OPTIMAL BUY TIME REGRESSION

ASHAY PATHAK, RITESH PATEL, CHATANA MANDAVA

1 Introduction

Cryptocurrency is a digital asset designed to work as a medium of exchange that uses strong cryptography to secure financial transactions. Cryptocurrencies are turning heads in the financial services space because of its marketplace dynamics are like equities. Furthermore, the biggest reason is financial returns are mind boggling. In a matter of weeks in November 2017, bitcoin surged from a fringe investment to a global sensation. In mid-November, the prices were around $3,000 for a single bitcoin; in December 6, 2017, it surpassed $19,000. At the time of publication, the value was hovering around $15,000. Cryptocurrency is changing the future of finance.

2 Motivation

These days many people seek cryptocurrency because of its volatility. Whether they do it consciously or not, they wish to make large money quickly. Further, we find a lot of newbies feeling bad that they lost money, or they could not make money as value of currency has not risen much especially with their expectations being high after the 2017 spike. Seeking such profits is not bad but before investing we feel there is a need to prioritize the needs that could be fulfilled with such an investment especially if investors are looking to make money soon rather than keeping the currency as a long-term investment.

This very motivation and idea behind the investment will speak volumes on the reaction of the investors once the market is down. Strangely, many people do not sell when the market price is high and even when their purpose of investment will get fulfilled. They wait for the price to soar even higher. And if the value starts trickling down, they still hold and wait for the price to move beyond the point where it had last peaked high. But again, when the market is in a steep downfall, instead of buying the asset many people will panic and sell it and therefore end up losing money.

So, the motivation behind this project is to show the direction to investor by giving them the best time to buy when the prices are going to be down in future so that they can buy at low price and sell when the market soars, which is the ultimate motive behind any short-term investment. The project would have potential application in portfolio management in which its very important to decide what coin to buy/sell and when. This would also help investors to hedge against the market risk.

3 Related Work

In this paper Laura Alessandretti, 1 Abeer El Bahrawy tried to find the price of cryptocurrencies using machine learning algorithms. Here, they tested the performance of three models in predicting daily cryptocurrency price for 1,681 currencies. Two of the models are based on gradient boosting decision trees and one is based on long short-term memory (LSTM) recurrent neural networks. In all the cases, they have built investment portfolios based on the predictions and we compare their performance in terms of return on investment [1]. And in this paper the author Younghoon kwaak explained how to use monte Carlo simulations to predict the risk in project management [2]. In Predicting Cryptocurrency Prices with Deep Learning, Sheehan uses LSTM to model price movement of Bitcoin and Ethereum [3]. In Analyzing Cryptocurrency Markets Using Python, Triest does statistical analysis of a handful of top cryptocurrencies [4].

4 Problem Statement

As we started with this project, we had a few questions in our minds. We also brainstormed ideas on what can be done differently to make our predictions accurate. Following are the questions we answer in this project.

- Can we predict the best time to buy cryptocurrency in coming n days?
- Can Monte Carlo Simulations be used to predict cryptocurrency closing prices in the future?
- What are the important factors/features that contribute towards the close price fluctuations?
- Is there is correlation between day to day bitcoin news and its close price movements?
- Can Exponential Moving Average (EMA) and Simple Moving Average (SMA) be used as indicators/additional features to determine close price?

The main challenge in making future predictions is that there are numerous factors/features that play a big role in determining the close price of cryptocurrency. Features like Open, High, Low, Close, Volume are easy and pretty straightforward to figure out their correlation with target variable “Close Price” and how to utilize them in building the model. However, there are features like news sentiment score, indicator-features (EMA, SMA, RSI) and unquantifiable features like investors emotions which makes this task difficult. If a large group of investors blindly follow some influential person’s investment decisions there is no feature we can add to account for the same, which make the problem becomes even harder.

5 Data Science Pipeline

5.1 Data Collection

All datasets were obtained from cryptocompare, yahoo finance and coinmarketcap.

