A Project Report

on

Bitcoin Price Prediction Using Machine Learning

carried out as part of the ***Minor Project IT3270*** *Submitted*

by

Saksham Agrawal
199302007

Varad Sharma
199302010

in partial fulfilment for the award of the degree of

Bachelor of Technology

in

Information Technology

School of Computing and Information Technology
Department of Information Technology

MANIPAL UNIVERSITY JAIPUR
JAIPUR-303007
RAJASTHAN, INDIA

May 2022

CERTIFICATE

Date:24/05/2022

This is to certify that the minor project titled **Bitcoin Price Prediction Using Machine Learning** is a record of the bonafide work done by **Saksham Agrawal** (199302007), **Varad Sharma**(199302010) submitted in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in Information Technology of Manipal University Jaipur, during the academic year 2021-22.

Dr. Sudhir Sharma
Project Guide, Department of Information Technology
Assistant Professor (Senior Scale)
Manipal University Jaipur

Dr. Pankaj Vyas
HoD, Department of Information Technology
Manipal University Jaipur

ABSTRACT

Bitcoin is a form of decentralized cryptocurrency which can be used for transaction without need for intermediaries using the peer-to-peer bitcoin network. These transactions are verified by network nodes through cryptography and then recorded in a public distributed ledger called blockchain. These Bitcoins are generated by the process of mining. Bitcoin is often said to be very volatile in nature. Volatility in finance is the degree of variations of trading price series over time, usually measured by standard deviation or logarithmic returns. The price of bitcoin has been through cycles of appreciations and depreciations over its lifetime, often referred to as bubble. Bitcoin is said to be more than 7 times more volatile than gold, 18 times more volatile than USD, which is the globally accepted currency for international trade. This volatility has often put traders and investors at a distance from bitcoin who don't want to risk a large sum of money. To solve this issue of volatility and predict the future prices of bitcoin, this aims to build machine learning and deep learning models in order to predict future prices of bitcoin to the highest possible degree of accuracy.

LIST OF TABLES

LIST OF FIGURES

CONTENTS

1. Introduction

Bitcoin a highly valued digital asset based on the technology of cryptography. Unlike normal currency, Bitcoin is entirely created, traded, distributed and stored by help of digitalized ledger system, known as blockchain. Thus, it uses the technology of cryptography to keep it secure. All transactions are verified by a massive amount of computing via the process of mining, keeping the public ledger secure and impossible to manipulate in any manner. Despite being not a legal tender across the world, bitcoin is extremely popular and the most popular form of cryptocurrency. This huge popularity makes the currency extremely volatile. It has gone through many cycles of bubbles and bust, despite being in existence for a very short period of time. Thus, the traders and investors may want to avoid investing in bitcoin to reduce the risk. In order to solve this issue, we want to develop machine learning and deep learning models which can predict the price of bitcoin. Predicting the price of bitcoin by help of these algorithms helps us to make better decisions about investing.

Therefore, in this project we plan on implementing 3 different price prediction models to find out which model is most suitable for the prediction. The models selected are – LSTM (Long Short Term Memory) based on Recurrent Neural Network, SVR (Support Vector Regression) and ARIMA (AutoRegressive Integrated Moving Average).Here, we use the 'Close' Price for each day.

1.1 **Problem Statement**

In this project, we aim to implement multiple deep learning and machine learning regression techniques for prediction of Bitcoin to find out which one achieves better results.

1.2 **Objectives**

Following are the objectives for this project-

a. To study and understand the nature of bitcoin and various machine learning models or algorithms that are associated with bitcoin.

b. To analyse, study and understand various machine learning models.

c. To analyse and collect various datasets for bitcoin price prediction.

d. To apply and compare suitable machine learning model for better price prediction of bitcoin.

1.3 **Scope of Project**

In this project, we aim at achieving better accuracy for our models to predict. However, due to the volatility and nature of bitcoin, prediction its accurate value over a long course of time proves to be difficult. To overcome this issue, we have implemented machine learning and deep learning by help of LSTM and SVR, which are 2 very powerful regression models.

2. Background Detail

2.1 Conceptual Overview / Literature Review

There have been other attempts at this problem with varying level of success. Many of these have an attempt to build 2 or more models and try to compare their performances based on accuracy and errors. Also, some researches tried to understand the volatility of Bitcoin prices and what causes them to fluctuate as much as they do, what can be done to overcome this sort of volatility. Also, in some cases we see that classification is being used at varying level in order to predict if prices shall fall or rise.

For example, in [1], the research aim to achieve highest accuracy model to predict Bitcoin's price by the use of various machine learning models. This research attempts to apply various models such as Theil-Sen regression and Huber regression, which are linear regression models for prediction; LSTM(Long Short Term Memory, a sequential model implemented using Recurrent Neural Network, designed to take in inputs sequentially and predict the next output based on both current input and previous output; GRU(Gated Recurrent Unit) which is a special implementation of LSTM. The model focuses on 1 day interval trading exchange data of Bitcoin. In [2], the research aims to predict the Bitcoin price by conducting a survey on parameters that affect bitcoin value. The research attempts at building 2 models, a Linear Regression Model. In [3], demonstration of high performance machine learning and deep learning models have been used. Classification models try to predict whether the prices go up or down for next time series value, with models such as ANN and SVM being prominently used while Regression models are used to predict price for next time series value with LSTM as main model along with various models such as SANN and ANN, with SANN having much more fluctuations. Used methods for metrics purpose were RMSE, MAE and MSE. As for classification, F1 precision recall has been used for accuracy measurement along with confusion matrix. Algorithms used here are ANN with SVM. For research done in [4], ARIMA model is the main focus for prediction of prices over a time period of 3 years and transform the dataset to improv ethe performance of model. In case of [5], the research tires to understand the dynamics of bitcoin returns with the help of neural networks and technical analysis along with density forecasting. The research tries to understand the volatility of Bitcoin. For paper[6], daily prices of bitcoin were considered. To scale the data, various forms of methods like standard deviation normalization, z-score normalization log normalization were considered among others. For prediction purposes the models proposed were Random Forest and Bayesian Network. For paper [7], time series model of ARIMA has been used along with advanced and specialized algorithms such as Facebook PROPHET for prediction of Bitcoin Prices. PROPHET is a forecasting algorithm implemented on Python and R and is completely automated.

