Bitcoin Price Prediction

Kartik Chhipa
chhipa.1@iitj.ac.in
Indian Institute of Technology, Jodhpur
Pattern Recognition and Machine Learning (CSL2050)
Professor: Richa Singh
richa@iitj.ac.in

Introduction

Bitcoin is a decentralized digital currency, without a central bank or single administration, that can be sent from users to users on peer to peer bitcoin network without the need of intermediaries. The dataset used in the project consist of the historical data of bitcoin. The dataset consist of attributes like 'Date', 'Open', 'Close', 'High', 'Low', 'Adj Close' and 'Volume'.

1. Open – refers to the opening price of the crypto for the day
2. High – refers to the highest price of the crypto achieved in a particular day
3. Close – refers to the cash value of the last transacted price of the crypto before the close of the day
4. Low – refers to the lowest price of the crypto achieved in a particular day
5. Volume – refers to the total number of crypto being exchange in a day
6. Adjusted Close – refers to the closing price of the crypto after accounting for any corporate actions

Types of Regressors Used

Decision Tree Regression

Decision Tree Regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete.

XGBoost Regression

XGBoost is an efficient implementation of **gradient boosting** that can be used for regression predictive modelling. Regression predictive modelling problems involve predicting a numerical value such as height and etc. Gradient Boosting refers to a class of ensemble learning algorithms that can be used for classification or regression predictive modelling. Ensembles are constructed from decision tree models. Trees are added one at a time and fit to correct the prediction errors made by prior models. Models are fit using gradient descent optimization algorithm.

Random Forest Regression

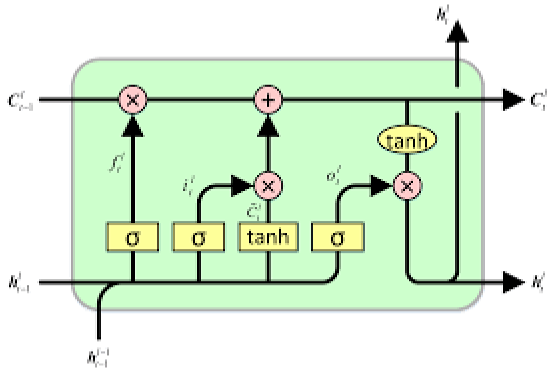
Random Forest Regression is also an ensemble learning method for regression. One of the important difference between XGBoost and Random Forest is that XGBoost always give more importance to functional space when reducing the cost of a model while Random Forest tries to give more preference to hyperparameters to optimize the model. Random Forest build and calculate each decision tree independently while in gradient boosting decision trees are built additively, in other words each decision tree is built one after the another.

LightGBM Regression

LightGBM is also a gradient boosting framework based on decision tree. The fundamental difference between LightGBM and XGBoost lies in the way they build the decision tree. In XGBoost trees grow depth-wise while in LightGBM, trees grow leaf-wise.

