.995992e+03
min	1.325318e+09	3.800000e+00	3.800000e+00	1.500000e+00	1.500000e+00	0.000000e+00	0.000000e+00	3.800000e+00
25%	1.398179e+09	4.438600e+02	4.440000e+02	4.435200e+02	4.438600e+02	4.097759e-01	4.521422e+02	4.438306e+02
50%	1.471428e+09	3.596970e+03	3.598190e+03	3.595620e+03	3.597000e+03	1.979811e+00	3.810124e+03	3.596804e+03
75%	1.544288e+09	8.627270e+03	8.632980e+03	8.621090e+03	8.627160e+03	7.278216e+00	2.569821e+04	8.627637e+03
max	1.617149e+09	6.176356e+04	6.178183e+04	6.167355e+04	6.178180e+04	5.853852e+03	1.390067e+07	6.171621e+04

Fig 3.3 Data set

3.4 Random Forest

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

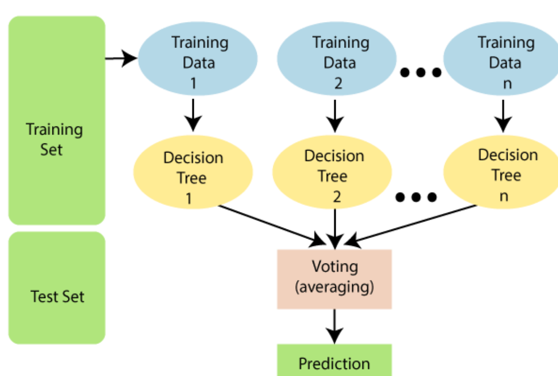


Fig 3.4 Working of Random Forest

Implementation of Random Forest-

Random Forest works in two-phase first is to create the random forest by combining N decision trees, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

Applications of Random Forest

There are mainly four sectors where Random forest mostly used:

- Banking: Banking sector mostly uses this algorithm for the identification of loan risk.
- Medicine: With the help of this algorithm, disease trends and risks of the disease can be identified.
- Land Use: We can identify the areas of similar land use by this algorithm.
- Marketing: Marketing trends can be identified using this algorithm.

Advantages of Random Forest

- Random Forest is capable of performing both Classification and Regression tasks.
- It is capable of handling large datasets with high dimensionality.
- It enhances the accuracy of the model and prevents the overfitting issue.

3.5 LSTM

Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies. They were introduced by Hochreiter & Schmidhuber (1997), and were refined and popularized by many people in following work.[1] They work tremendously well on a large variety of problems, and are now widely used.

LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn!

All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.

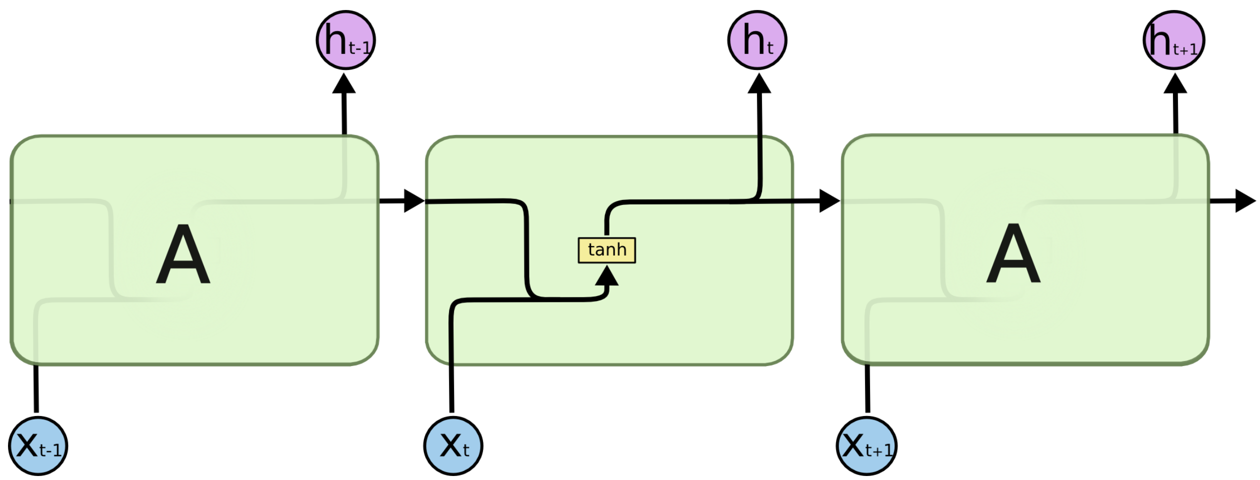


Fig 3.5 Basic RNN Architecture

LSTMs also have this chain-like structure, but the repeating module has a different structure. Instead of having a single neural network layer, there are four, interacting in a very special way.