3. **System Design & Methodology**

3.1. *System Architecture*

The following block diagram demonstrates the procedure followed during the entire project.

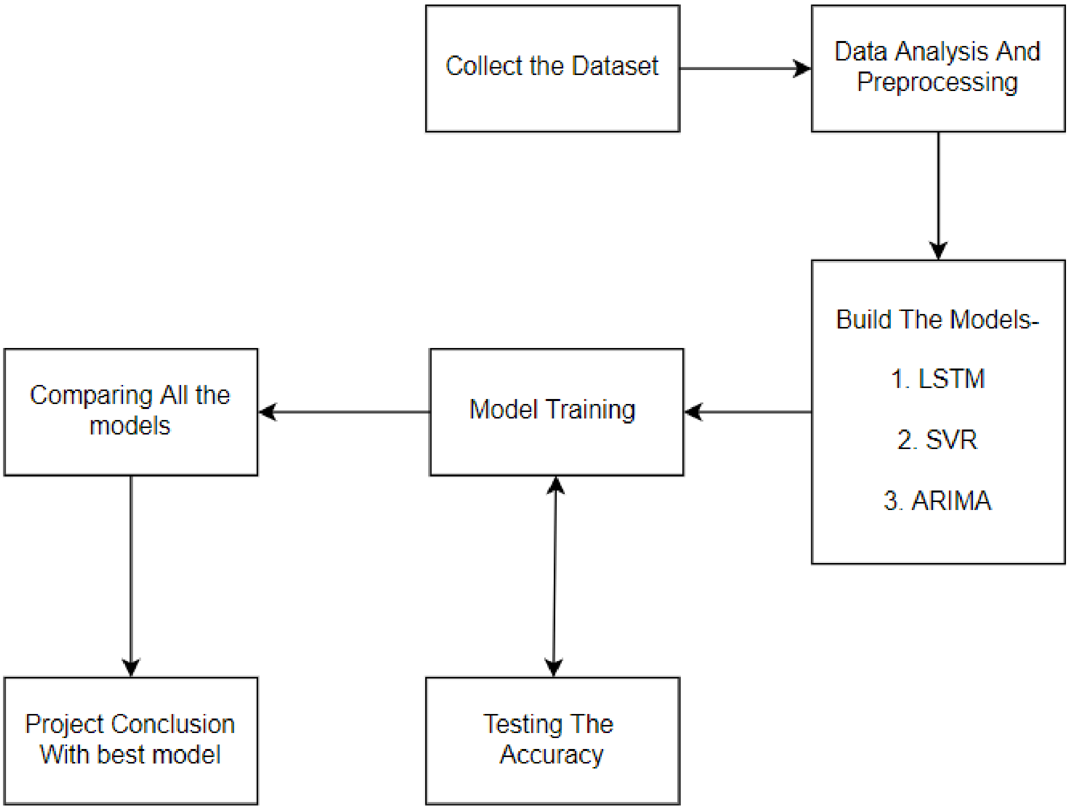


Figure 1:Project Architecture

3.2 *Development Environment.*

hardware – The model is developed on system with 4 core /8 threads processor running at 2.3 Ghz. Memory of the system is 16 GB DDR4.

software – The OS where system works on is Windows 11 OS. Language used is python 3.8 Environment used is Google Colab which has all necessary libraries pre-installed, up-to-date and ready to use which we need in this project. Various Libraries were used in this project such as Numpy, Pandas, Matplotlib, Sklearn, Plotly, TensorFlow etc.

3.3. *Methodology: Algorithm/Procedures*

For the implement of our proposed idea, the project is divided into following important steps-

a) **Data Collection And Literature Research** – This is the beginning of our project. In this step, we aim to understand and define the problem statement along with objectives we need to achieve. One aspect of understanding the project is to understand and study the past attempts at prediction. Also we collect the dataset we need for prediction purposes. We get our Dataset from Kaggle with following properties-

Shape of Data – 2739 Rows x 7 Columns
Time Period Covered– 17 Sep 2014 – 17 Mar 2022 (2739 Days or 7.5 Years)
Currency used to refer prices of Bitcoin – USD (United States Dollar)

b) **Preparation and Analysis of Data** – This is a crucial step in our project as this step determines what is the data, we train our model on. We try to understand the dataset, its properties, patterns and perform exploratory data analysis. This procedure is preformed in following steps-

- Analysis of our data enables us to understand various statistic features of dataset along with some patterns. Thus analysis would help us reach informed conclusions and understandings of data. Some of the statistics of the dataset are as follows-
  We can see the historic price of Bitcoin (fig 2)of the entire dataset.
  We see that near the end of 2017 and 2020, there were massive fluctuations in the price of bitcoin.