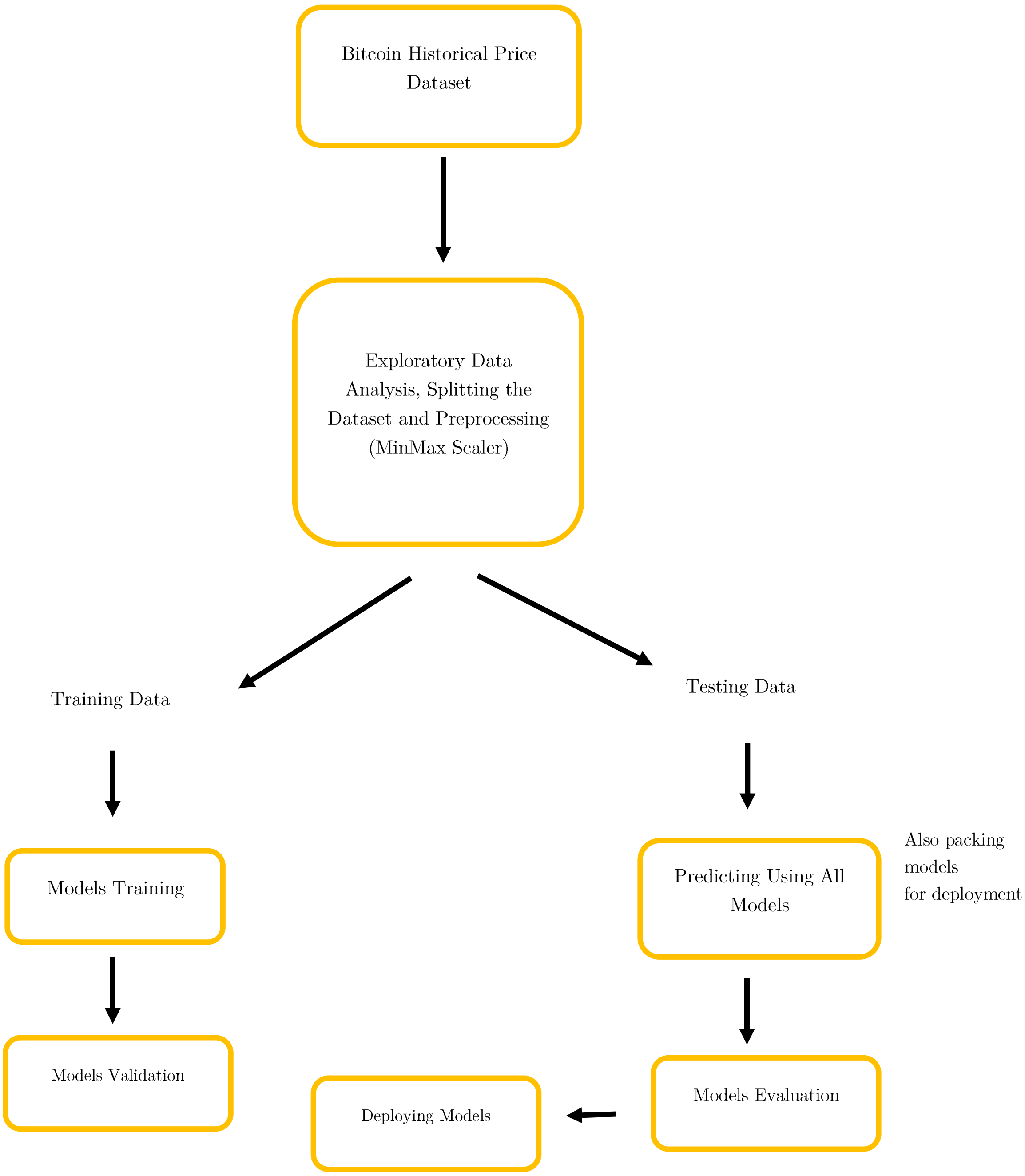
LSTM Neural Network

LSTM network is short term for "Long Short Term Memory Network". It is a variety of recurrent neural network that are capable of learning long-term dependencies, especially in sequence prediction problems. LSTM has feedback connections i.e. it is capable of processing the entire sequence of data, apart from single data point such as images. LSTMs are often referred to as fancy RNNs. Vanilla RNNs do not have a cell state. They only have hidden states and those hidden states serve as the memory for RNNs. Meanwhile, LSTM has both cell states and a hidden states. An LSTM recurrent unit tries to “remember” all the past knowledge that the network is seen so far and to forget irrelevant data. This is done by introducing different activation function layers called “gates”. Each LSTM recurrent unit also maintains a vector called the Internal Cell State which conceptually describes the information that was chosen to be retained by the previous LSTM recurrent unit. The LSTM structure is shown below



Pipeline

The pipeline followed is shown in the below

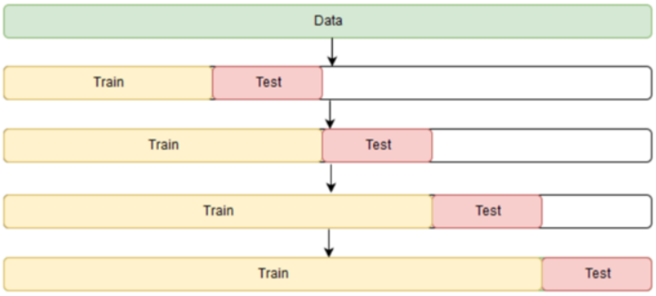


Time Series Validation

The models were cross validated for their performance using the time series split validation technique. The models were trained with different splits of the training data. The predictions were made with the **test split** of the **training data** and evaluated using the metrics **"Mean Squared Error"** and **"Mean Absolute Error"**. The average of **MSE** and **MAE** obtained from all the splits and by training all the models reported in form of a table.

