BITCOIN PRICE PREDICTION

Ang Kai En
Kim Yeontae
Ng Jing Xiang
Sim Jun You
Wang Zihan
Wong Cheuk Wah (Jane)

BRIEF BITCOIN HISTORY

2009

START

Start of use of Bitcoin as a cryptocurrency

2021

PEAK

Value reached its all time high at 65,000 USD in November 2021

2022

FALL

Prices crashed due to several crises, bankruptcies and market turmoil

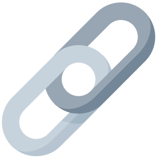
TABLE OF CONTENTS

Data Collection

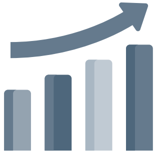
What information can we use to predict bitcoin price changes?

OVERVIEW OF DATA SOURCES

Market Data



On-Chain Data



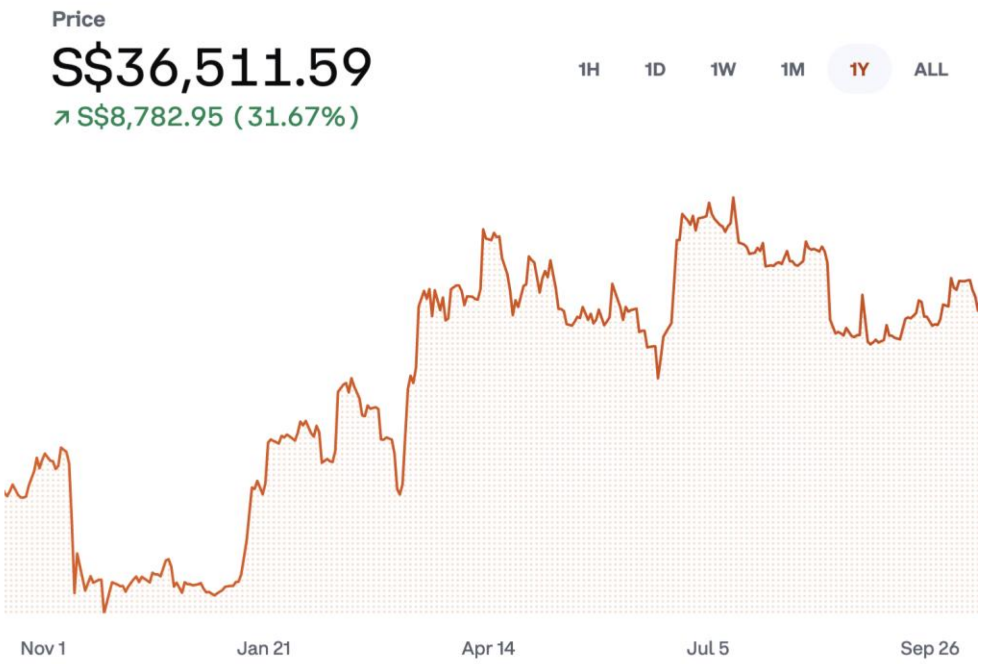
Autoregressive Data

Futures Market

Google Trends

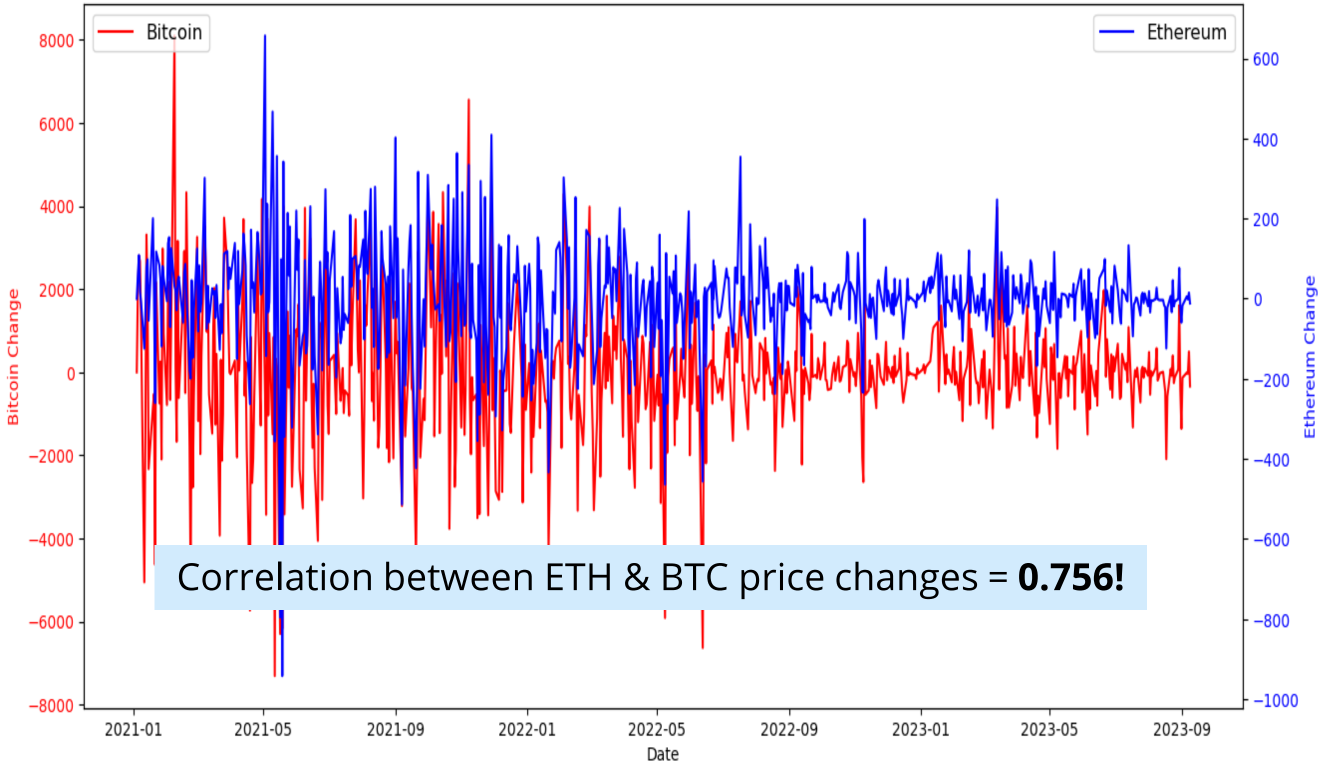
VADER

MARKET DATA



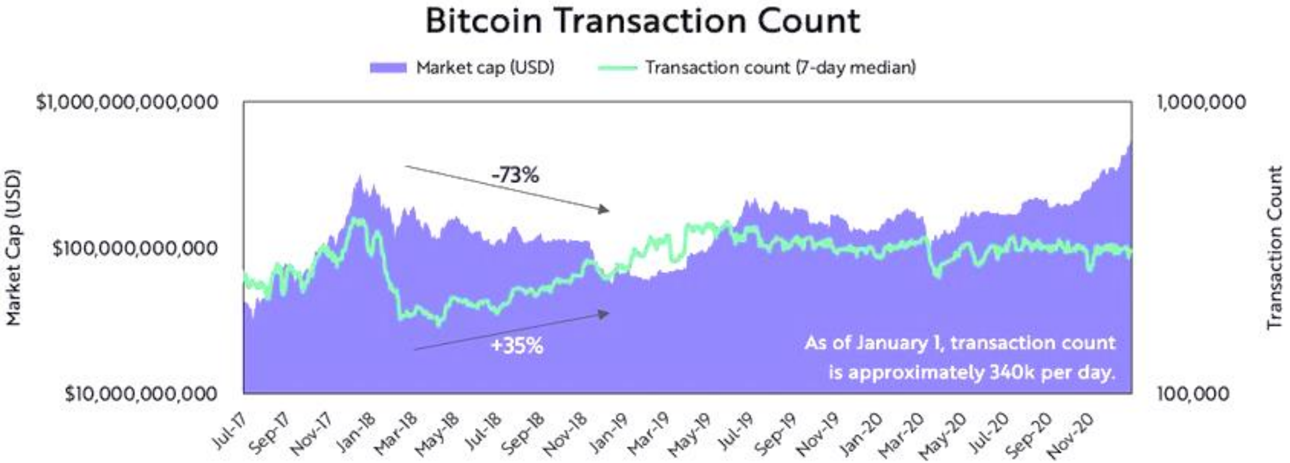
https://www.coinbase.com/price/bitcoin/

ETHEREUM DATA



Correlation between ETH & BTC price changes = **0.756!**