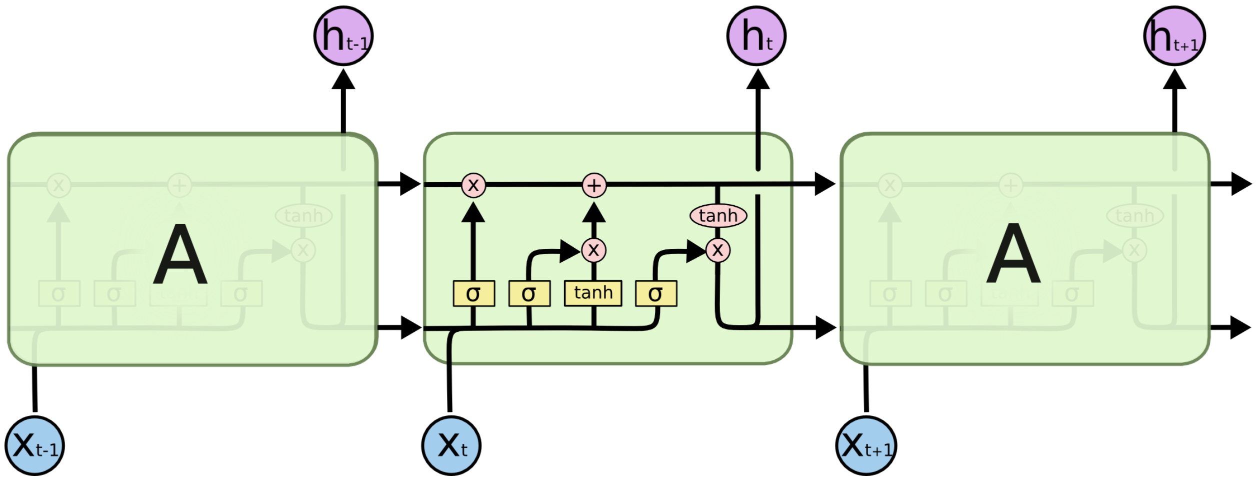


Fig 3.6 LSTM Architecture

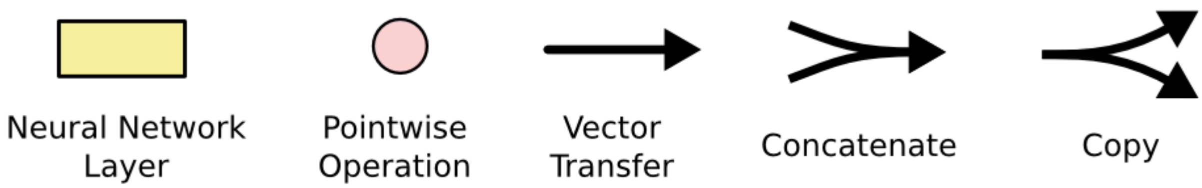


Fig 3.7 Representation of LSTM

In the above diagram, each line carries an entire vector, from the output of one node to the inputs of others. The pink circles represent pointwise operations, like vector addition, while the yellow boxes are learned neural network layers. Lines merging denote concatenation, while a line forking denotes its content being copied and the copies going to different locations.

Chapter 4

Implementation & Experimentation

Predicting a Bitcoin Price is a challenging problem in the Machine learning domain. In this article, we will use different machine learning algorithms such as Random Forest and LSTM. For getting more insights from Dataset we had plotted different graphs such as Bar, Histogram and Pie charts. Also for detecting whether the dataset is stationary or not we had used the Dicker Filler test.

For training our model I'm using the "bitstampUSD_1-min_data_2012-01-01_to_2021-03-31.csv". In this we have a total of 8 features such as Timestamp, Open, High, Low, Closed, Volume (BTC), Volume (Currency), and Weighted Price values. The Open, High, Low, and Closed columns in the dataset show the opening, the highest, the lowest, and the closing prices of Bitcoin against US $ in that minute.

It consists of Bitcoin Price from 2012 to 2021.