- **CoinMarketCap** is a platform created to track the capitalization of different cryptocurrencies, the amount of trades that use them and the current price converted into fiat currencies. The information updates every five minutes. From coinmarketcap we scrapped the names of top 100 cryptocurrencies.
- **Yahoo! Finance** is a media property that is part of Yahoo!'s network. It provides financial news, data and commentary including stock quotes, press releases, financial reports, and cryptocurrency-data. We implemented the data importer module what would import data for each coin-USD market through yahoo finance API for the daily close prices.
- **CryptoCompare** is a global cryptocurrency market data provider, giving institutional and retail investors access to real-time, high-quality, reliable market and pricing data on 5,300+ coins and 240,000+ currency pairs. We used cryptocompare API to get the daily news for cryptocurrencies and hourly OHLC data.

5.2 Data Preprocessing

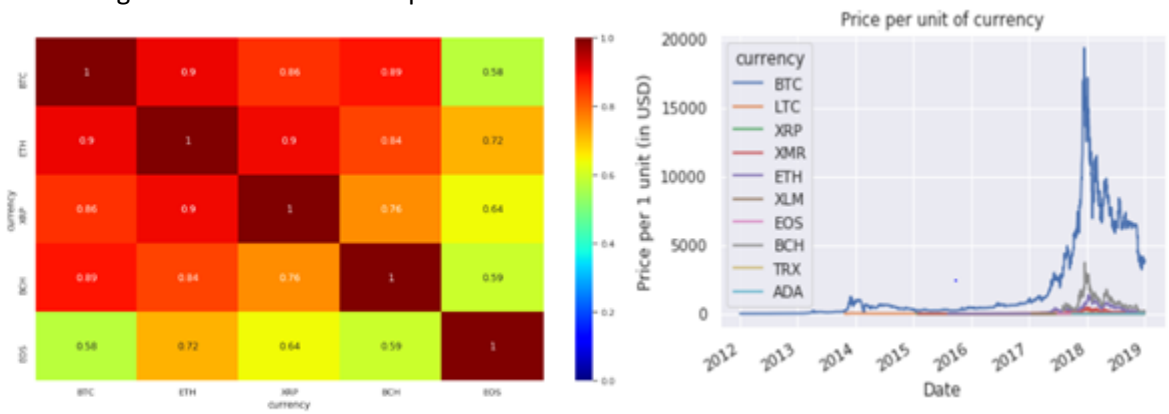
When we collected the data from the above-mentioned sources. We changed the time stamp from unix timestamp format to standard timestamp format. We observed that data was not enough for a deep learning models. Hence, we shifted from daily data to hourly data which significantly improved the model accuracy which will be discussed below. The original data is OHLV. For CNN, we normalized the data in windows and for every window we divided the data by the last closing price for that window. For CNN we decided the window size to be 24**.** Hence, we divided the data into the buckets of 24. For Normalization we divided the data by the closing price on the 24th hour of each bucket. The reason behind doing this

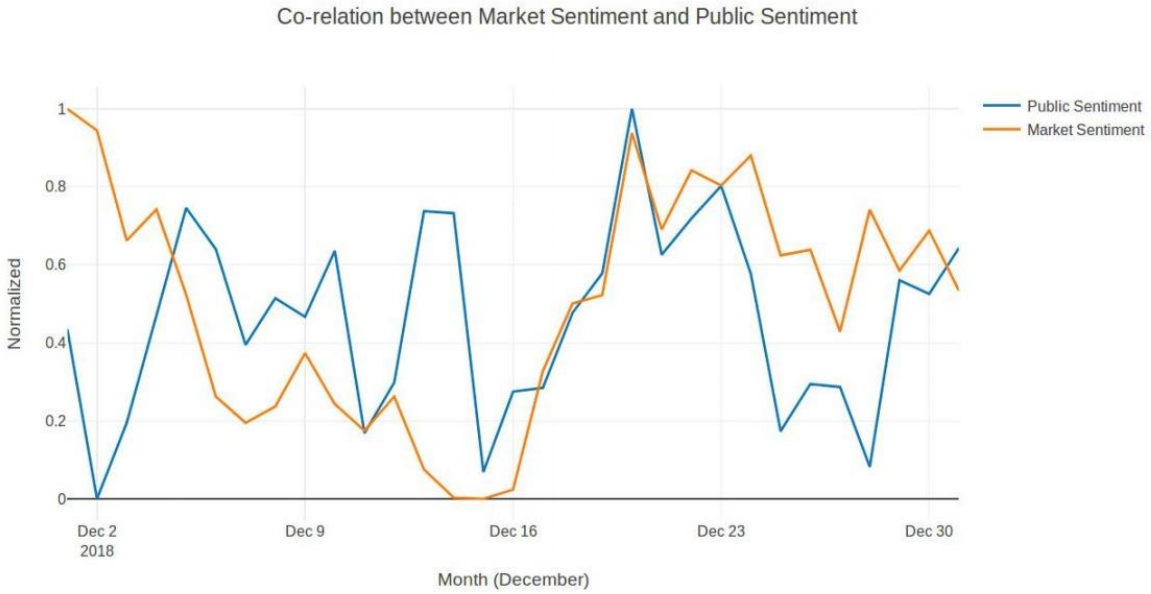
was it will provide the curve where the data is heading towards every 24 Hours. We removed Volume from the data after trying because it degraded the model accuracy. For LSTM we used minmax scaler for normalizing and a window size of 40 and batch size of 20.