	Mean Squared Error	Mean Absolute Error
XGBRegressor	0.005645	0.059303
LGBMRegressor	0.004962	0.054485
RandomForestRegressor	0.004130	0.049343
DecisionTreeRegressor	0.008161	0.072058
LSTM Network	0.003334	0.045826

Validation Results*

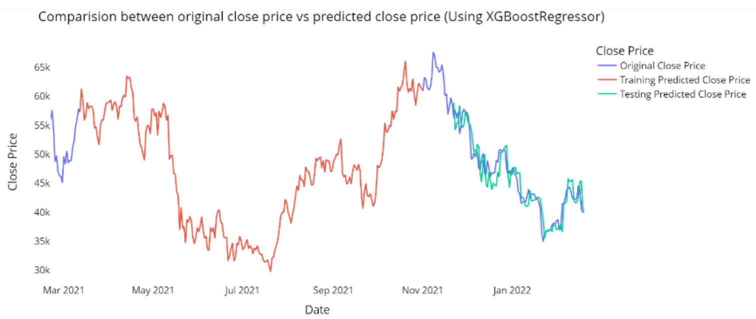


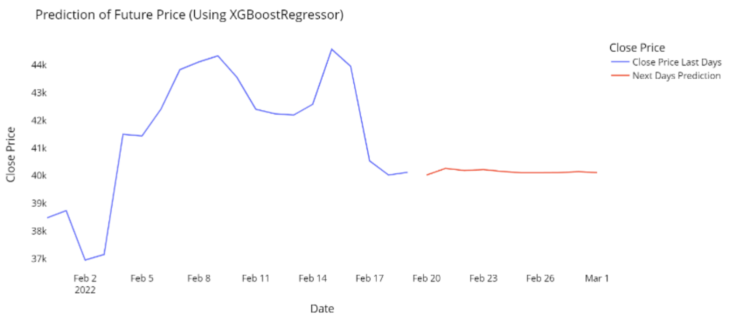
Time Series Split Strategy

*Note the results of LSTM Network are subjected to changed because the LSTM Neural network was not seeded to produce the same results. However the relative order of performance will remain the same. Also Note that the Error values are calculated on scaled data