Step 1: Import the required libraries

```
import pandas as pd
import numpy as np
import seaborn as sns
from datetime import datetime
import matplotlib.pyplot as plt
import matplotlib as mpl
import statsmodels.api as st
import warnings
from itertools import product
from sklearn import preprocessing
from sklearn import model_selection
from sklearn.ensemble import RandomForestRegressor
```

Fig 4.1

Step 2: Load the descriptions

The format of our file is CSV. It contains 4857377 Rows and 8 columns. It is Bitcoin Price minute by minute.

```
[ ]  data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4857377 entries, 0 to 4857376
Data columns (total 8 columns):
 #   Column             Dtype
---  ------             -----
 0   Timestamp          int64
 1   Open               float64
 2   High               float64
 3   Low                float64
 4   Close              float64
 5   Volume_(BTC)       float64
 6   Volume_(Currency)  float64
 7   Weighted_Price     float64
dtypes: float64(7), int64(1)
memory usage: 296.5 MB
```

Fig 4.2

Step 3: Data Study using different Technique

Resampling:- Resampling is the method that consists of drawing repeated samples from the original data samples. The method of Resampling is a nonparametric method of statistical inference.

```
data=data.resample('D').mean()
data_week=data.resample('W').mean()
data_month=data.resample('M').mean()
data_year=data.resample('A-DEC').mean()
fig = plt.figure(figsize=[15, 6])
plt.suptitle('Bitcoin exchange', fontsize=22)
plt.subplot(221)
plt.plot(data.Weighted_Price, '-',label='Daily')
plt.legend(["Diffrence by Days"],loc="upper left")
plt.xlabel('Time',fontsize=12)
plt.ylabel('Price',fontsize=12)
plt.subplot(222)
plt.plot(data_week.Weighted_Price, '-',label='Weakly')
plt.legend(["Diffrence by Weakly"],loc="upper left")
plt.xlabel('Time',fontsize=12)
plt.ylabel('Price',fontsize=12)
plt.subplot(223)
plt.plot(data_month.Weighted_Price, '-',label='Monthly')
plt.legend(["Diffrence by Monthly"],loc="upper left")
plt.xlabel('Time',fontsize=12)
plt.ylabel('Price',fontsize=12)
plt.subplot(224)
plt.plot(data_year.Weighted_Price, '-',label='Yearly')
plt.legend(["Diffrence by Yearly"],loc="upper left")
plt.xlabel('Time',fontsize=12)
```

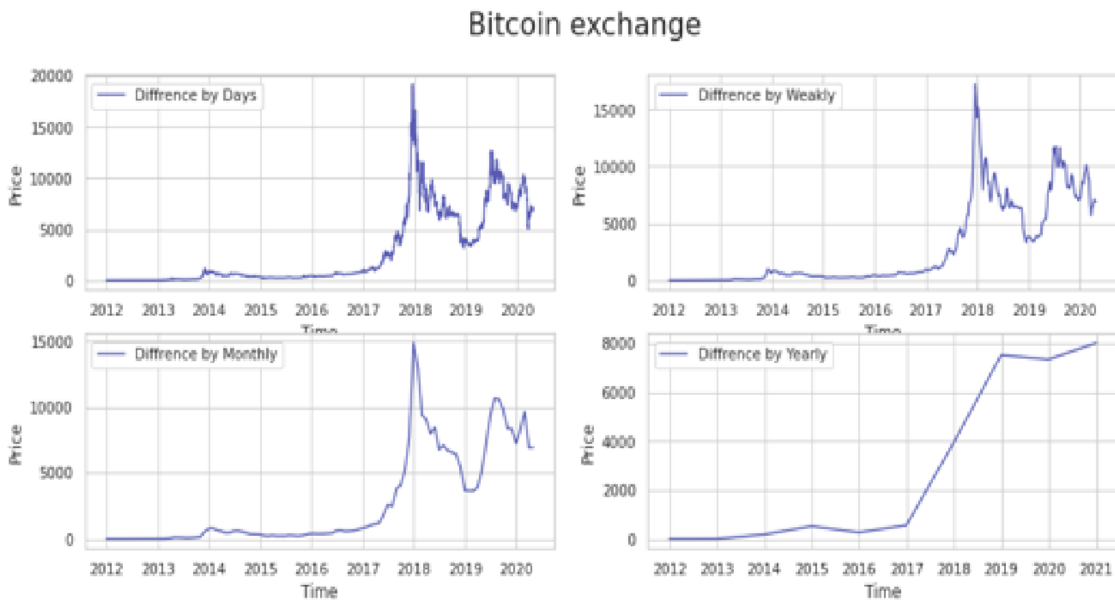


Fig 4.3

Histogram:- A histogram is a graphical display of data using bars of different heights. In a histogram, each bar groups numbers into ranges. Taller bars show that more data falls in that range. A histogram displays the shape and spread of continuous sample data.

```
fig = plt.figure(figsize=[17,7])
plt.suptitle('USING HISTOGRAM', fontsize=22)
plt.subplot(221)
plt.hist(data.Weighted_Price,density=True,histtype='bar',color="yellow",label="Diffrence by daily")
plt.legend(["Diffrence by daily"],loc="upper right")
plt.subplot(222)
plt.hist(data_week.Weighted_Price,density=True,histtype='bar',color="yellow",label="Diffrence by weekly")
plt.legend(["Diffrence by weekly"],loc="upper right")
plt.subplot(223)
plt.hist(data_month.Weighted_Price,density=True,histtype='bar',color="yellow",label="Diffrence by monthly")
plt.legend(["Diffrence by monthly"],loc="upper right")
plt.subplot(224)
plt.hist(data_year.Weighted_Price,density=True,histtype='bar',color="yellow",label="Diffrence by yearly")
plt.legend(["Diffrence by Yearly"],loc="upper right")
```

```
<matplotlib.legend.Legend at 0x7fd29a94da00>
```

