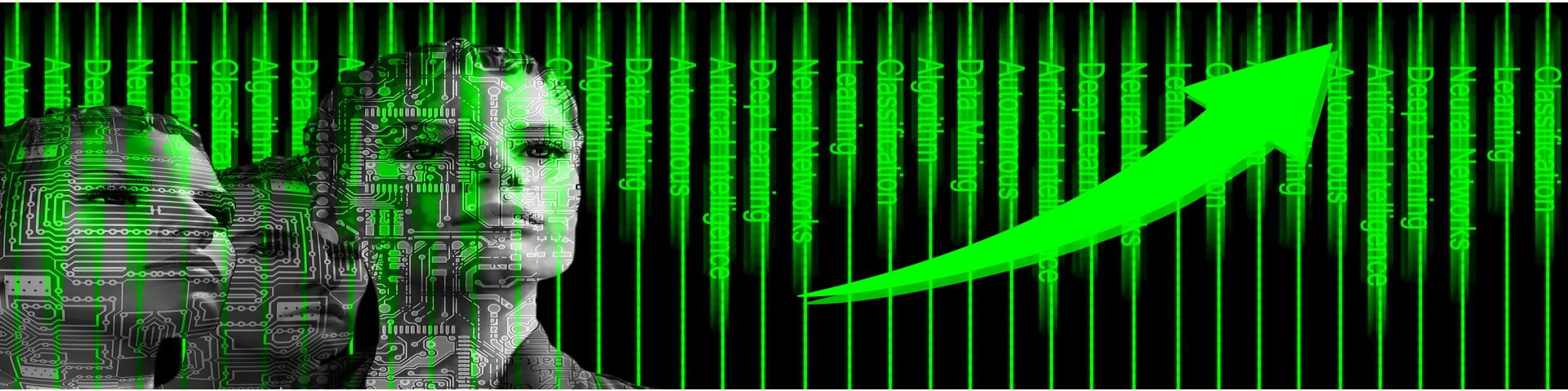
Crypto Prediction

Using Machine Learning to Predict Cryptocurrency Market Data

Reason for the Selected Topic



Due to dramatic changes in the world financial environment, cryptocurrencies have gained popularity as one of the alternative investment available to most. The volatility of cryptocurrency assets would pose a challenge to predict as there are multiple factors that influenced market data. Using Machine Learning, we hope to create a way to predict these crypto market data by studying the behavior of past data. We will assess and analyze historical data of six most popular cryptocurrencies and hope to compare the findings to real world market data.

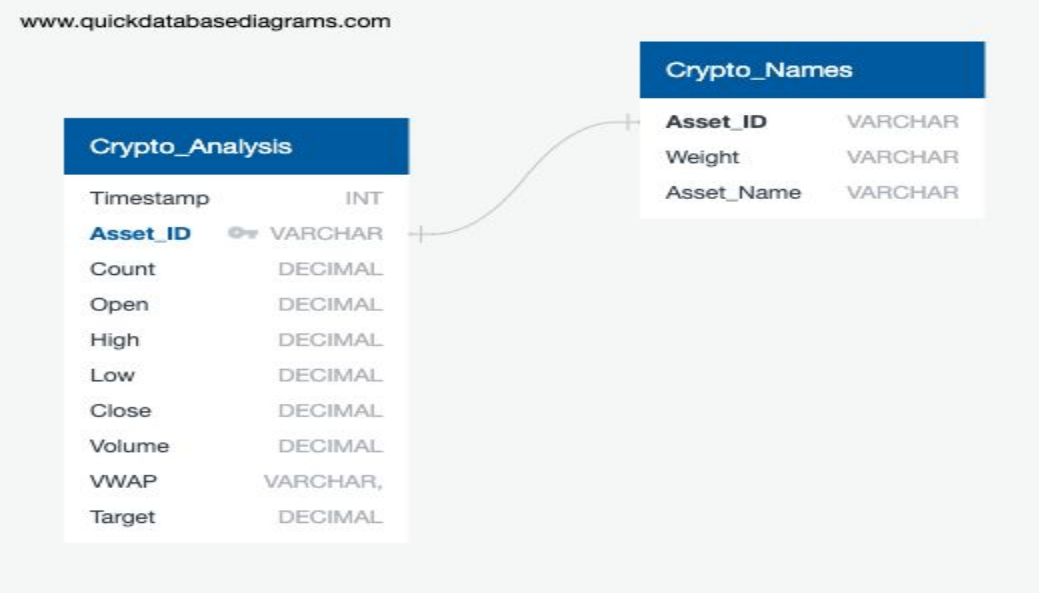
Questions We Hope to Answer with Data

- Which Machine Learning Model would best predict price changes on cryptocurrency from historical price?
- Which asset(s) is more stable out of the 6 cryptocurrencies chosen for prediction?
- Which features affect the close price the most?

