Data Science Capstone Project:

BTC Price Predictor

Using Regression Model to forecast next closing price.

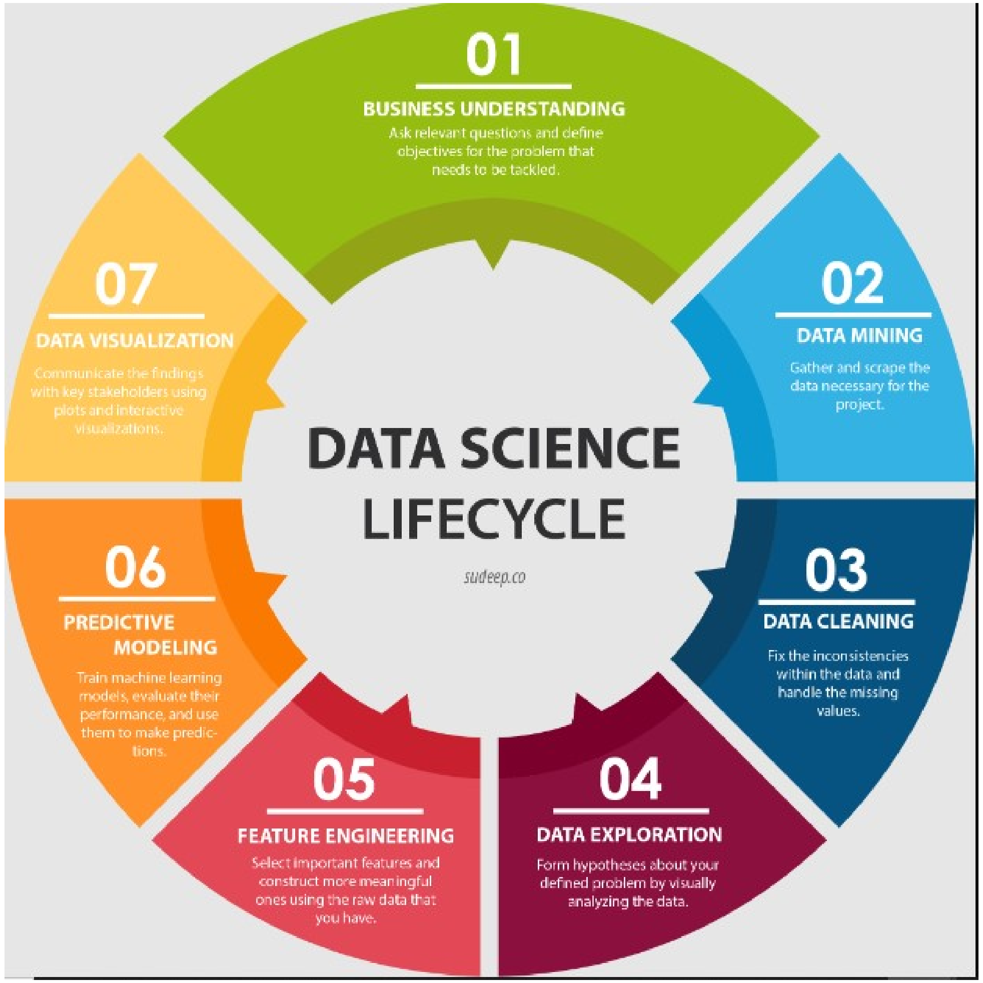
Chan Jiunn Hwa

Problem Statement

Since inception, Bitcoin has been both controversial and volatile and this creates good trading conditions, as well as encouraging research and use of new or modern analytical techniques for forecasting of prices.

This project aims not so much as to discover the ‘holy grail’ in a single exercise, but to build an initial model for further continual improvement; applying Machine Learning models to allow forecast of future closing price.