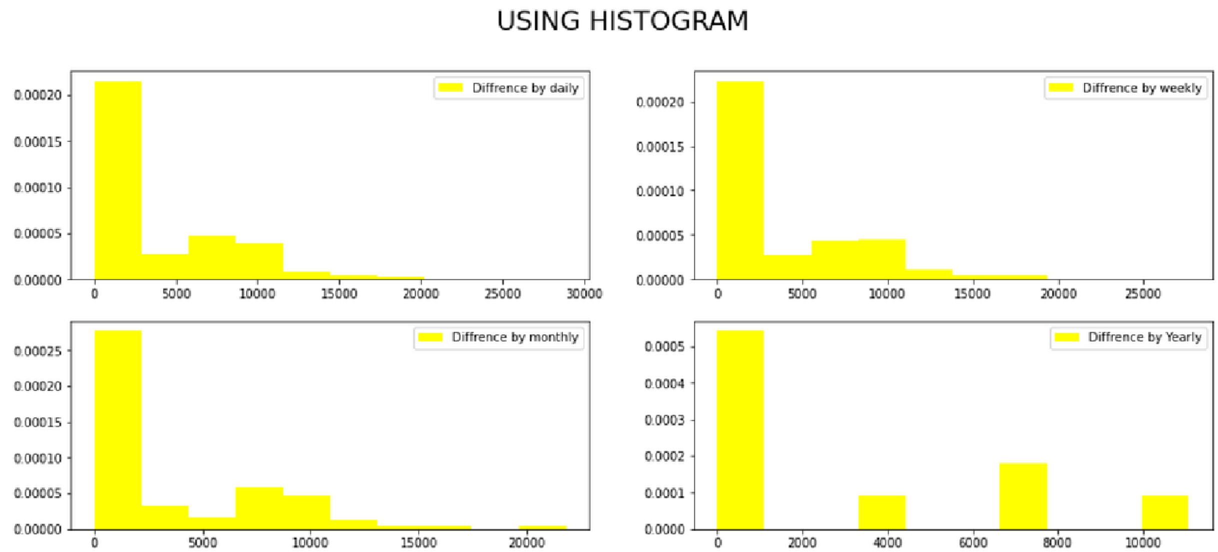


Fig 4.4

Bar:- A bar chart or bar graph is a chart or graph that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent. The bars can be plotted vertically or horizontally.

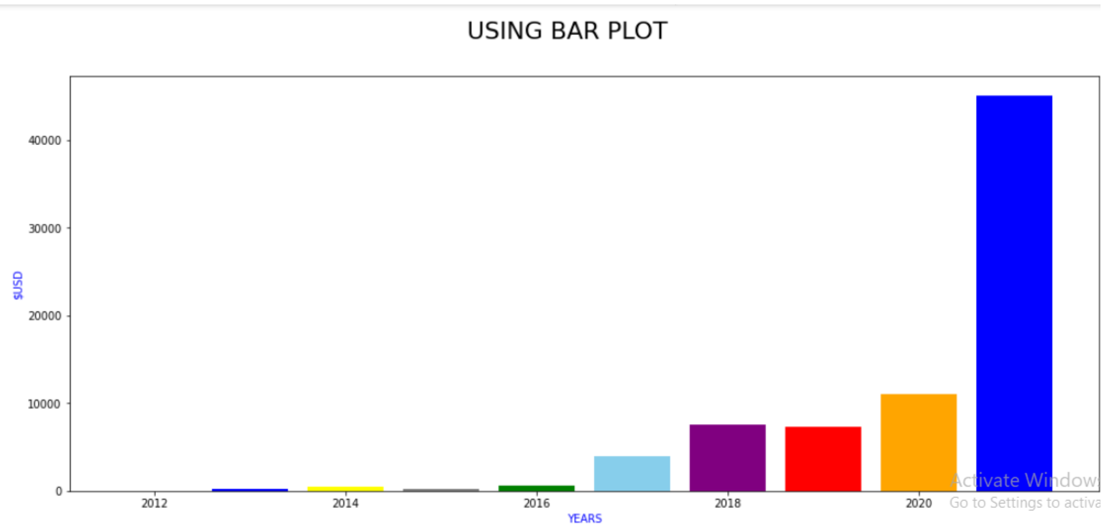


Fig 4.5

Step 4: **Dickey Fuller Test:**

In statistics, the stationarity tests such as the KPSS test that consider as null hypothesis H0 that the series is stationary, and unit root tests, such as the Dickey-Fuller test and its augmented version, the augmented Dickey-Fuller test (ADF), or the Phillips-Perron test (PP), for which the null

It is a common statistical test used to test whether a given Time series is stationary or not. It is one of the most commonly used statistical tests when it comes to analyzing the stationary of a series.

