Project 1: Bitcoin Butler

Josh, Daniel, and Kenny

Project Overview

*Utilization of machine learning and training data to predict crypto price.

Selected Models

*Linear regression

*Dense/Dropout/LSTM

*Sequential

*MinMaxScaler

*Train_test_split

Data Prep

*Upload the "BTC-USD.csv" file into Jupyter, then store in a Pandas DataFrame

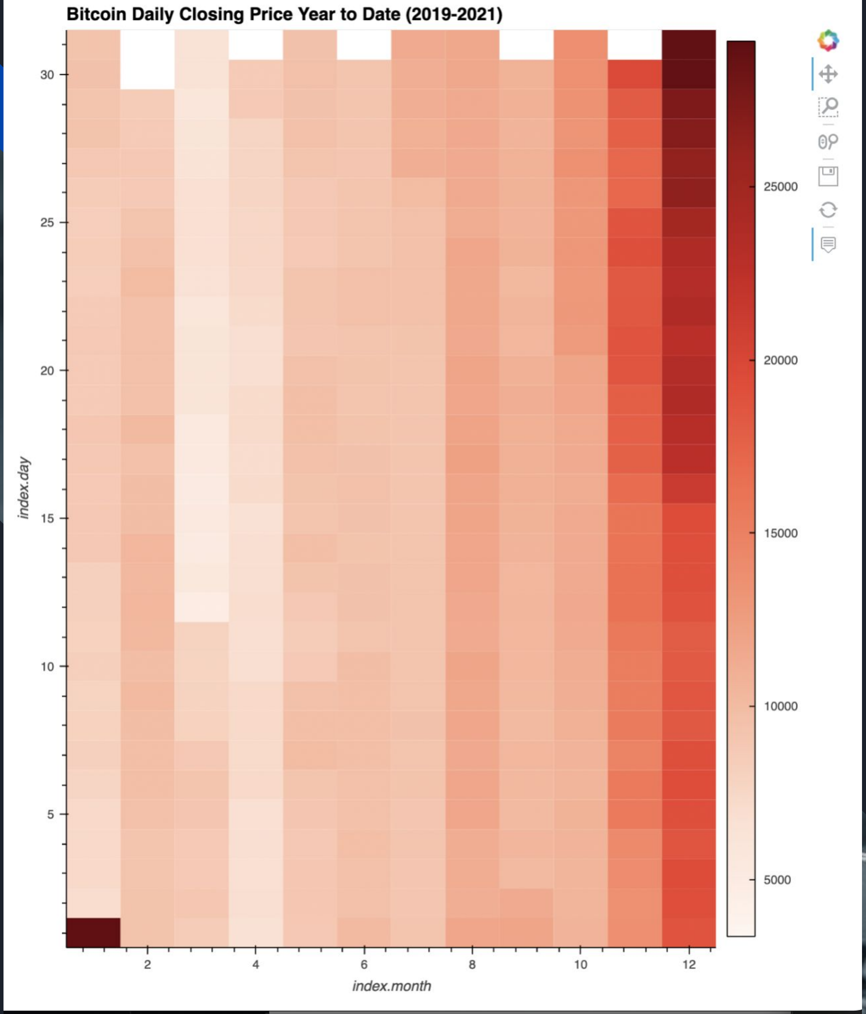
*BTC data from 2019-01-01 to 2021-01-01 via yfinance API