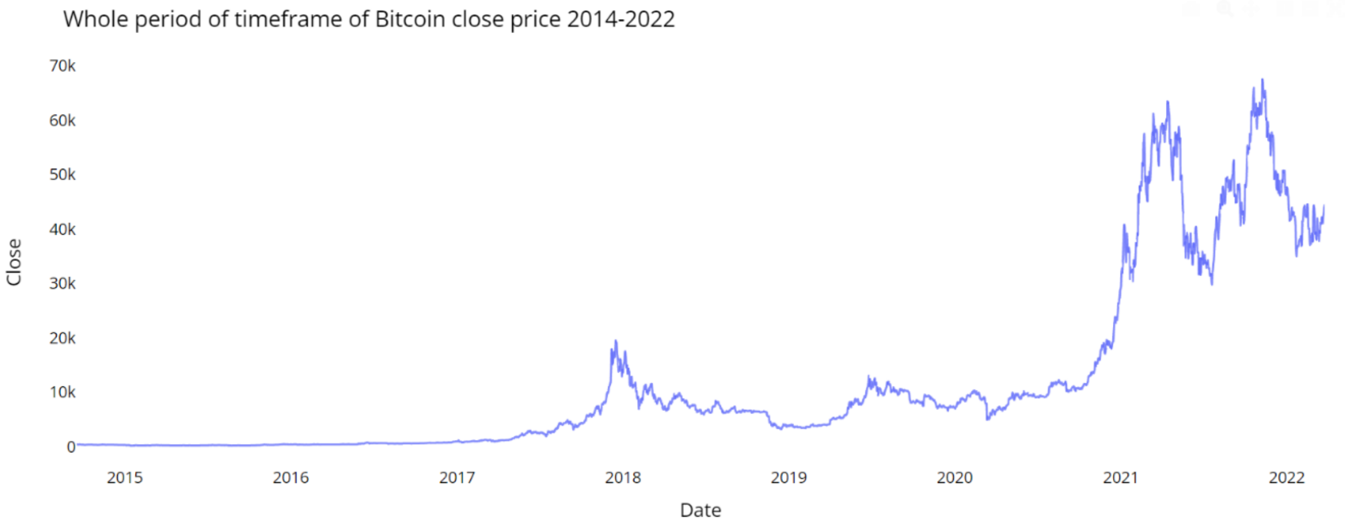
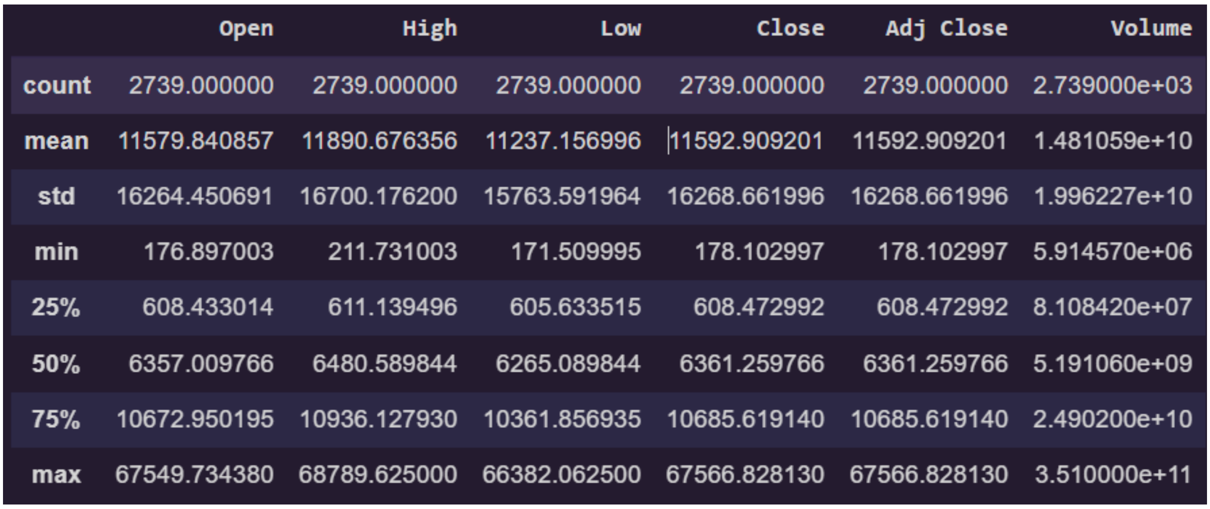


Figure 2: Historical prices of Bitcoin

Features	Definition
Open	Opening price of Bitcoin at particular date
High	Highest price of bitcoin at the date
Low	Lowest price of bitcoin at the day
Close	Price at which trading of bitcoin closed
Adj Close	Adjacent Close Value
Volume	Amount of Bitcoins traded that day
Date	Date at which prices have been recorded

Table 1: About various attributes in Dataset

Below what we see(fig 3) are some of the statistical values of Bitcoin. We see that there are some massive differences in the min and max prices for a period of just over 7 years, with maximum values growing over more than 350 times. Also we see that more than 75% of prices of bitcoin are values at lower than a quarter of its all time high. All these values are an indication of volatility of Bitcoin. Volatility of Bitcoin makes it difficult for the prediction.