Stationarity is an important concept in time series analysis. Stationarity means that the statistical properties of a time series (or rather the process generating it) do not change over time. Stationarity is important because many useful analytical tools and statistical tests and models rely on it.

```
plt.style.use('seaborn-poster')
sns.set_style('whitegrid')
sns.set_context('talk')
st.tsa.seasonal_decompose(data_month.Weighted_Price).plot()
result=st.tsa.stattools.adfuller(data_month.Weighted_Price)[1]
print("Dickey Fuller test value of P is = %f"%result)
if result>0.05:
    print("The time series is not stationary")
else:
    print("The time series is stationary")
plt.show()
```

```
Dickey Fuller test value of P is = 0.998863
The time series is not stationary
```

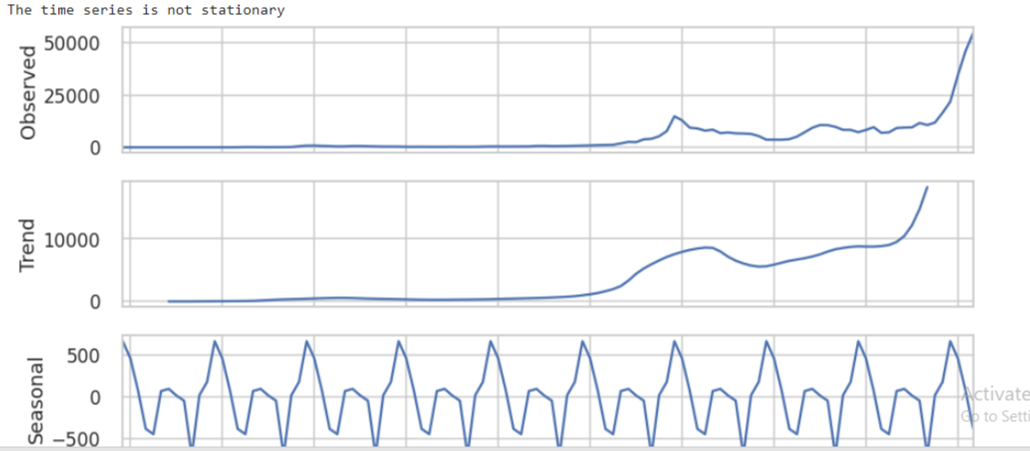


Fig 4.6

Step 5: Plot ACF and PACF

Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF). Since we are working with daily data, the ACF shows us which day in the past correlates the most with the current day with respect to the days in between. PACF shows us which day in the past correlates directly to the current day by ignoring the days in between.By knowing the PACF and ACF, we now better understand our dataset and the parameters to potentially choose.

```
plt.figure(figsize=(15,7))
ax = plt.subplot(211)
st.graphics.tsa.plot_acf(data_month.prices_box_diff2[13:].values.squeeze(), lags=48, ax=ax)
ax = plt.subplot(212)
st.graphics.tsa.plot_pacf(data_month.prices_box_diff2[13:].values.squeeze(), lags=48, ax=ax)
plt.tight_layout()
plt.show()
```