Description of Data Source

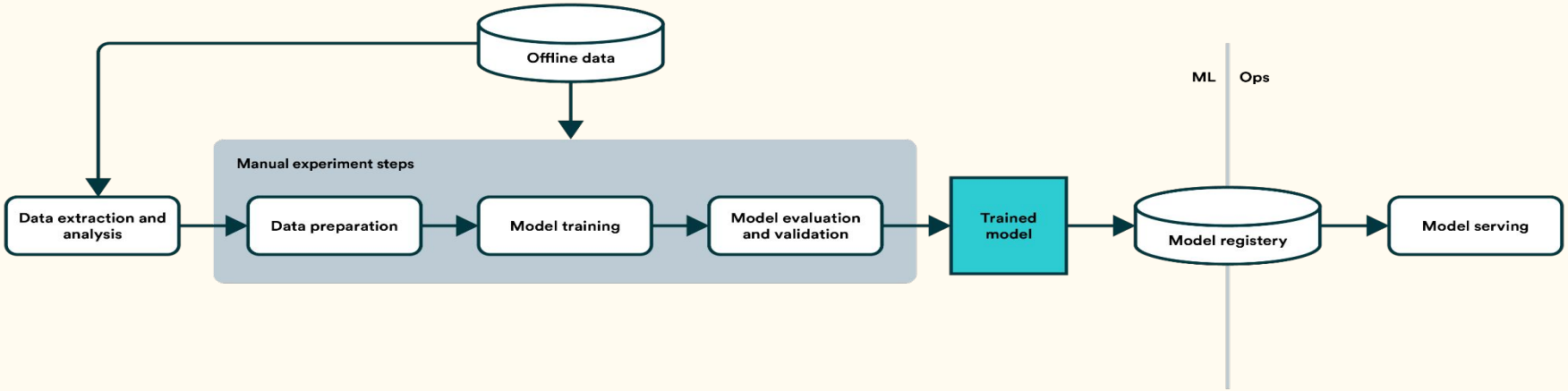
- Kaggle - the dataset contains historical trades on several crypto assets such as Bitcoin, Binance Coin, Etherium, Litecoin and Monero.
- G-Research - is a quantitative finance research firm is Europe. They utilized machine learning, big data and the most advanced technology to predict movements in the financial markets.

ERD


www.quickdatabasediagrams.com
Crypto_Names
Asset_ID VARCHAR
Weight VARCHAR
Asset_Name VARCHAR
Crypto_Analysis
Timestamp INT
Asset_ID VARCHAR
Count DECIMAL
Open DECIMAL
High DECIMAL
Low DECIMAL
Close DECIMAL
Volume DECIMAL
VWAP VARCHAR,
Target DECIMAL

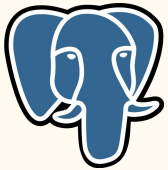
Process Map

Data Cleaning	Data Processing	Machine Learning	Optimization	Visualization
• Filling NaN data • Forward fill of missing values in time series data • Convert the minute-to-minute data to day-to-day data • Merge Datasets	• Manual split of training/testing set • Feature importances to determine best features	• Linear Regression • Random Forest Regression • XGBooster Regression • Artificial Neural network	• RandomGridSearchCV to determine best parameter for HyperTuning • Finding best hyperparameters for Neural Network using GridSearchCV	• Tableau Public • Matplotlib



Tools and Resources Used

- PostgreSQL
- Amazon Web Services (AWS)
- Jupyter Notebook
- Tableau Public
- Google Collab
- Google Slides
- Python language
- Python Dependencies: Scikit-learn, pandas, numpy, matplotlib, SciPy, tensorflow, StandardScaler, xgboost



Data Exploration

- timestamp - A timestamp for the minute covered by the row.
- Asset_ID - An ID code for the crypto asset.
- Count - The number of trades that took place this minute.
- Open - The USD price at the beginning of the minute.
- High - The highest USD price during the minute.
- Low - The lowest USD price during the minute.
- Close - The USD price at the end of the minute.
- Volume - The number of crypto asset units traded during the minute.
- VWAP - The volume weighted average price for the minute.
- Target - 15 minute residualized returns.

1. Check the missing values in each column and fill NAN values
2. Choose 6 most popular crypto assets for analysis
3. Forward filling gaps using the .reindex() method for each crypto

```
crypto_df = pd.read_csv('Resources/train.csv')
crypto_df.head()
```

	timestamp	Asset_ID	Count	Open	High	Low	Close	Volume	VWAP	Target
0	1514764860	2	40.0	2376.5800	2399.5000	2357.1400	2374.5900	19.233005	2373.116392	-0.004218
1	1514764860	0	5.0	8.5300	8.5300	8.5300	8.5300	78.380000	8.530000	-0.014399
2	1514764860	1	229.0	13835.1940	14013.8000	13666.1100	13850.1760	31.550062	13827.062093	-0.014643
3	1514764860	5	32.0	7.6596	7.6596	7.6567	7.6576	6626.713370	7.657713	-0.013922
4	1514764860	7	5.0	25.9200	25.9200	25.8740	25.8770	121.087310	25.891363	-0.008264

```
asset_df = pd.read_csv('Resources/asset_details.csv')
asset_df
```