Overall Approach



DATA SCIENCE LIFECYCLE

sudeep.co

01
BUSINESS UNDERSTANDING
Ask relevant questions and define objectives for the problem that needs to be tackled.

02
DATA MINING
Gather and scrape the data necessary for the project.

03
DATA CLEANING
Fix the inconsistencies within the data and handle the missing values.

04
DATA EXPLORATION
Form hypotheses about your defined problem by visually analyzing the data.

05
FEATURE ENGINEERING
Select important features and construct more meaningful ones using the raw data that you have.

06
PREDICTIVE MODELING
Train machine learning models, evaluate their performance, and use them to make predictions.

07
DATA VISUALIZATION
Communicate the findings with key stakeholders using plots and interactive visualizations.

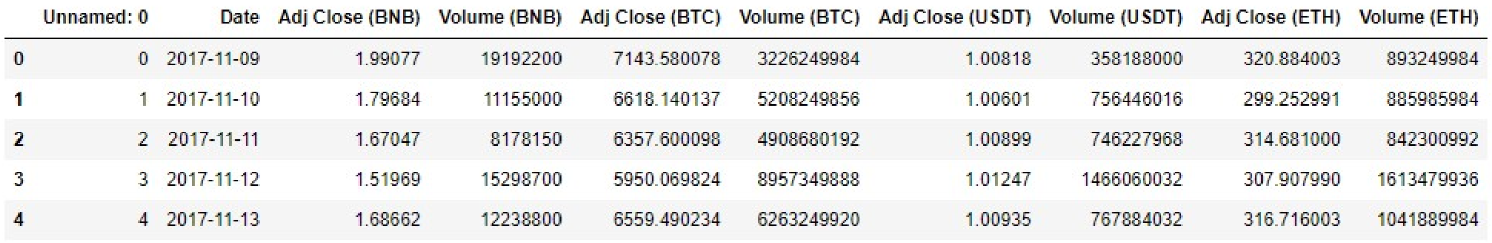
Adopt the use of standard DS Lifecycle steps.

01 - Problem Statement

The feature of high volatility in crypto markets creates good trading conditions. Therefore, we would like to see if Machine Learning can be applied to help predict future prices.

02 - Data Mining

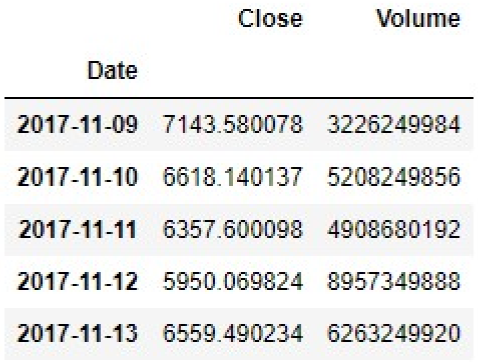
In this project, the data comes prepared from Kaggle sources. In real world, data gathering would involve data scraping and mining from the web, and so on.



	Unnamed: 0	Date	Adj Close (BNB)	Volume (BNB)	Adj Close (BTC)	Volume (BTC)	Adj Close (USDT)	Volume (USDT)	Adj Close (ETH)	Volume (ETH)
0	0	2017-11-09	1.99077	19192200	7143.580078	3226249984	1.00818	358188000	320.884003	893249984
1	1	2017-11-10	1.79684	11155000	6618.140137	5208249856	1.00601	756446016	299.252991	885985984
2	2	2017-11-11	1.67047	8178150	6357.600098	4908680192	1.00899	746227968	314.681000	842300992
3	3	2017-11-12	1.51969	15298700	5950.069824	8957349888	1.01247	1466060032	307.907990	1613479936
4	4	2017-11-13	1.68662	12238800	6559.490234	6263249920	1.00935	767884032	316.716003	1041889984

03 - Data Cleaning

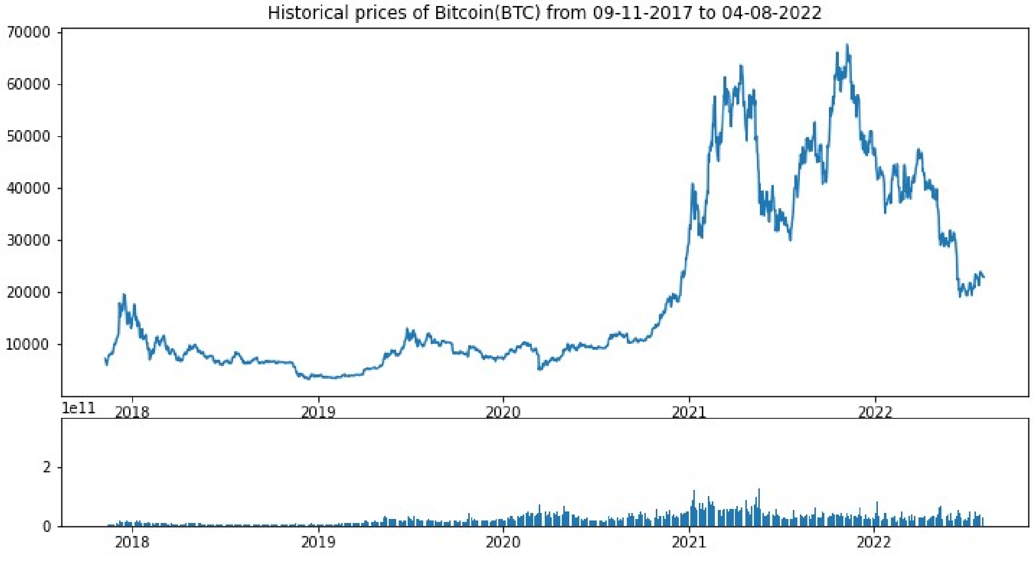
This step is to fix inconsistencies, extract and transform raw data to a format that our models can consume.