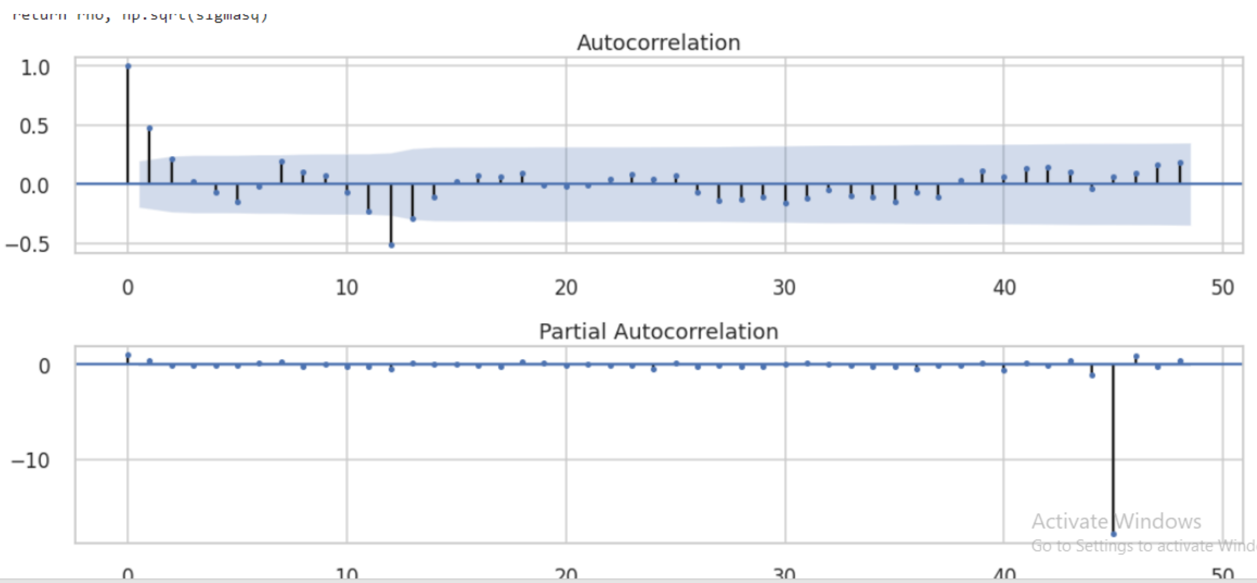
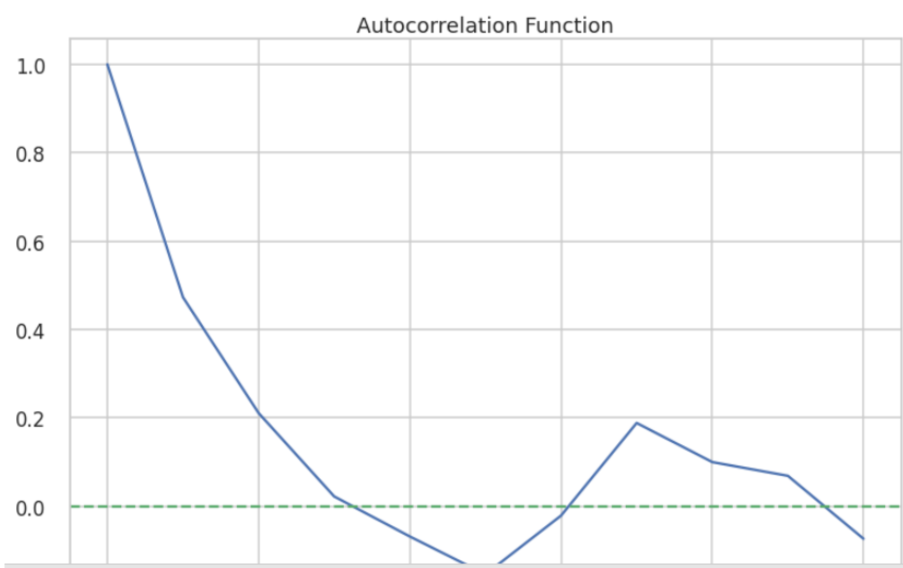


Fig 4.7

Step 6: Define the Model

Now, we can move on to modeling our data by using the SARIMA model.In order to get the best performance out of the model, we must find the optimum parameters. We do this by trying many different combinations of the parameters and selecting the one with the relatively lowest AIC score.

```
[ ] from statsmodels.tsa.stattools import acf, pacf
    from statsmodels.tsa.arima_model import ARIMA
    lag_acf = acf(data_month.prices_box_diff2[13:], nlags=10)
    lag_pacf = pacf(data_month.prices_box_diff2[13:], nlags=10, method='ols')
    plt.subplot(1,1,1)
    plt.plot(lag_acf)
    plt.axhline(y=0,linestyle='--',color='g')
    plt.title('Autocorrelation Function')
    plt.show()
    plt.subplot(1,1,1)
    plt.plot(lag_pacf)
    plt.axhline(y=0,linestyle='--',color='green')
    plt.title('Partial Autocorrelation Function ')
    plt.tight_layout()
    plt.show()
```