	Asset_ID	Weight	Asset_Name
0	2	2.397895	Bitcoin Cash
1	0	4.304065	Binance Coin
2	1	6.779922	Bitcoin
3	5	1.386294	EOS.IO
4	7	2.079442	Ethereum Classic
5	6	5.894403	Ethereum
6	9	2.397895	Litecoin
7	11	1.609438	Monero

Data Exploration

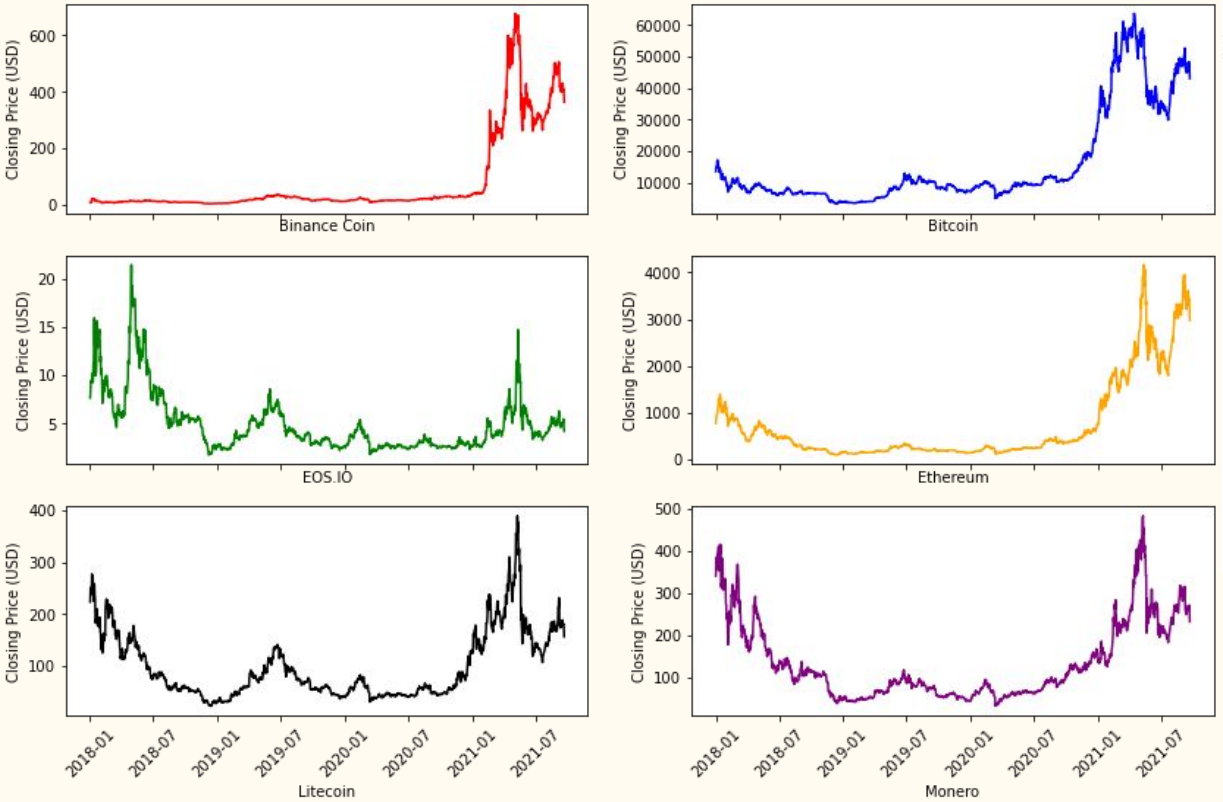
4. Convert minutes data to daily data for each cryptocurrency