5.3 Data Analysis

Exploratory Analysis with Jupyter Notebooks

Jupyter is a free, open-source, interactive web tool known as a computational notebook, which researchers can use to combine software code, computational output, explanatory text and multimedia resources in a single document. Jupyter notebooks provide clean, interactive environments for reproducible research. This process proved to be extremely rewarding by making it easy to recall and reuse previous work. It also has the beneficial side effect of producing a portfolio of the work that we have done. We have performed exploratory data analysis using jupyter notebook and we have got some useful insights on the data we scraped.





Firstly, we tried to find the trend of the 10 cryptocurrencies and we found that Bitcoin remains in the top position followed by Ethereum when compared to other cryptocurrencies. We have also found the correlation between top 5 cryptocurrencies using Pearson as well as spearman correlation and found that Bitcoin and Ethereum are highly correlated and spearman correlation scores are high when compared to Pearson. We also tried to analyze the trend in cryptocurrencies within the change of years and the increase in price of bitcoin in 2018 is mind boggling. And we also found the factors that are affecting the increase or decrease in price of a particular cryptocurrency.

Moreover, we also tried to find if there is any correlation between the news and closing price. This can be seen in the 3rd figure above. For this, we scraped the news for all the days and tried to generate a score for it in the range of 0-1. On the other side, we normalized the closing price also so that both the news score and price score have the same range. Then we plotted the data for 30 days. Above shown graph is for the month of December. It can be seen that closing price and news sentiment are correlated and both shows the same curve.

6 Methodology

Forecasting the price of next n days with Deep Learning

Inspired by few other works that are done on predicting the price of cryptocurrency we also decided to move forward in predicting the price of not just the consecutive day but the price of coming n days. We have used three deep learning methods to do so. Those are LSTM, CNN and Monte Carlo simulations.

6.1 LSTM

The LSTM Network has its origin in Recurrent Neural Network. This kind of network is used to recognize patterns when past results have influence on the present result. An example of RNN usage is the time-series functions, in which the data order is extremely important. A small change in the RNN cell architecture can solve the memory loss issue for LSTM. LSTMs are considered to be the go-to deep learning models for solving any time series problems.

Model Architecture

We want to predict the n = 10 days ahead (foward_days) having the m = 40 past observed days (look_back) as an input. So, if we have an input of m past days, the network output will be the prediction for the n next days. We will split the data in Train and Test. The test will be composed of k = 20 periods (num_periods), in which every period is a series of n days prediction. Data is divided into batch size of 20. Data is feed to the first LSTM cell with 50 neurons. Second Layer is a dropout layer with value 0.25. Third layer is again LSTM cell with 30 neurons and is followed by another dropout layer with value 0.25. Lastly a Dense Layer which outputs 10 close prices is added to the model. The Loss function used for training the model is Mean squared Error and optimizer was ‘adam’.