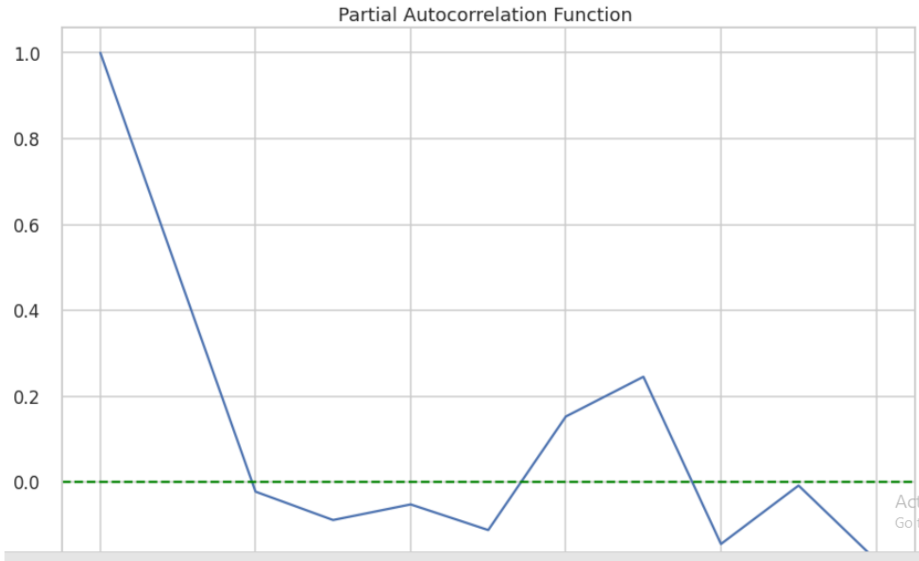


Fig 4.8

Step 7: Model Summary

In this step, we will see different statistical information of our model . Total no of observations are 112 and Lilelihood is -83.

```
[ ] result_table = pd.DataFrame(results)
    result_table.columns = ['parameters', 'aic']
    print(result_table.sort_values(by = 'aic', ascending=True).head())
    print(best_model.summary())
```

```
      parameters         aic
19  (1, 0, 0, 1)  173.616185
21  (1, 0, 1, 1)  174.766385
25  (1, 1, 0, 1)  175.546746
37  (2, 0, 0, 1)  175.553909
7   (0, 1, 0, 1)  175.589435
                                 Statespace Model Results
==========================================================================================
Dep. Variable:                 Weighted_Price_box   No. Observations:                  112
Model:             SARIMAX(1, 1, 0)x(0, 1, 1, 12)   Log Likelihood                 -83.808
Date:                            Tue, 21 Dec 2021   AIC                            173.616
Time:                                    03:48:32   BIC                            181.402
Sample:                                12-31-2011   HQIC                           176.766
                                     - 03-31-2021
Covariance Type:                              opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.4250      0.085      5.025      0.000       0.259       0.591
ma.S.L12      -0.9954      5.690     -0.175      0.861     -12.148      10.157
sigma2         0.2437      1.373      0.178      0.859      -2.446       2.934
===================================================================================
Ljung-Box (Q):                       28.07   Jarque-Bera (JB):                 3.59
Prob(Q):                              0.92   Prob(JB):                         0.17
Heteroskedasticity (H):               1.16   Skew:                             0.29
Prob(H) (two-sided):                  0.68   Kurtosis:                         3.73
```

Fig 4.9

Step 8: Training the model

For training our model first we splitted our dataset in a training and testing set by using scikit-learn model_selection.train_test_split. It is a function in Sklearn model selection for splitting data arrays into two subsets: for training data and for testing data. The train-test split is a technique for evaluating the performance of a machine learning algorithm. It can be used for classification or regression problems and can be used for any supervised learning algorithm. The procedure involves taking a dataset. The data are converted into train and test sets. The training data will include 75% of images and the remaining 25% was used to test the model. To avoid getting different values for train and test every time you select the random_state you want to use. This will also have an impact on the evaluation metrics as if the random_state is not selected the values from the evaluation will differ. The values 0, 1 are the most common values used. After splitting the dataset we used the Random Forest regressor algorithm.

```
[ ] data['Price_After_Month']=data['Close'].shift(-30)
```

```
data.dropna(inplace=True)
X=data.drop('Price_After_Month',axis=1)
X=preprocessing.scale(X)
y=data['Price_After_Month']
```

```
[ ] X_train,X_test,y_train,y_test=model_selection.train_test_split(X,y,test_size=0.3,random_state=101)
```

```
[ ] reg=RandomForestRegressor(n_estimators=200,random_state=101)
    reg.fit(X_train,y_train)
    accuracy=reg.score(X_test,y_test)
```