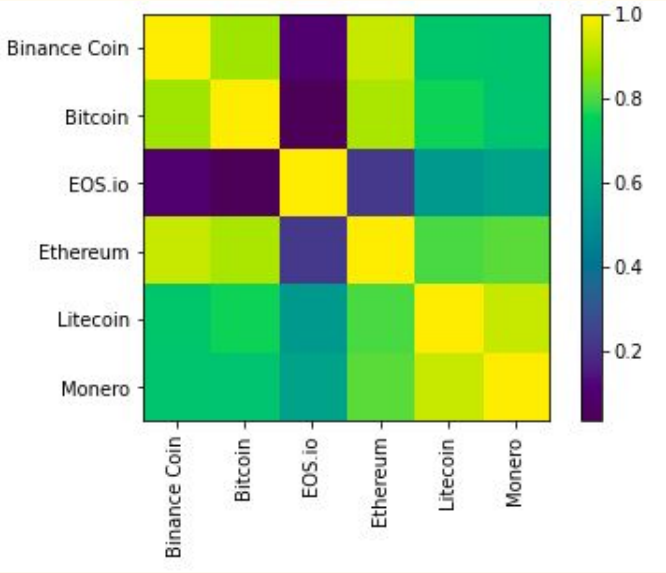
```
# convert minute data to daily data for Bitcoin
bit.reset_index(inplace=True)
bit["timestamp"]=bit["timestamp"].apply(lambda x: x-60) # run only once
bit["date"]=pd.to_datetime(bit["timestamp"], unit='s')
# create a new daily DataFrame for Bitcoin
bit_daily_df=pd.DataFrame(columns=["Asset_ID", "Open", "High", "Low", "Close", "Volume", "VWAP"])
bit_daily_df['High'] = bit.groupby(pd.Grouper(freq='D', key='date')).max()['High']
bit_daily_df['Low'] = bit.groupby(pd.Grouper(freq='D', key='date')).min()['Low']
bit_daily_df['Open'] = bit.Open[0:-1:24*60].values
bit_daily_df['Close'] = bit.Close[24*60-1:len(bit):24*60].values
bit_daily_df['Asset_ID'] = bit.Asset_ID[0:-1:24*60].values
bit_daily_df['Volume'] = bit.groupby(pd.Grouper(freq='D', key='date')).sum()['Volume']
bit_daily_df['VWAP'] = bit.groupby(pd.Grouper(freq='D', key='date')).mean()['VWAP']
```

date	Asset_ID	Open	High	Low	Close	Volume	VWAP
2018-01-01	1	13835.194	14442.9	12750.00	13468.698	57889.811032	13439.704032
2018-01-02	1	13459.606	15500.0	12798.00	14743.424	105541.245764	13885.548774
2018-01-03	1	14739.148	15599.7	14150.00	15107.792	83829.059632	14968.234757
2018-01-04	1	15096.770	15500.1	13918.04	15158.492	101148.935997	14820.300206
2018-01-05	1	15158.346	17200.0	14600.00	16935.556	108673.177856	15828.101106

Overview of Data

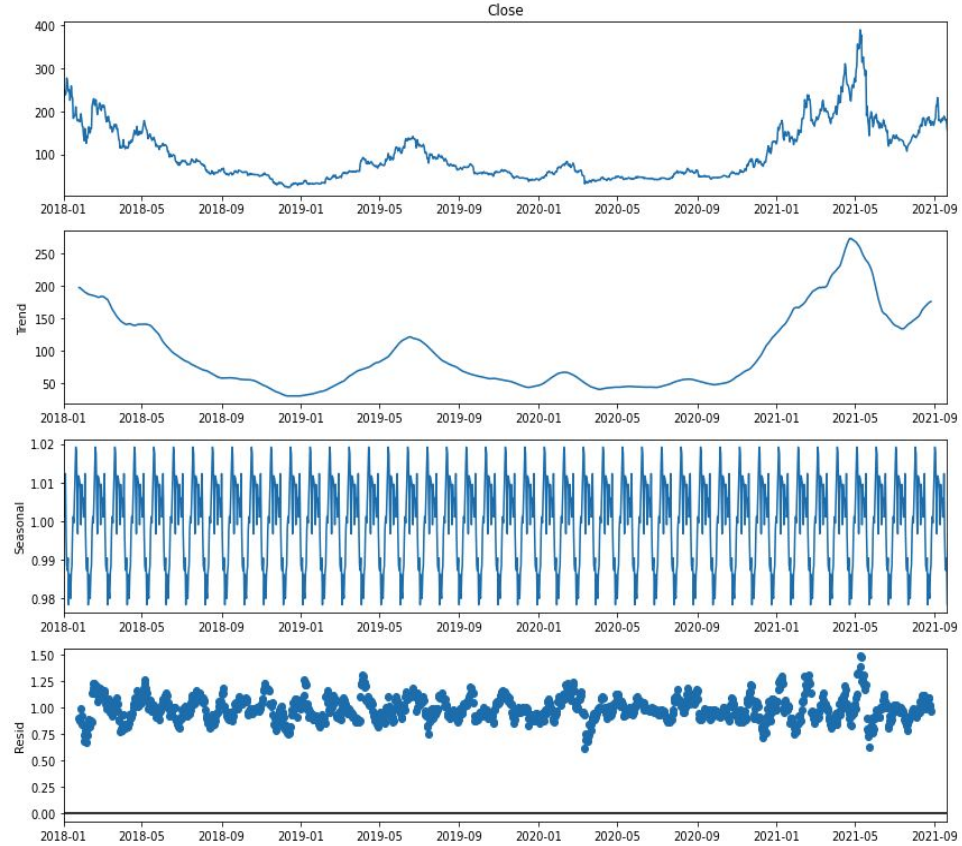
Closing price in USD of 6 chosen crypto assets from 2018 to 2021