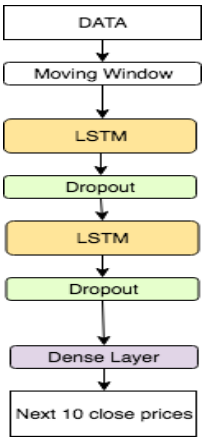


Fig. LSTM model architecture

6.2 CNN (Multi Channel and Multi Step CNN)

A CNN works well for identifying simple patterns within the data which will then be used to form more complex patterns within higher layers. A 1D CNN is very effective when we expect to derive interesting features from shorter (fixed-length) segments of the overall data set and where the location of the feature within the segment is not of high relevance.

Multi-Channel CNN

CNN is mainly considered to be the state-of-the-art for image related deep learning implementations. The reason behind CNN giving good outputs is its ability to work with multivariate data and dealing each variable separately. These variables which are treated individually are called channels.
We have used OHLC data for this and in our case, we have used top 5 cryptocurrencies from 30th Jan 2018 to 4th April 2019. Data Collection for this as discussed above, we used OHLC data. We used 2 sources to collect the data Cryptocompare API and Yahoo Finance.

Data Augmentation

When we collected the data from the above-mentioned sources, we observed that data was not enough for a deep learning model especially for CNN. Hence, we shifted from daily data to hourly data which significantly improved the model accuracy which will be discussed below. We collected the hourly data for 5 cryptocurrencies from 30th Jan 2018-4th April 2019. The total number of rows in our data for 1 currency is 10005.

Data cleaning and Normalization

This is one of the most important part where we focused on. For CNN we decided the window size to be 24. This was decided after checking the outputs with different window size. The main reason we decided CNN was to use all the multivariate data we had such as Open, High, Low, Close, Volume. Hence, we divided the data into the buckets of 24. For Normalization we divided the data by the closing price on the 24th hour of each bucket. The reason behind doing this was it will provide the curve where the data is heading towards every 24 Hours. We removed Volume from the data after trying since, it was decreasing the model accuracy. The output in our case is the closing price for next consecutive 12 hours.

Model Structure

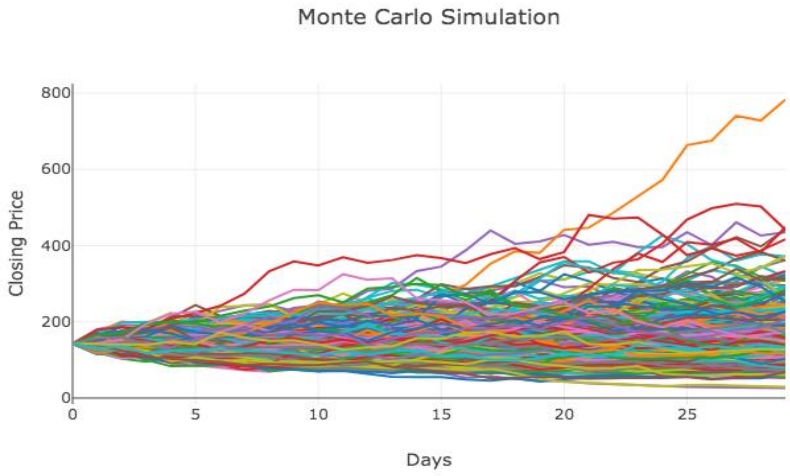
The model used is Multi Channel Multi Step Convolutional Neural Network. The input to the model is the normalized data for past 24 days. Hence the shape of input data is (1,15,4). The output of the model is closing price for next 12 consecutive days. Hence the output shape is (1,12,1). In-between this, the data is passed through the convolutional model. We have used 1D convolution with filter size of 3. We have used Max-Pooling layer after convolution. Hyper Parameter Tuning is very important when working with deep learning models. We decided optimizer as Adam. The loss function used for CNN is MSE.

6.3 Monte Carlo

Monte Carlo simulations is a statistical technique used to model probabilistic (or “stochastic”) systems and establish the odds for a variety of outcomes. Here we have used Monte Carlo simulations to understand the market trends over a period of time. The world, however, is full of more complicated systems than a shot-put toss. In these cases, the complex interaction of many variables or the inherently probabilistic nature of certain phenomena rules out a definitive prediction. So, a Monte Carlo simulation uses essentially random inputs (within realistic limits) to model the system and produce probable outcomes. Here we have used 500 Monte Carlo simulations of close prices for coming 30 days. The close prices were repeatedly picked from normal distribution at random.