	Open	High	Low	Close	Adj Close	Volume
count	2739.000000	2739.000000	2739.000000	2739.000000	2739.000000	2.739000e+03
mean	11579.840857	11890.676356	11237.156996	11592.909201	11592.909201	1.481059e+10
std	16264.450691	16700.176200	15763.591964	16268.661996	16268.661996	1.996227e+10
min	176.897003	211.731003	171.509995	178.102997	178.102997	5.914570e+06
25%	608.433014	611.139496	605.633515	608.472992	608.472992	8.108420e+07
50%	6357.009766	6480.589844	6265.089844	6361.259766	6361.259766	5.191060e+09
75%	10672.950195	10936.127930	10361.856935	10685.619140	10685.619140	2.490200e+10
max	67549.734380	68789.625000	66382.062500	67566.828130	67566.828130	3.510000e+11

Figure 3 Statistical Data analysis of Bitcoin

- Data Preparation – This is a crucial step in our entire project. Here we have to transform our data so that it can fit in our models to train and also find relevant features from entire dataset. We need to prepare the data so that our model is able to perform time series prediction on the same. For this purpose need to specify the time horizon. Here we have set time frame of 5 days with regular intervals of 1 day as time frame for out prediction. The data also needs to be split in order to find its accuracy and error values. Thus, we decided to go for 70:30 split for train: test data.

- Data Scaling – This step enables us to scale the data and adapt it for our model. The reason we do scaling is to reduce any chances of error due to outliers and inconsistency in data. In this project, we have implemented the Minimax normalization as it is a widely used method to scale data for machine learning purposes. Feature range for normalization is 0,1

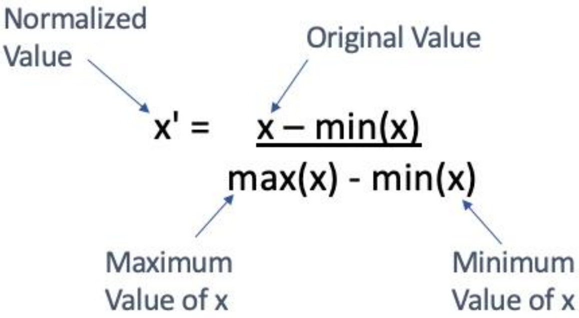
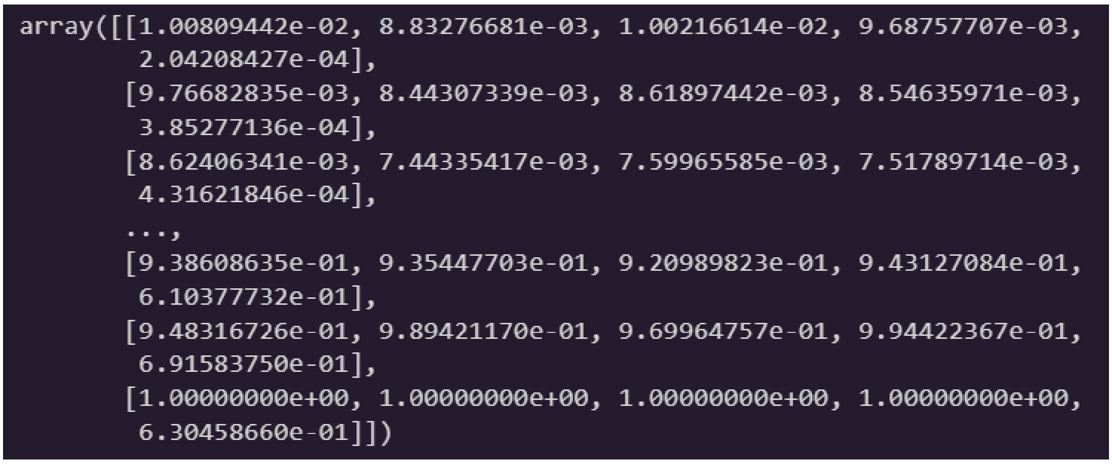


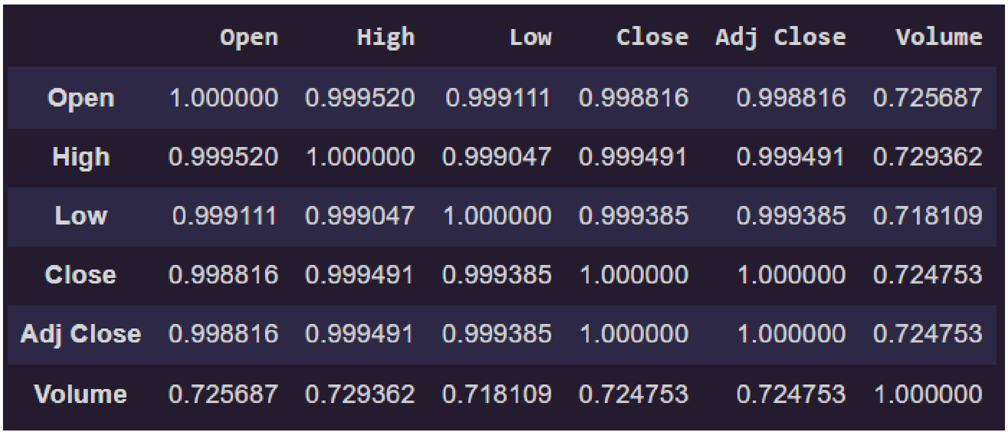
Figure 4 Minimax normalization Formula



```
array([[1.00809442e-02, 8.83276681e-03, 1.00216614e-02, 9.68757707e-03,
         2.04208427e-04],
        [9.76682835e-03, 8.44307339e-03, 8.61897442e-03, 8.54635971e-03,
         3.85277136e-04],
        [8.62406341e-03, 7.44335417e-03, 7.59965585e-03, 7.51789714e-03,
         4.31621846e-04],
        ...,
        [9.38608635e-01, 9.35447703e-01, 9.20989823e-01, 9.43127084e-01,
         6.10377732e-01],
        [9.48316726e-01, 9.89421170e-01, 9.69964757e-01, 9.94422367e-01,
         6.91583750e-01],
        [1.00000000e+00, 1.00000000e+00, 1.00000000e+00, 1.00000000e+00,
         6.30458660e-01]])
```