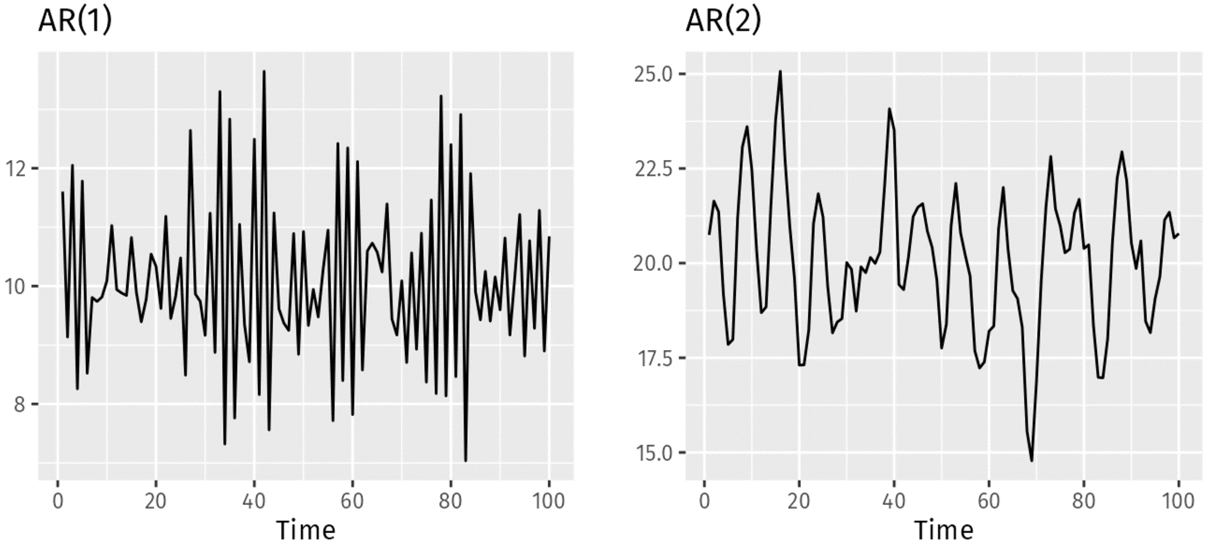
ON-CHAIN DATA



https://ark-invest.com/articles/analyst-research/on-chain-data-bitcoin/

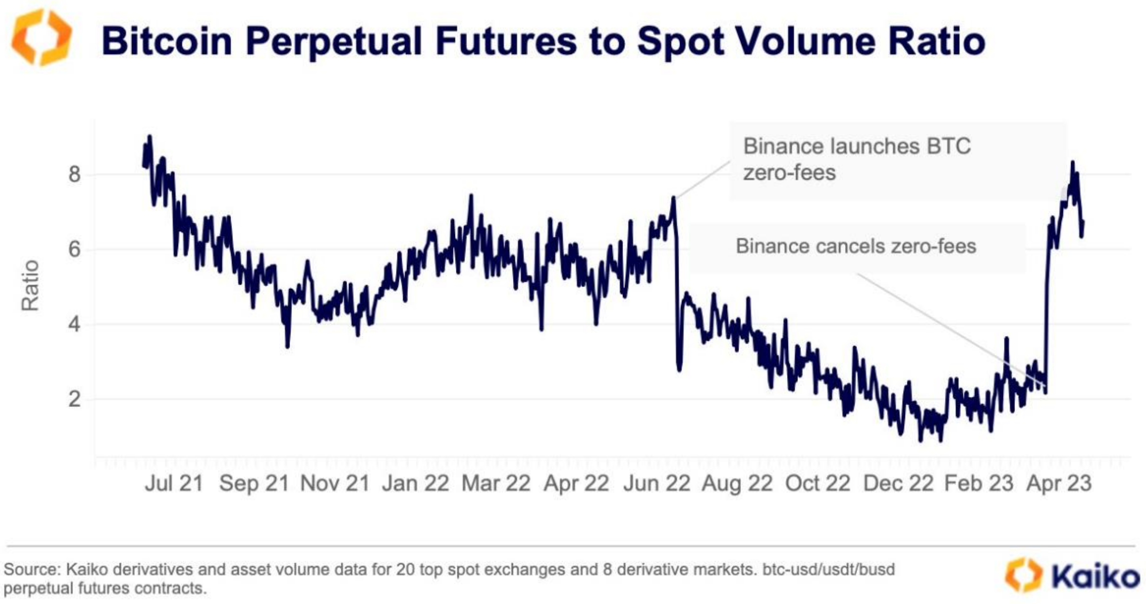
AUTOREGRESSIVE DATA

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \epsilon_t$$



https://otexts.com/fpp2/AR.html

FUTURES DATA



https://www.bloomberg.com/news/articles/2023-05-05/bitcoin-btc-perpetuals-seen-becoming-an-even-bigger-driver-of-prices?embedded-checkout=true

GOOGLE TRENDS DATA

Interest over time

trends.google.com

SENTIMENT ANALYSIS DATA

```
arted/enable-your-device-for-development
  warnings.warn(message)
Downloading (…)lve/main/config.json: 100%|██████████| 758/758 [00:00<?, ?B/s]
Downloading (…)solve/main/vocab.txt: 100%|██████████| 232k/232k [00:00<00:00, 28.9MB/s]
Downloading (…)cial_tokens_map.json: 100%|██████████| 112/112 [00:00<00:00, 112kB/s]
Downloading pytorch_model.bin: 100%|██████████| 438M/438M [00:04<00:00, 101MB/s]