Implementation:

We implemented the Monte Carlo method in python. We used the concept of Brownian Motion to get the values. It depends upon the concept of Drift.



Above figure shows the closing price simulation for 30 days. We tried to obtain 500 simulations over the period. So, it can be observed that mostly the simulations are overlapping which shows that market for the coming time is not that volatile and the impact of risk is also less. After 20th day, it is seen that one simulation shows the soaring trend.

7 Model Evaluation

7.1 LSTM

We started with a simple model with only one input feature i.e. close prices and after tuning hyperparameters and adding dropout layers best results were achieved by following model summary

Layer (type)	Output Shape	Param #
lstm_1 (LSTM)	(None, 40, 50)	10400
dropout_1 (Dropout)	(None, 40, 50)	0
lstm_2 (LSTM)	(None, 30)	9720
dropout_2 (Dropout)	(None, 30)	0
dense_1 (Dense)	(None, 10)	310

Total params: 20,430
Trainable params: 20,430
Non-trainable params: 0

None

Fig. LSTM_CP Model Summary

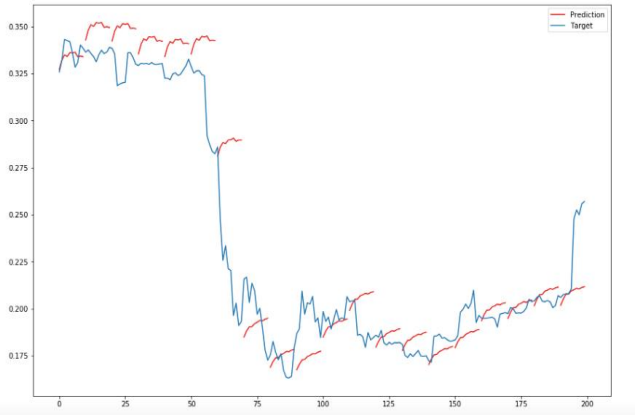


Fig. LSTM_CP_Predictions

We obtained MAE(prices): 232.697 and MAE(days): 3.95. To improve the Mean Absolute Error further we added Exponential Moving Average as a derived feature based on Close Prices remaining model specifications remains the same. Later we obtained MAE(prices):380.32 and MAE(days): 3.5. The MAE value further degraded after adding the EMA. So, finally to increase the training dataset size we got hourly bitcoin close prices and trained the model which significantly improved our model accuracy.

Layer (type)	Output Shape	Param #
lstm_5 (LSTM)	(None, 40, 50)	10400
lstm_6 (LSTM)	(None, 30)	9720
dense_3 (Dense)	(None, 10)	310

Total params: 20,430
Trainable params: 20,430
Non-trainable params: 0

None

Fig. LSTM_hourly Model Summary

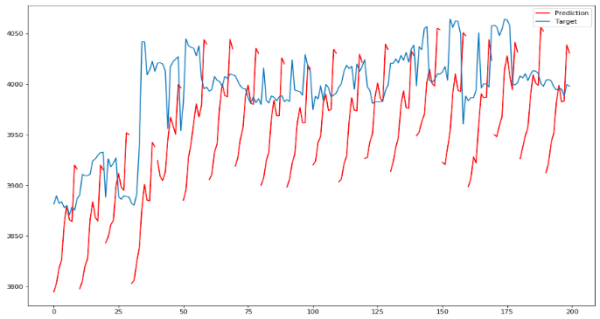
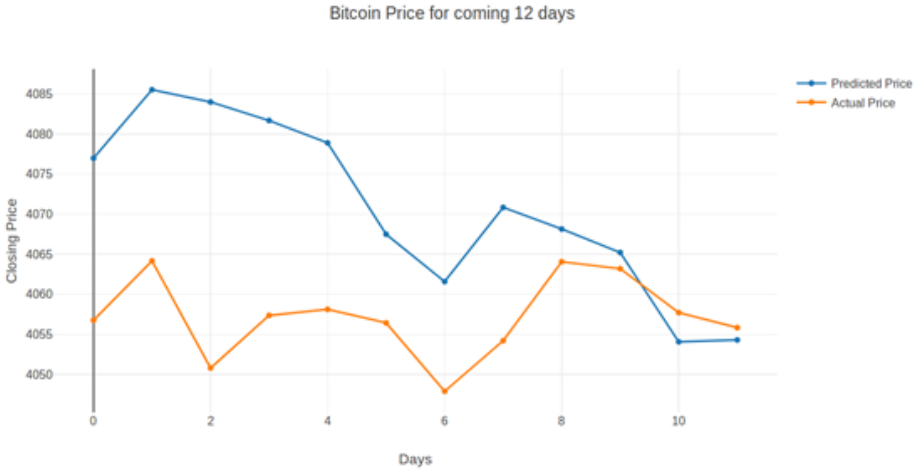