Figure 5 normalized dataset

- Correlation Among Features (fig 6) Another crucial step in our project is to understand the correlation among the features . At this stage, we intend to select only those feature sets which are most useful for our prediction purpose. It is



	Open	High	Low	Close	Adj Close	Volume
Open	1.000000	0.999520	0.999111	0.998816	0.998816	0.725687
High	0.999520	1.000000	0.999047	0.999491	0.999491	0.729362
Low	0.999111	0.999047	1.000000	0.999385	0.999385	0.718109
Close	0.998816	0.999491	0.999385	1.000000	1.000000	0.724753
Adj Close	0.998816	0.999491	0.999385	1.000000	1.000000	0.724753
Volume	0.725687	0.729362	0.718109	0.724753	0.724753	1.000000

Figure 6: Feature Correlation

desirable to reduce the number of input variables so that we can both reduce the computational cost of modeling and, in some cases, to improve the performance of the model. Feature correlation helps us in better understanding the relationship between the features of our dataset.

c) **Design of the models used** – For Modelling we chose regression and time-series based methods because we aim to predict next price Bitcoin based on previous price. With the help of libraries such as sklearn and tensorflow.keras, we selected SVR (Support Vector Regression), a powerful regression model and a deep learning model LSTM (Long-Short Term Memory), an implementation of RNN based Neural Network and also ARIMA, (AutoRegressive Integrated Moving Average), a model based around the concept of time series prediction.

- LSTM (Long Short-Term Memory)

LSTM(fig 4) is a powerful recurrent neural network architecture used to implement in fields of Deep Learning. LSTM is a great model where can data can be fed sequentially in the model. They are capable of learning order dependence in sequence prediction problems. This enables LSTM to work in complex domains such as machine translation, speech generation and more .

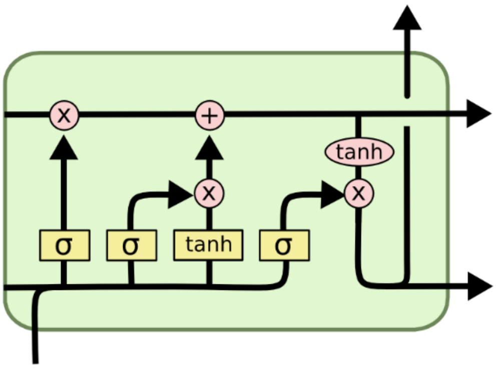


Figure 7 LSTM node architecture

LSTM Parameters are set as-

Epochs	200
Neurons	256
Layers	2
Batch Size	64
Activation Function	relu

- SVR (Support Vector Regression)

Support Vector Regression gives us the flexibility to define how much error is acceptable in our model and will find an appropriate line (or hyperplane in higher dimensions) to fit the data.

The objective of the support vector regression is to find a hyperplane in an N dimensional space (Where N is the number of features) that distinctly classifies the data points.

Parameters taken into consideration here are –

Kernel	‘rbf’
C	1000
Gamma	0.15

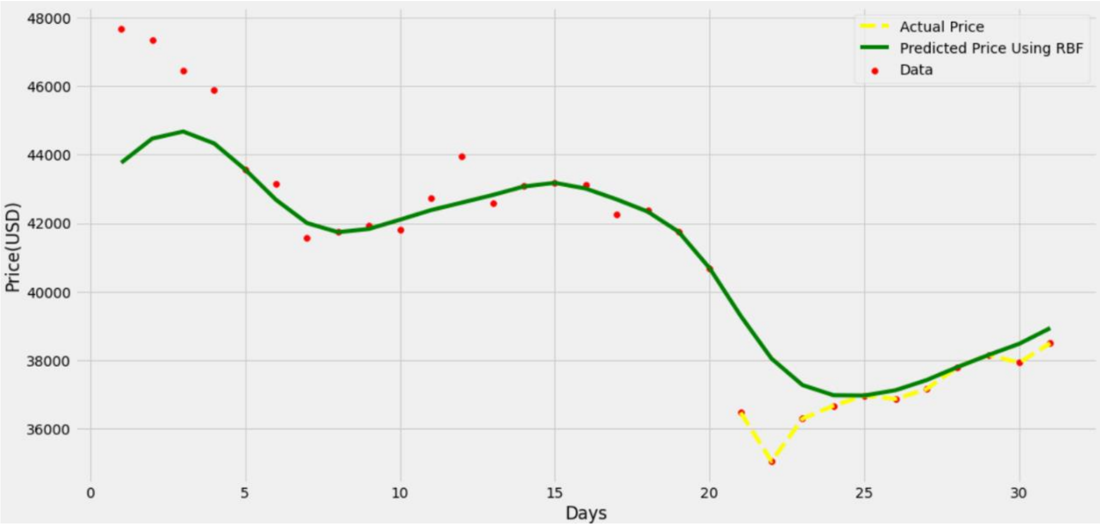


Figure 8 Comparison Among Various SVR Parameters

- ARIMA (Auto Regressive Integrated Moving Average)