Fig 4.10

Step 9: Results

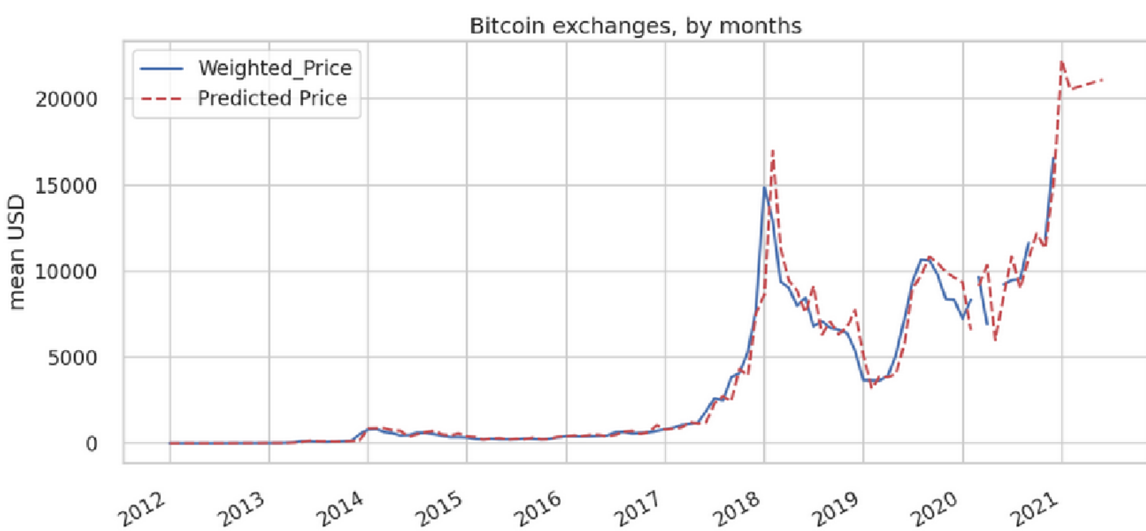
We got 93.3% accuracy and Predicted one bitcoin price.

```
reg=RandomForestRegressor(n_estimators=200,random_state=101)
reg.fit(X_train,y_train)
accuracy=reg.score(X_test,y_test)
accuracy=accuracy*100
accuracy = float("{0:.4f}".format(accuracy))
print('Accuracy is:',accuracy,'%')
```

```
Accuracy is: 93.2773 %
```

```
preds = reg.predict(X_test)
print("The prediction is:",preds[1],"But the real value is:" ,y_test[1])
```

```
The prediction is: 109.62722298600916 But the real value is: 103.83312362030892
```



Chapter 5

Conclusion and Scope for further work

5.1 Conclusion

In this work, we have implemented a model to predict real time bitcoin prices using various parameters such as High, low, open , volume etc. Thus drawing a comparison between the actual prices and the predicted prices by taking into consideration various time series data to ultimately reduce the difference between the predicted and actual prices, aiming to get the highest accuracy out of it using appropriate machine learning algorithms.

5.2 Scope for further work

The aim for the next semester will be to improve the performance of the system. This can be achieved by considering the points such as not overfitting on the dataset on the labels. Training our model for better price prediction thus improving the performance of the system. Also, trying out new architectures will help us compare the existing architecture. The goal will be to try out different algorithms along LSTM and Random forest. Finally, the system will be implemented as a web service.

Bibliography

[1] Issac Madan, Shaurya Saluja, Aojia Zhao, “Automated Bitcoin Trading via Machine Learning Algorithms”,Department of Computer Science , Stanford University, Stanford , 2015.

[2] Brian Vockathaler, “ The Bitcoin Boom: An In Depth Analysis of The Price Of Bitcoins”, Thesis, University Of Ottawa, Ontario, Canada, June 2017.

[3] D. Shah and K. Zhang, “Bayesian regression and Bitcoin,” in 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton), 2015, pp. 409-415.

[4] Huisu Jang and Jaewook Lee, “An Empirical Study on Modelling and Prediction of Bitcoin Prices with Bayesian Neural Networks based on Blockchain Information,” in IEEE Early Access Articles, 2017.

[5] Siddhi Velankar, Sakshi Valecha , Shreya Maji, “Bitcoin Price Prediction using Machine Learning”, International Conference onAdvanced Communications Technology(ICACT) ,Dept of Electronics and Telecommunication, Pune Institute of Computer Technology, Pune Maharashtra, India,February 2018.

[6]https://towardsdatascience.com/using-machine-learning-to-predict-future-bitcoin-prices-6637e7bfa58f

[7] https://arxiv.org/ftp/arxiv/papers/2006/2006.14473.pdf

[8] https://www.irjet.net/archives/V6/i12/IRJET-V6I12344.pdf

[9] https://medium.com/activewizards-machine-learning-company/bitcoin-price-forecasting-with-deep-learning-algorithms-eb578a2387a3

<u>Acknowledgement</u>

We would like to thank K.J. Somaiya College of Engineering for giving us this opportunity to work on this project. We express our gratitude towards **Guide – Prof. Nilkamal More** and **Co-Guide – Prof. Ruchira Jadhav** for their guidance for giving us this opportunity and their valuable guidance. We would like to thank Project coordinators for allowing us to perform the project and helping us to select the related topics and projects. We would also like to thank **Principal Dr. Shubha Pandit** for creating such opportunities. Each of us have studied the concept through various information sources and tried our best to make this project successful.