	Close	Volume
Date		
2017-11-09	7143.580078	3226249984
2017-11-10	6618.140137	5208249856
2017-11-11	6357.600098	4908680192
2017-11-12	5950.069824	8957349888
2017-11-13	6559.490234	6263249920

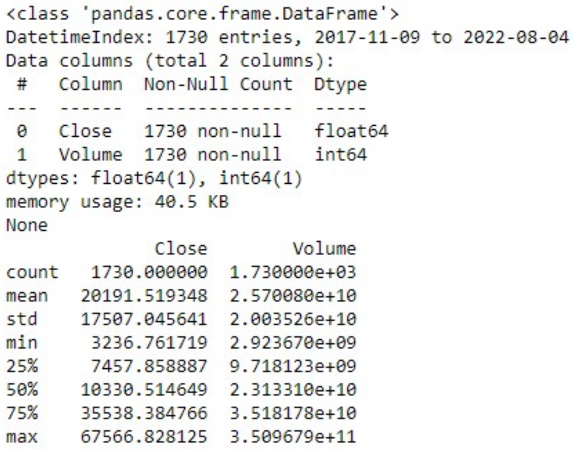
04 - Data Exploration

With the data cleaned, we can do some exploration.



Historical prices of Bitcoin(BTC) from 09-11-2017 to 04-08-2022

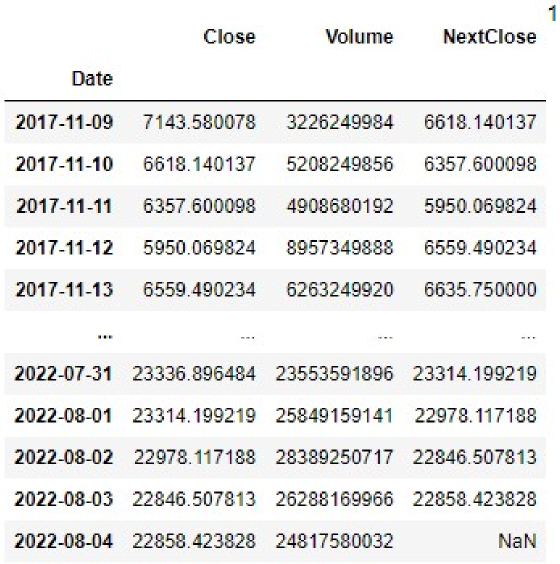
Trading Volume



```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 1730 entries, 2017-11-09 to 2022-08-04
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   Close   1730 non-null   float64
 1   Volume  1730 non-null   int64  
dtypes: float64(1), int64(1)
memory usage: 40.5 KB
None
              Close        Volume
count   1730.000000  1.730000e+03
mean   20191.519348  2.570080e+10
std    17507.045641  2.003526e+10
min     3236.761719  2.923670e+09
25%     7457.858887  9.718123e+09
50%    10330.514649  2.313310e+10
75%    35538.384766  3.518178e+10
max    67566.828125  3.509679e+11
```

05 - Feature Engineering

This step is to select important features, but in this case, we will try to model 'NextClose' which is the closing price tomorrow.