An autoregressive integrated moving average, or ARIMA, is a statistical analysis model that uses time series data to either better understand the data set or to predict future trends. A statistical model is autoregressive if it predicts future values based on past values. The “AR” stands for autoregression, which refers to the model that shows a changing variable that regresses on its own prior or lagged values. “MA” represents the moving average, which is the dependency between an observed value and a residual error from a moving average model applied to previous observations. “I” stands for integrated, which means it observes the difference between static data values and previous values.

Values of p q d used are – 2, 1, 0

d. **Evaluation, training and testing** – At this stage, we train the model and test it to make sure it is working properly and as expected. We try to evaluate the model on basis of its behavior and fine tune the necessary parameters which define the model architecture. These parameters are referred to as hyperparameters and thus this process of searching for the ideal model architecture is referred to as hyperparameter tuning.

4. Implementation and Result

4.1. Modules/Classes of Implemented Project

The project has been divided into 4 major modules for easier functionality. They are as follows-

a. Data analysis Module – In this section, we take the entire dataset and try to understand various aspects of the same. From general shape of dataset, checking for null values and exploring the values in dataset, to correlation analysis of various features of dataset, many operations have been performed.

b. LSTM model implementation – In this section, we perform the process of training the dataset on LSTM architecture based on RNN.

c. SVR model implementation – In this section, we perform the process of training on the dataset on SVR model

d. ARIMA model implementation- In this section, we perform the process of training on the dataset on ARIMA model.

4.2. Implementation Detail

Following table describes the various metric values that have been used in order to determine which of the models performs best.

Table 2: Result Metrics for all models

Method implemented	MSE	RMSE	MAE	R2 value
LSTM Unscaled	61907363	7868	7141	-1.65
LSTM Scaled	0.0027	0.052	0.041	0.799
ARIMA	0.0073	0.085	0.069	0.753
SVR	0.0106	0.103	0.076	0.719

As the table shows, the LSTM model performs the best with significantly less errors as compared to others such as SVR and ARIMA models. The metric used to find best model is R2 value or coefficient of determination. For reference, we have also considered the LSTM values when data is not scaled so as to understand and determine what significance normalization of data holds and how it helps the model to be trained better.

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{\left[n\sum x^2 - (\sum x)^2\right]\left[n\sum y^2 - (\sum y)^2\right]}}$$

Figure 9: R2 value formula

4.3 Results and Discussion

As we have seen above, the LSTM model performs the best with significantly less errors as compared to others such as SVR and ARIMA models. This is due to the fact that RNN is a powerful neural network which is meant to understand the underlying patterns of data. While ARIMA and SVR are power models, with ARIMA being built especially for time series regression, it does not hold up well in case of very volatile predictions as there is no form of seasonality or recurrence. SVR, is not able to handle the extreme outliers or spikes which is part of nature of bitcoin.

While LSTM is able to perform the best, it too might suffer from these issues. But, with the level of hyperparameter tuning available for LSTM where we can change no of neurons, to layers, to layers in each neuron, learning rate, activation function, etc., we can significantly improve its performance.

The current model too can be improved upon with help of hyperparameter tuning, as the landscape changes and bitcoin values fluctuate, hyperparameter tuning is the way to ensure that our model always stays up to date and able to predict latest upcoming prices.

Below we have is a test graph for LSTM. While we see that the model works pretty well for most part, it does not hold up quite well in case of extreme spikes found near November 2021. But, for most part, LSTM does handle pretty well.