Fig. LSTM_hourly Predictions

We obtained MAE(prices): 52.79510131835938 MAE(days): 3.35. Data augmentation certainly helped us with model accuracy.

7.2 CNN



The output of the model is the closing price for next consecutive 12 hours. Attention must be paid that the obtained prices from model output are normalized. hence it has to be de-normalized to obtain the original closing price. Here, we can see that the predicted model has suggested the 10th day as the best day to buy the asset, which is 2nd best day in the all given days.

Result Comparison (LSTM vs CNN):

Model	Mean Absolute Error(days)	Mean Absolute Error(prices)
LSTM-CP	3.8	232.69741235351566
LSTM-CP-EMA	3.5	380.3250916503906
LSTM-CP-HOURLY	3.35	52.79510131835938
CNN	2.3	10.22332344244

8 Data Product

A product that facilitates an end goal through the use of data is a data product. We have 2 products that we tried to obtain from the data.

8.1 Optimal Time in next N days

Figure 8.1 shows the optimal time to buy Ethereum in next 12 days. Here we can observe that 6th day is the best day as the price is least in the period. Hence, our model suggests the investor to buy the Ethereum on that day since the main motivation behind any short-term investment is to gain maximum profit. Maximum profit can be gained by buying at less cost and selling at high cost. Hence our product fulfills this requirement and tells the optimum time to invest.

8.2 Market trend and impact of risk in next n days.

Figure 8.2 is our 2nd data product. It is based on probabilistic model that generates simulations for a period of time. As shown in the above figure, there are 500 simulations for the next 30 days. So, it can be observed that mostly the simulations are overlapping which shows that market for the coming time is not that volatile and the impact of risk is also less. After 20th day, it is seen that one simulation shows the soaring trend. Hence this product gives the idea of whether the day to buy currency suggested in the above data product is reliable or not. If there is high overlapping and the scatter is less on that day, then the investor can make his mind to buy since the impact of risk is less.