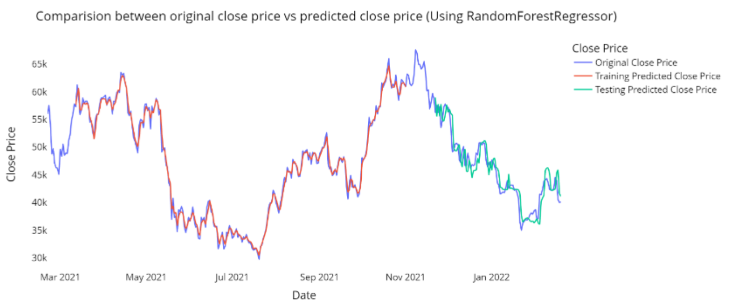
Final Model Implementation

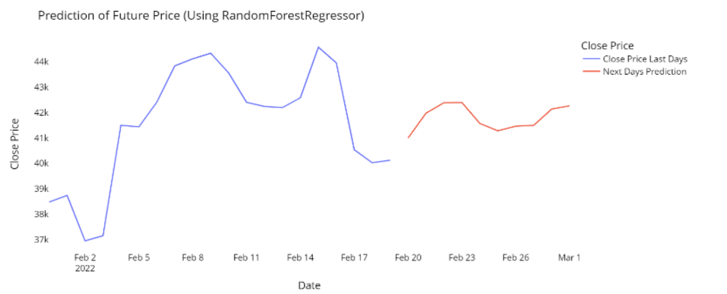
XGBoost Regression



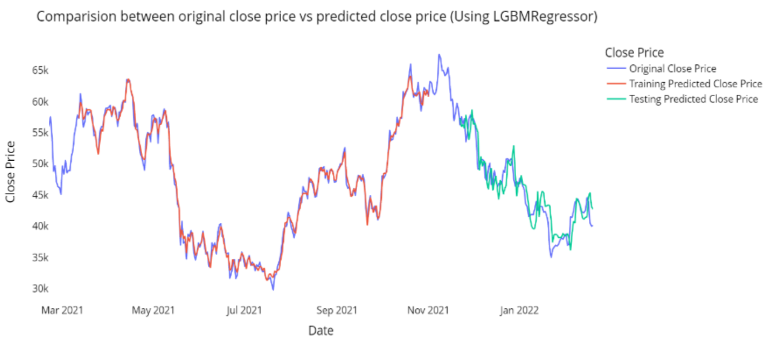


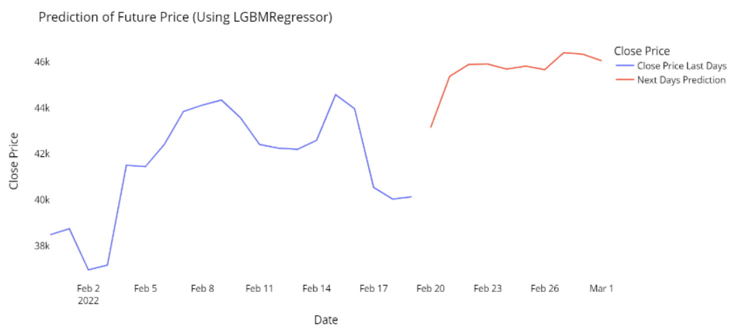
Random Forest Regression



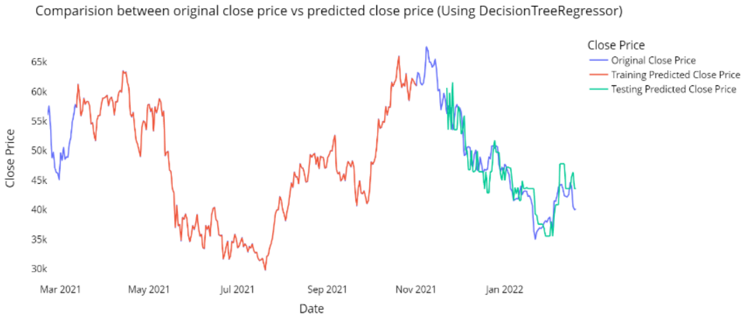


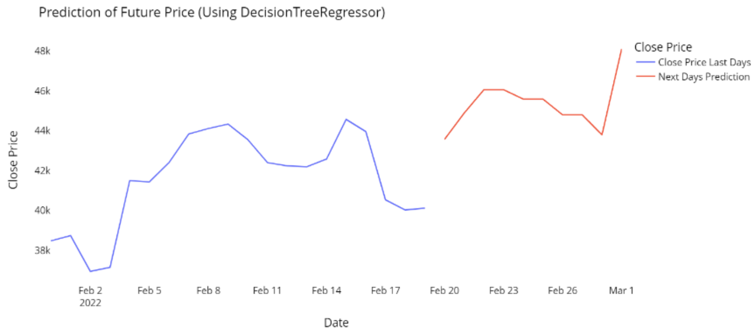
LightGBM Regression



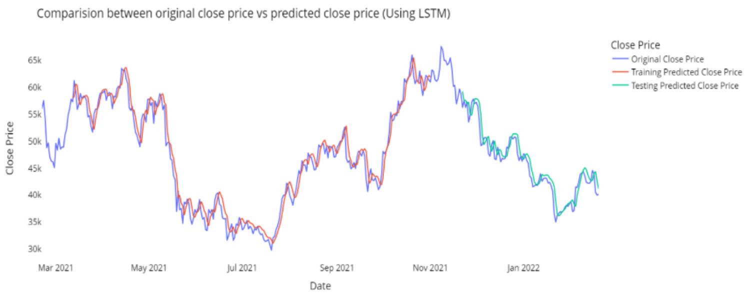


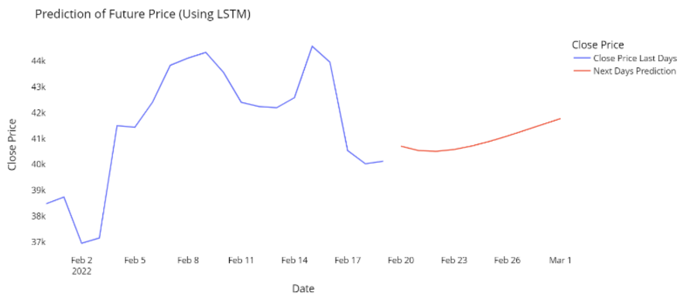
Decision Tree Regression





LSTM Network





Final Mean Squared Error and Mean Absolute Error*

*The scores are based on the scaled data and not on the original data

	Root Mean Squared Error	Mean Absolute Error
XGBoostRegressor	0.057064	0.043213
RandomForestRegressor	0.054021	0.040776
LGBMRegressor	0.062074	0.047749
DecisionForestRegressor	0.075263	0.060184
LSTM	0.049165	0.037610

Link to Web APP

https://share.streamlit.io/kartikchhipa01/bitcoinwebapp/main/main.py

References

1. https://machinelearningmastery.com/gentle-introduction-long-short-term-memory-networks-experts/
2. https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/
3. https://www.simplilearn.com/tutorials/machine-learning-tutorial/stock-price-prediction-using-machine-learning
4. https://www.geeksforgeeks.org/long-short-term-memory-networks-explanation/