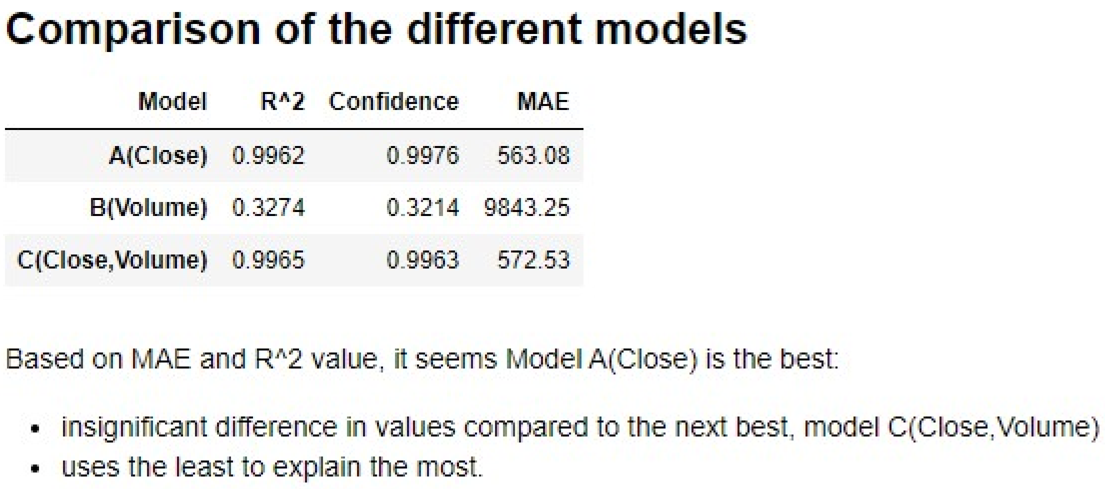


1

	Close	Volume	NextClose
Date			
2017-11-09	7143.580078	3226249984	6618.140137
2017-11-10	6618.140137	5208249856	6357.600098
2017-11-11	6357.600098	4908680192	5950.069824
2017-11-12	5950.069824	8957349888	6559.490234
2017-11-13	6559.490234	6263249920	6635.750000
...	...	...	...
2022-07-31	23336.896484	23553591896	23314.199219
2022-08-01	23314.199219	25849159141	22978.117188
2022-08-02	22978.117188	28389250717	22846.507813
2022-08-03	22846.507813	26288169966	22858.423828
2022-08-04	22858.423828	24817580032	NaN

06 - Predictive Modeling

Using LinearRegression as a first model, we use it to train and predict. Our model will try to predict next day closing(forecast_period = 1). We then use other prossible X(independent) variables to apply the same LinearRegression for comparative purpose.



Comparison of the different models

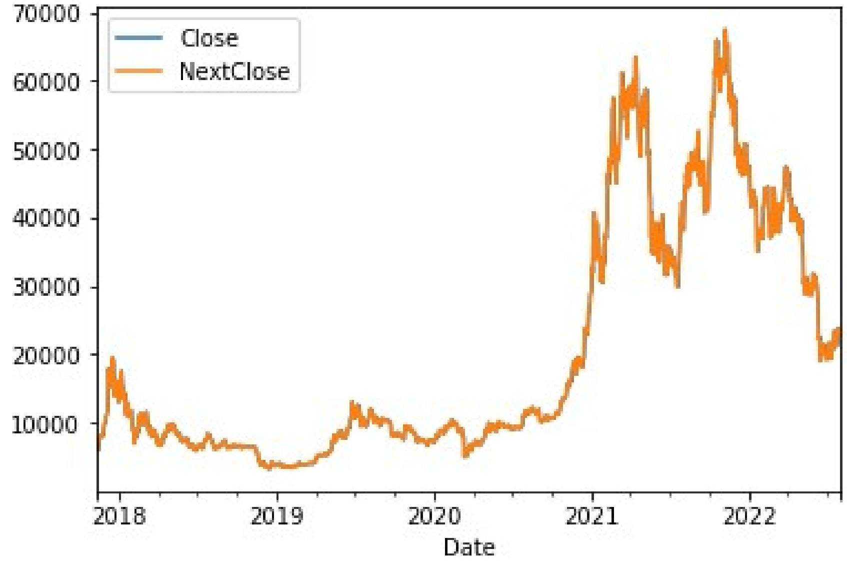
Model	R^2	Confidence	MAE
A(Close)	0.9962	0.9976	563.08
B(Volume)	0.3274	0.3214	9843.25
C(Close,Volume)	0.9965	0.9963	572.53

Based on MAE and R^2 value, it seems Model A(Close) is the best:

- insignificant difference in values compared to the next best, model C(Close,Volume)
- uses the least to explain the most.

07 - Data Visualization

Plotting Close and NextClose shows that the predicted value tracks the actual value very well.



Summary

The modeling looks promising but can be made more robust by:

- researching other independent variables, such as time of day, opening/closing hour, and so on.
- expanding comparison or integration with other models like LSTM,etc.
- expanding to other assets and see if we have similar results.
- research whether sudden deviations like price spikes are related to sudden news, and hence it could be useful to add other Machine Learning tools like Sentiment Analysis.

Also, the process can be further refined and standardized.

In real use, friction like slippages, execution latencies, liquidity have to be taken into consideration.