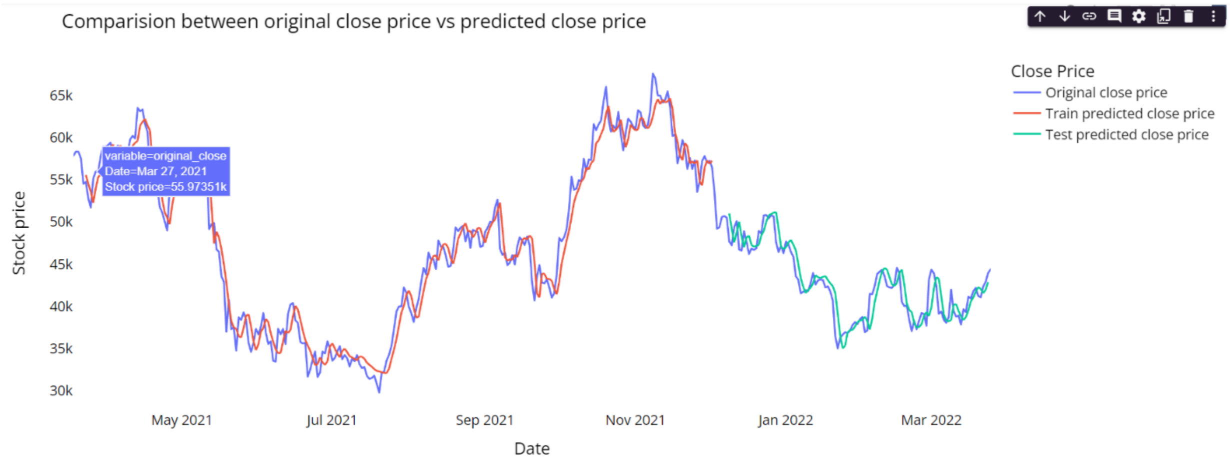


Figure 10: Train and test result for bitcoin

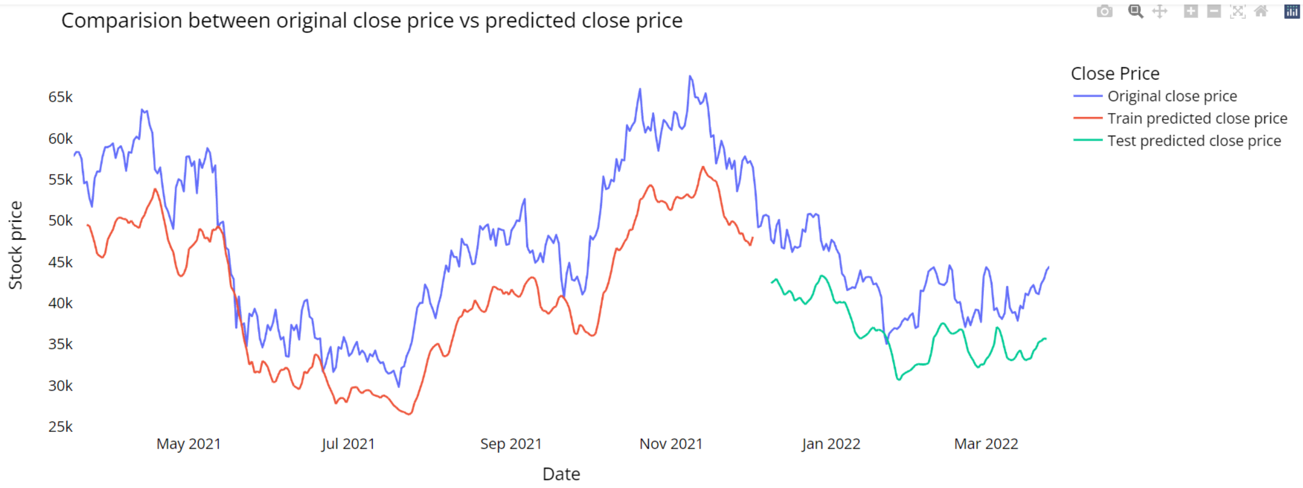


Figure 11 Results for Unscaled Dataset using LSTM

Below we have graph for ARIMA model which had issues with overfitting of dataset. We can see that the prediction follows exactly as the original price. This can be an issue for future prediction as the model would catch all noise in dataset, reducing the accuracy of model and thus, predicting wrong outputs.