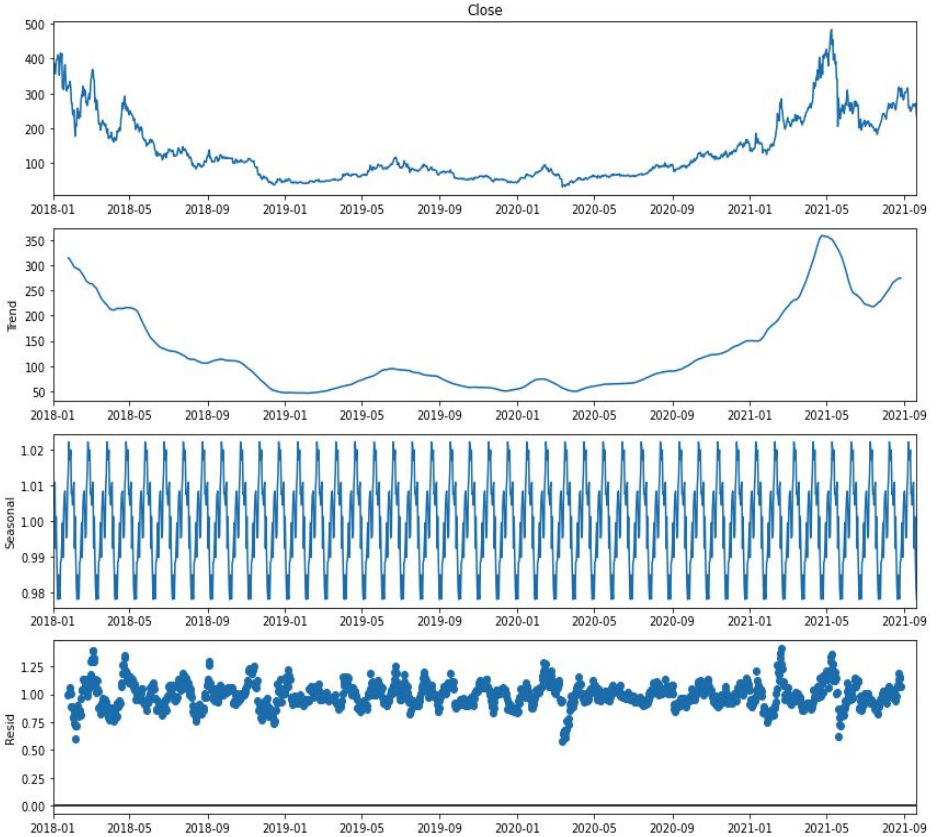


the correlation between each asset

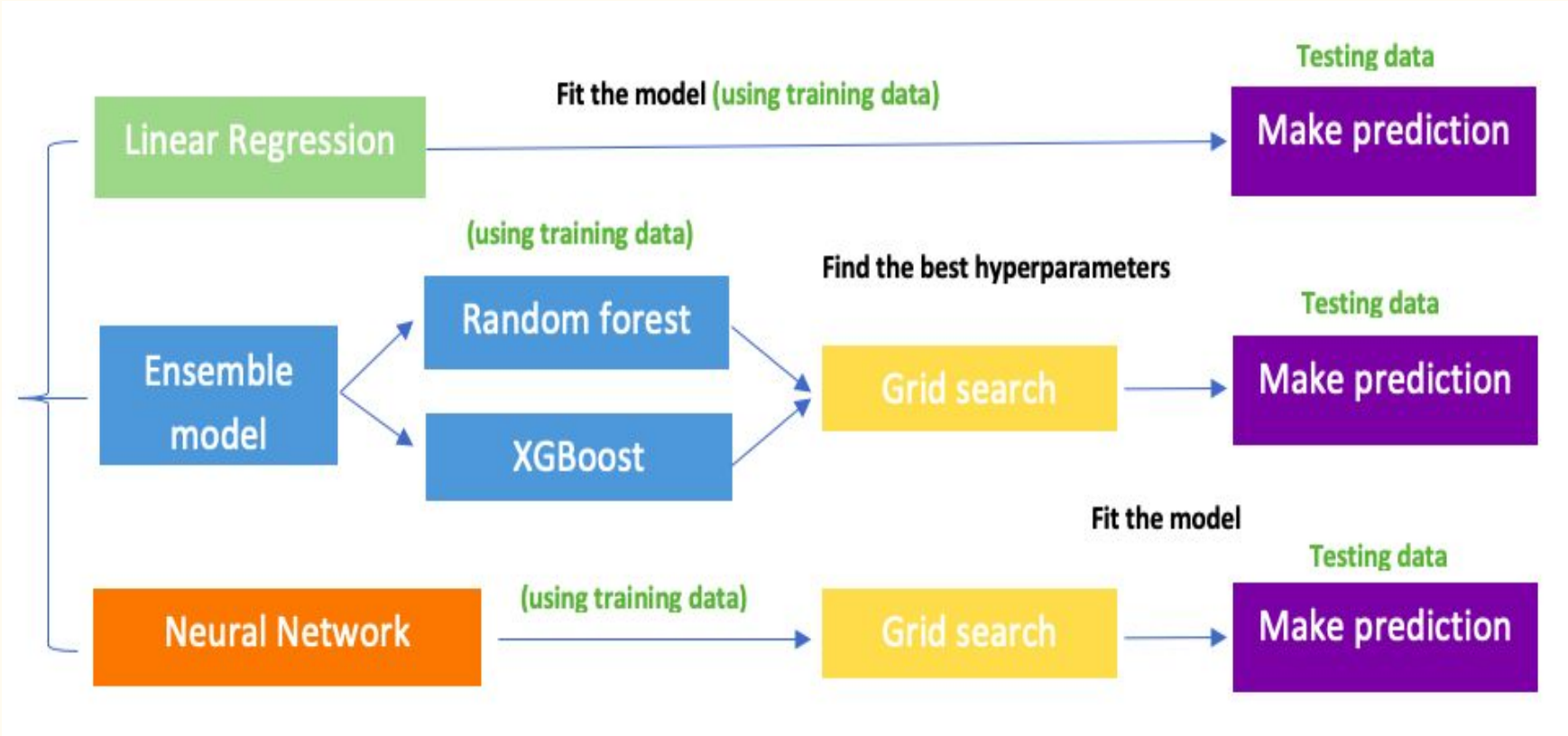
Litecoin



Monero

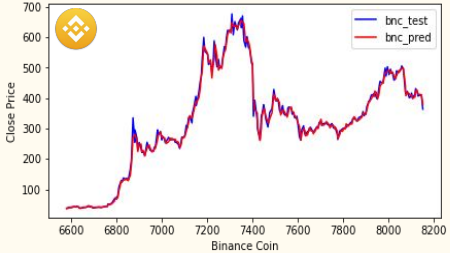


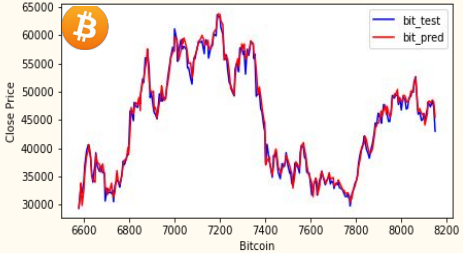
Machine Learning models

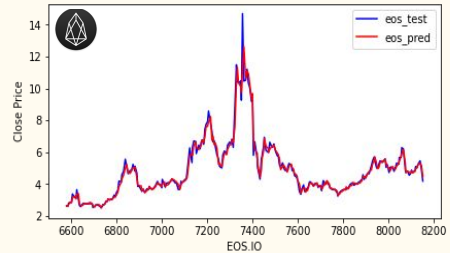


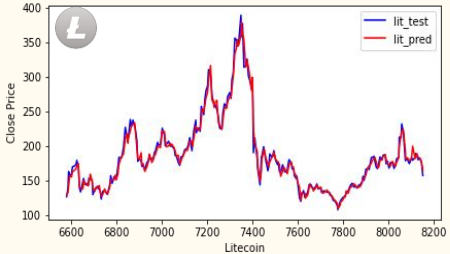
Analysis Results

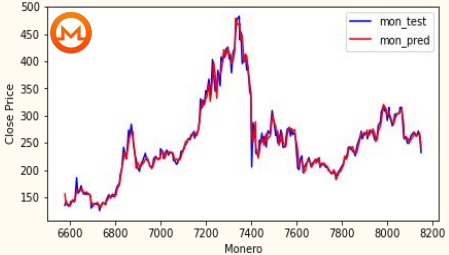
Preditions by Linear Regression



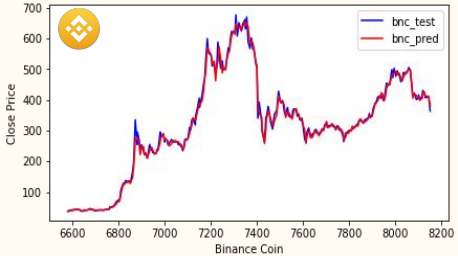


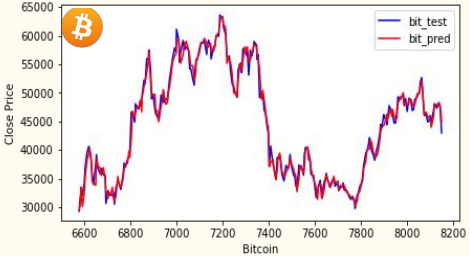


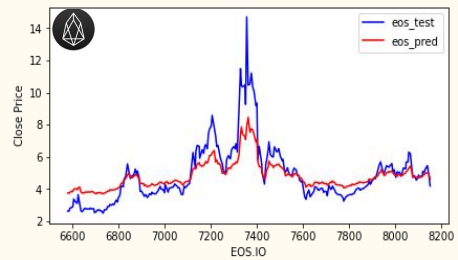


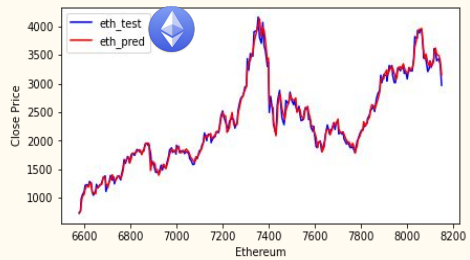


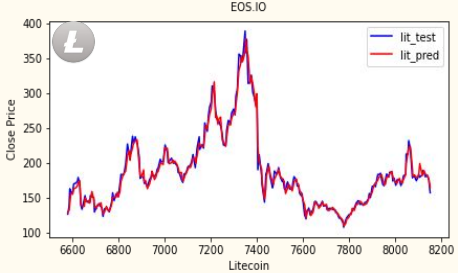
Preditions by Artifical Neural Network