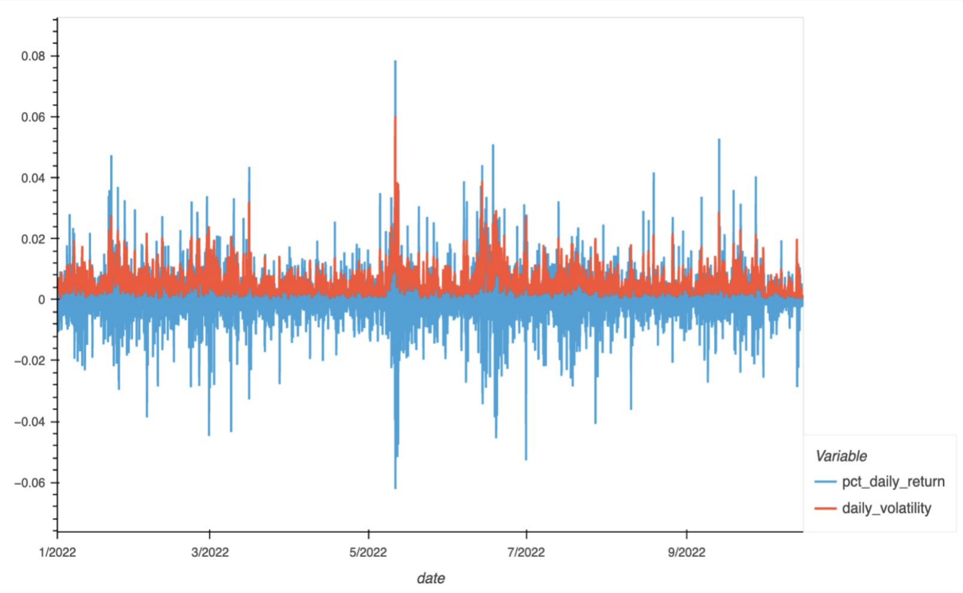
Group Approach

*Josh-time series

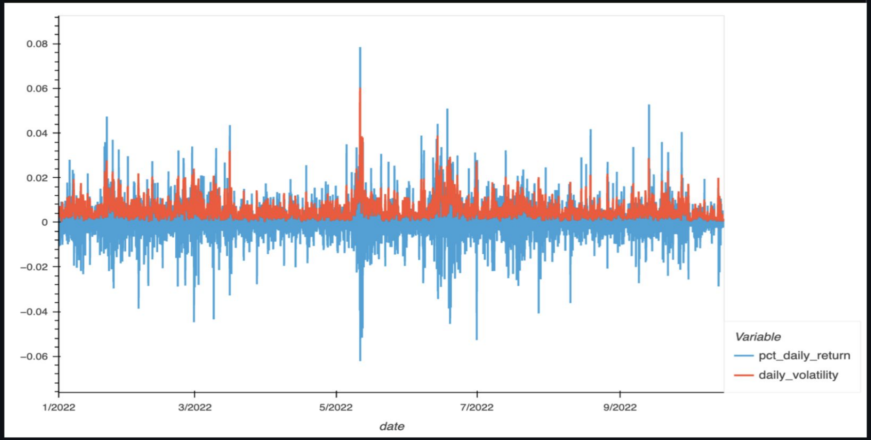
*Kenny- neural network model

*Daniel-Linear Regression

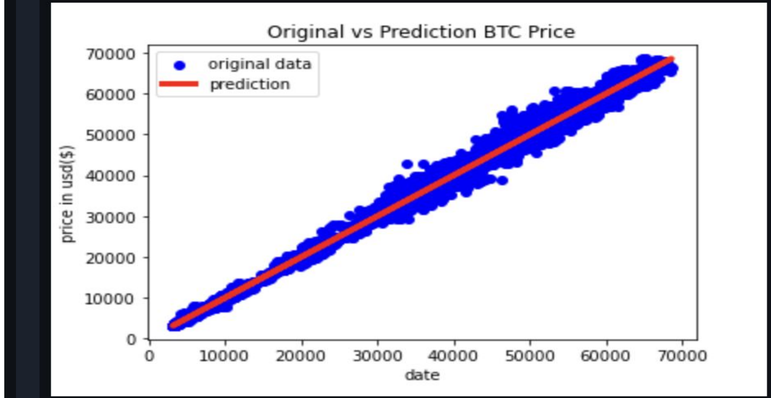

Bitcoin Daily Closing Price Year to Date (2019-2021)
30
25
20
15
10
5
index.day
2
4
6
8
10
12
index.month
25000
20000
15000
10000
5000

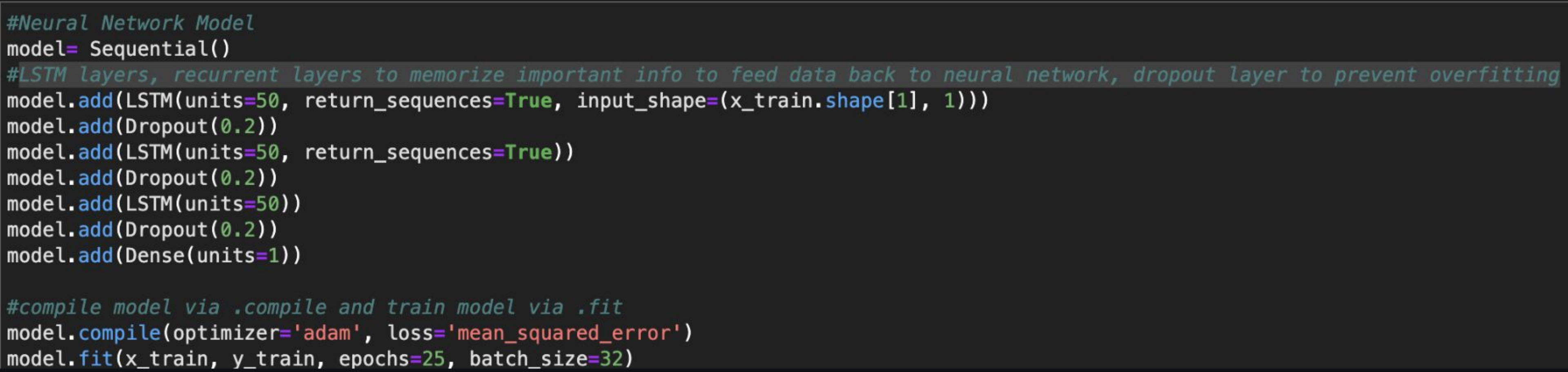

0.08
0.06
0.04
0.02
0
−0.02
−0.04
−0.06
1/2022
3/2022
5/2022
7/2022
9/2022
date
Variable
pct_daily_return
daily_volatility

Plot of BTC daily returns and volatility



Plot of Original vs Prediction BTC Price





```
#Neural Network Model
model= Sequential()
#LSTM layers, recurrent layers to memorize important info to feed data back to neural network, dropout layer to prevent overfitting
model.add(LSTM(units=50, return_sequences=True, input_shape=(x_train.shape[1], 1)))
model.add(Dropout(0.2))
model.add(LSTM(units=50, return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(units=50))
model.add(Dropout(0.2))
model.add(Dense(units=1))

#compile model via .compile and train model via .fit
model.compile(optimizer='adam', loss='mean_squared_error')
model.fit(x_train, y_train, epochs=25, batch_size=32)
```

Results and Conclusions

*Results were achieved

*Machine learning is difficult