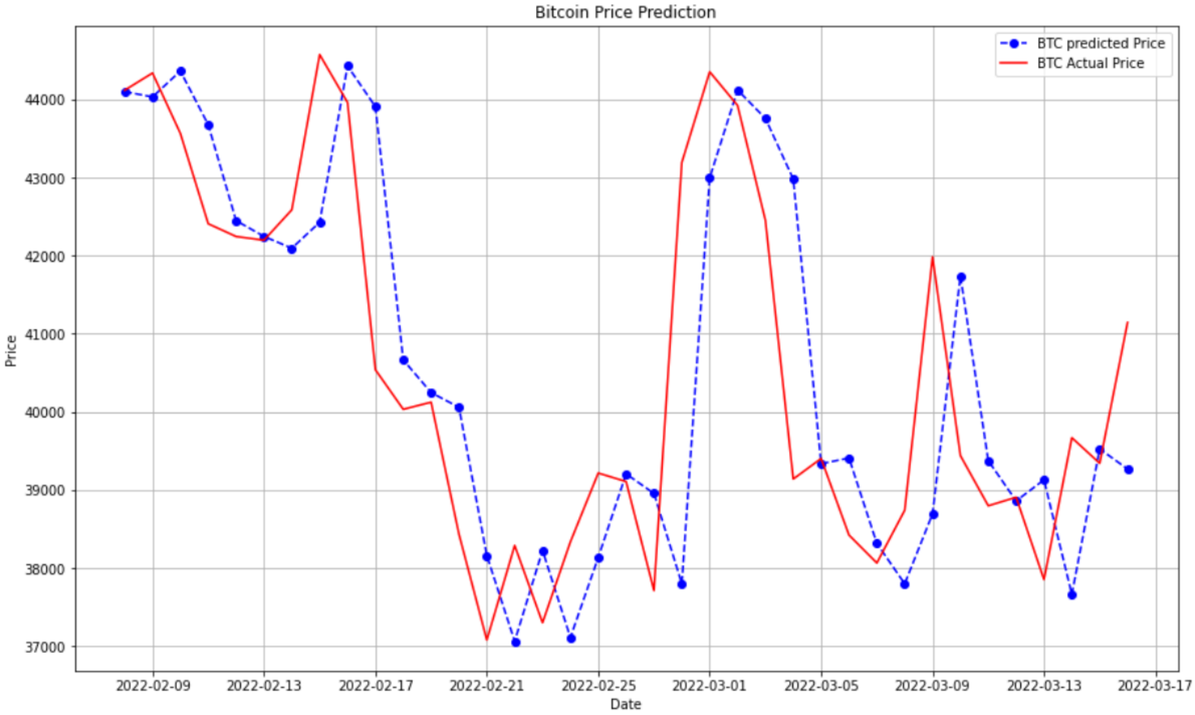


Figure 12 Overfitting example for ARIMA model

4.4 Month wise plan of work

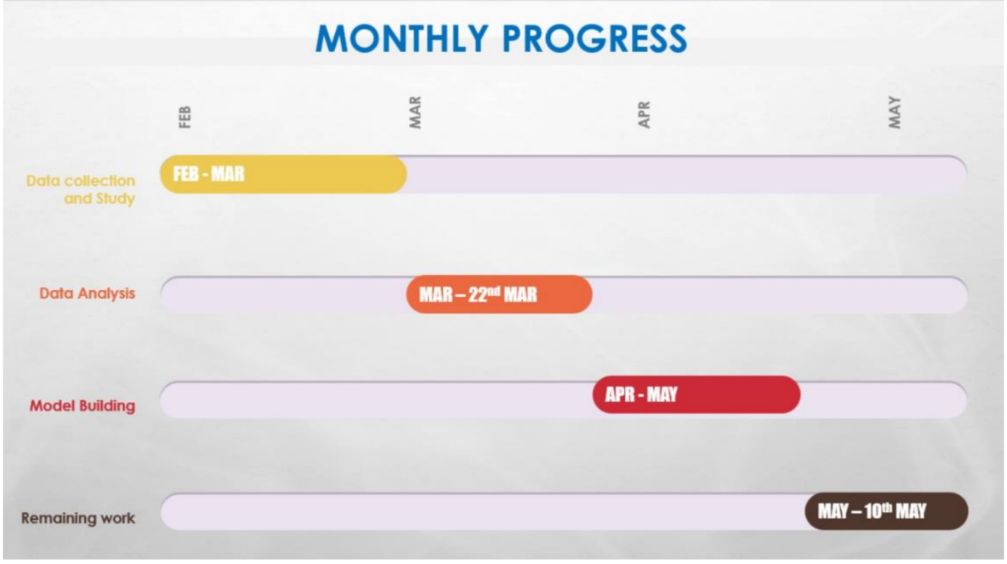


Figure 13 Month Wise Progress of report

5. Conclusion and Future Plan

In Conclusion we can say that Bitcoin is one of the most important aspects of Cryptocurrency world and future prediction of its price is of utmost importance given the volatility it holds. In this project, we explored various machine learning algorithms and deep learning models to determine which is the most suitable method for prediction and found out that Recurrent Neural Network Along With Long Short Term Memory based architecture provides the best results. With the help of Parameter Tuning, we were also able to improve the accuracy of this model.

Future Plan - Bitcoin Price prediction is a topic which is always an influential one because of the aspect of Bitcoin and the fact the Bitcoin is the most important form of cryptocurrency in the world and demonstrates how cryptocurrency and blockchain are one of the most important topics for the world of finance. While we were able to accurately measure the prices of this Cryptocurrency, we mostly focused on the aspect of regression-based modelling. Another aspect as to future prediction to future prices can be done with help of Classification models, especially with the help of CNN or Convolutional Neural Network or Support Vector Machine, which could determine whether the price does go up or down. These type of classification as to whether price goes up or down may not be as accurate as one needs but could work well with cryptocurrency such as bitcoin which has huge amount of volatility.

REFERENCES

[1] Phaladisailoed, T., & Numnonda, T. (2018, July). Machine learning models comparison for bitcoin price prediction. In 2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE) (pp. 506-511). IEEE.

[2] Pandey, M. S., Chavan, M. A., Paraskar, M. D., & Deore, S. (2021). Bitcoin Price Prediction using Machine Learning.

[3] Mudassir, M., Bennbaia, S., Unal, D., & Hammoudeh, M. (2020). Time-series forecasting of Bitcoin prices using high-dimensional features: a machine learning approach. Neural computing and applications, 1-15.

[4] Azari, Amin. "Bitcoin price prediction: An ARIMA approach." *arXiv preprint arXiv:1904.05315* (2019).

[5] Adcock R, Gradojevic N (2019) Non-fundamental, non-para-metric Bitcoin forecasting. Physica A: Stat Mech Appl531:121727

[6] Velankar, S., Valecha, S., & Maji, S. (2018, February). Bitcoin price prediction using machine learning. In 2018 20th International Conference on Advanced Communication Technology (ICACT) (pp. 144-147). IEEE.

[7]

[8] ChristosChristofidis (30 Nov 2020) “Awesome Deep Learning” https://github.com/ChristosChristofidis/awesome-deep-learning

[9] Source of Dataset - https://www.kaggle.com/datasets/varpit94/bitcoin-data-updated-till-26jun2021

Appendix

LSTM Code for model –

```
model = Sequential()
model.add(LSTM(units = 128, activation = 'relu', return_sequences = True, input_shape = (None,1)))
model.add(Dropout(0.2))
model.add(LSTM(units = 128, activation = 'relu'))
model.add(Dropout(0.2))
model.add(Dense(units =1))
model.summary()
```

SVR code for model –

```
#create and train SVR model using RBF kernel
from sklearn.svm import SVR


regressor = SVR(kernel='rbf', C=1000.0, gamma=0.15)
regressor.fit(days_train, close_train)
```

ARIMA c0de for model –

```
  model = ARIMA(train_data, order = (4,1,0))
  model_fit = model.fit()
  output = model_fit.forecast()
```