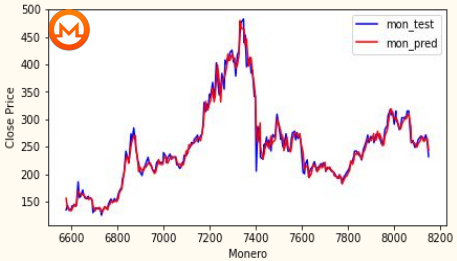




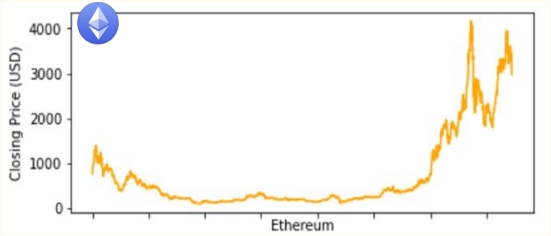






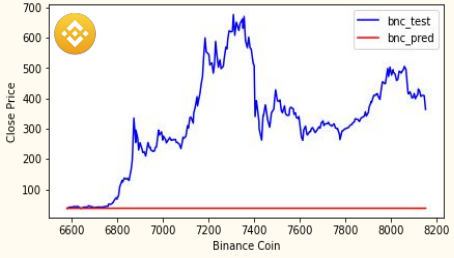


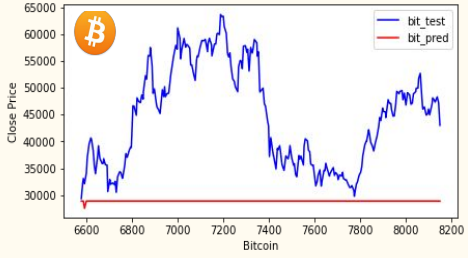
Analysis Results

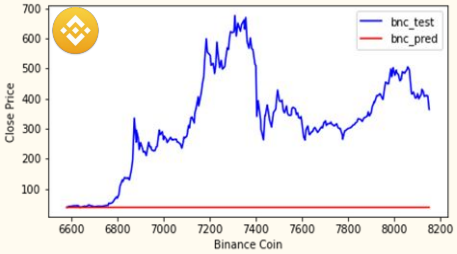


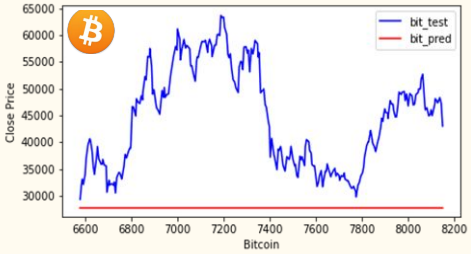
Preditions by XGBoost

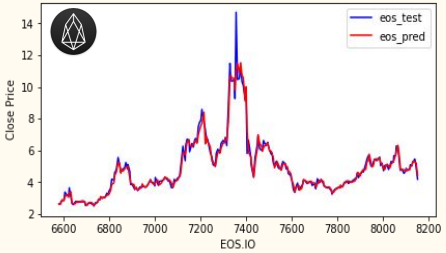
Preditions by Random Forest

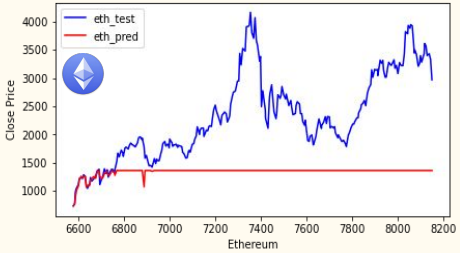


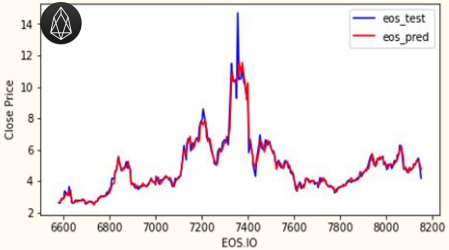


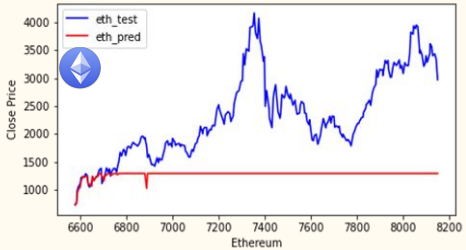


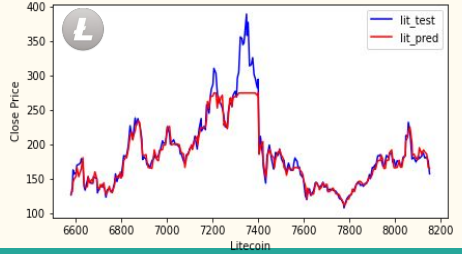


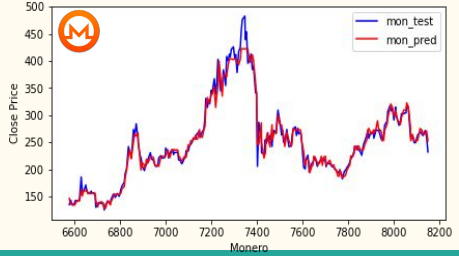


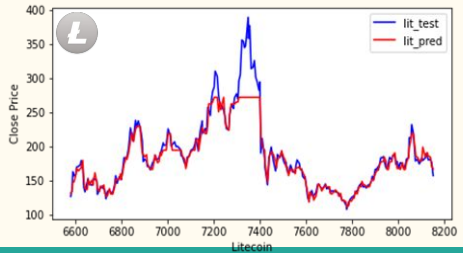


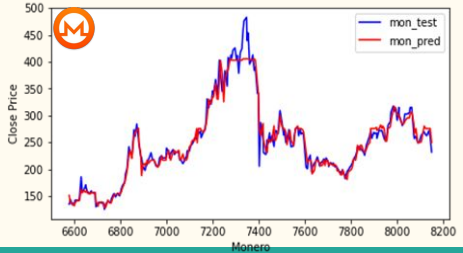