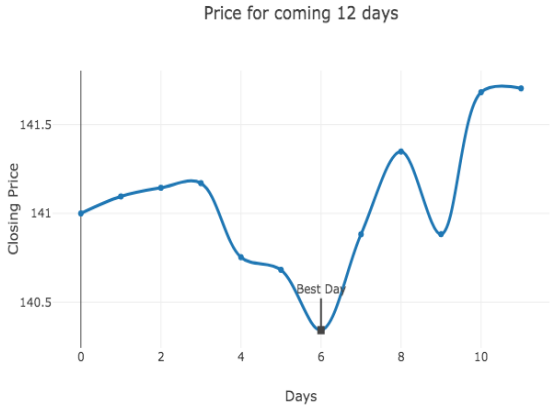


Fig 8.1

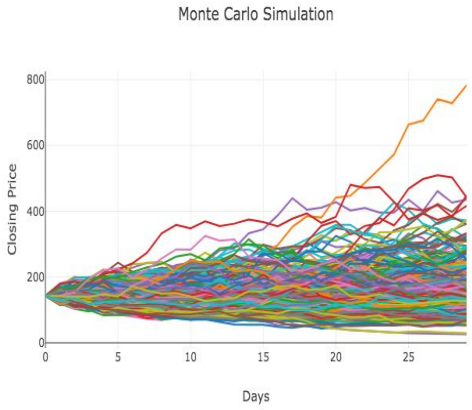
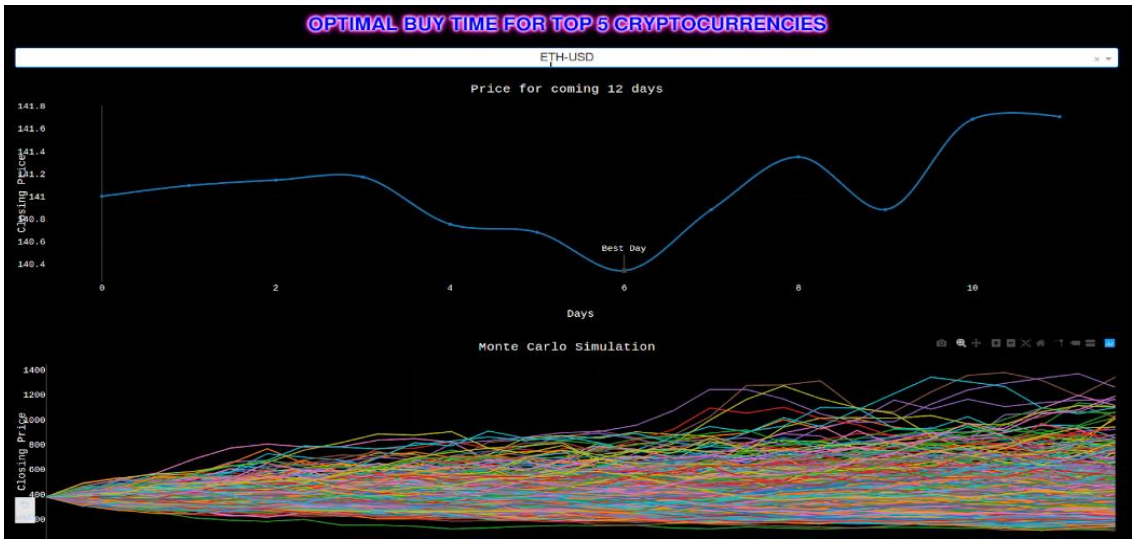


Fig 8.2

Dashboard and Technologies Used

We have created our dashboard using plotly**.** Dash apps are composed of two parts. The first part is the layout of the app and it describes what the application looks like. The second part describes the interactivity of the application. Dash provides Python classes for all of the visual components of the application. It is customizable and own CSS, HTML, JavaScript and React.js codes can also be added.



So, the above figure integrates both the data products explained above in the dashboard. This product gives the drop-down menu for the user to select the cryptocurrency. For this project we have worked on top 5 currencies. Once the user selects the currency, the graph gets generated below as shown in the figure. Both the graphs are the individual data products.

9 Lessons Learned

We have learned how to build the whole pipeline for a project and present it. We also developed our presentation skills and we started thinking like a data scientist. We learned how to define the problem and solve it, we also learned cleaning the data, analyzing the data, building the models, evaluating the models, and to create an interactive dashboard to display our findings. There were many critical decisions that had to be taken during the development and implementation of our project. When working with deep learning model data will be the main concern and, in our case, there was lack of data and we had to do data augmentation to overcome the problem. Not all features were highly correlated with our target variable and we used feature selection and feature engineering to overcome it. And we also found that our model was overfitting and we had to add a drop out layer to overcome that. By poster presentation and making the YouTube video we have learned how to present our entire project in a given amount of time.

10 Summary

We have found that there is a relation between news sentiment and public sentiment while investing in cryptocurrencies. We have also performed window-based normalization and improved the accuracy of CNN model. Adding EMA, SMA as feature to the dataset degraded the model's performance. And, Monte Carlo cannot be used to predict the close prices however we can use it to understand market trends over a period. LSTM performs better with large data. With additional data and tuning we see a potential application of this model in production. But as of now CNN outperforms LSTM and gives us promising results.