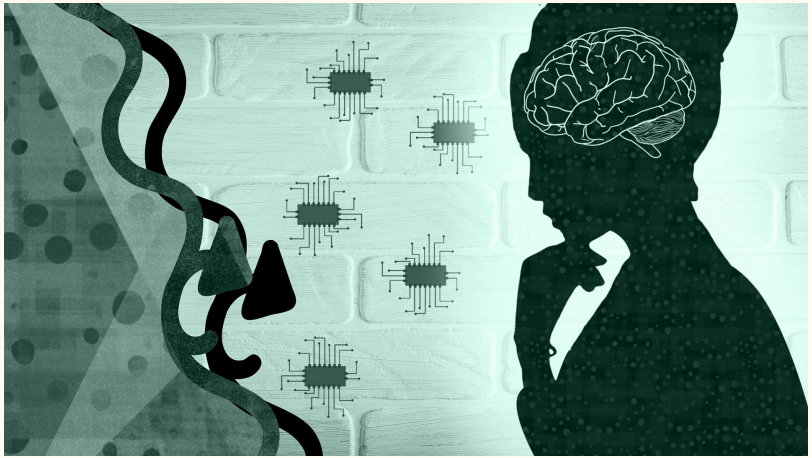
Analysis Results continued...

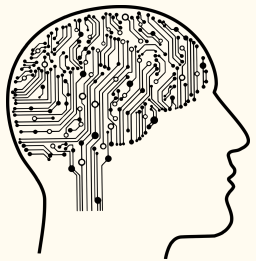
Tableau Public Dashboard

ML Models	Binance Coin	Bitcoin	Eos.IO	Ethereum	Litecoin	Monero
Linear Regresssion	0.993	0.985	0.964	0.990	0.980	0.976
Random Forest	-3.132	-3.333	0.958	-1.803	0.889	0.961
xGBoost Regression	-3.132	-2.921	0.965	-1.556	0.895	0.971
Artificial Neural Network	0.993	0.986	0.694	0.990	0.980	0.976

Questions We Hope to Answer with Data

- Which Machine Learning Model would best predict price changes on cryptocurrency from historical price?
- Which asset(s) is more stable out of the 6 cryptocurrencies chosen for prediction?
- Which features affect the close price the most?





Recommendations for Future Analysis

- Use Time Series API for the forecasting portion of the analysis
- Utilized open-sourced library called Prophet designed for making forecasts
- Utilized yfinance package to pull real time data to make comparison to model prediction
- Utilized other models for forecasting (LSTM)

Anything The Team Would Have Done Differently

- Allocate time to explore time series forecasting, experiment with other libraries like Prophet and other ML models (LSTM, ...)
- Ensemble models has limitations - they cannot extrapolate beyond what they are trained for. Be mindful of the training data.