C:\Users\JunYou\Downloads\tweets_sentiment\process_tweets.py:80: DtypeWarning: Columns (4,5,6,7,12) have mixed t
ypes. Specify dtype option on import or set low_memory=False.
  df1 = pd.read_csv(f"Bitcoin_tweets.csv", lineterminator='\n')
C:\Users\JunYou\Downloads\tweets_sentiment\process_tweets.py:9: UserWarning: Could not infer format, so each ele
ment will be parsed individually, falling back to `dateutil`. To ensure parsing is consistent and as-expected, p
lease specify a format.
  df['date'] = pd.to_datetime(df['date'], errors='coerce')
Traceback (most recent call last):
  File "C:\Users\JunYou\Downloads\tweets_sentiment\process_tweets.py", line 131, in <module>
    main()
  File "C:\Users\JunYou\Downloads\tweets_sentiment\process_tweets.py", line 125, in main
    agg_df = combined.groupby('date')['Vader_Comp', 'Vader_Pos', 'Vader_Neg', 'Vader_Neu', 'BERT_SCORE'].mean()
  File "C:\Users\JunYou\miniconda3\envs\tweets\lib\site-packages\pandas\core\groupby\generic.py", line 1767, in
__getitem__
    raise ValueError(
ValueError: Cannot subset columns with a tuple with more than one element. Use a list instead.
```

SENTIMENT ANALYSIS DATA

FinBERT

(Financial Sentiment Analysis with BERT)

- Pre-trained NLP model to analyse sentiment of financial text

https://www.dailyfx.com/forex/market_alert/2021/03/24/Bitcoin-BTCUSD-Latest-Buy-Your-Tesla-With-Bitcoin-Tweets-Elon-Musk.html

SENTIMENT ANALYSIS DATA

VADER

(Valence Aware Dictionary and sEntiment Reasoner)

- Natural Language Toolkit (NLTK) model that provides sentiment scores based on the words used

https://www.analyticsvidhya.com/blog/2021/06/vader-for-sentiment-analysis/

OVERVIEW OF DATA SOURCES

Market Data

On-Chain Data

Autoregressive Data

Futures Market

Google Trends

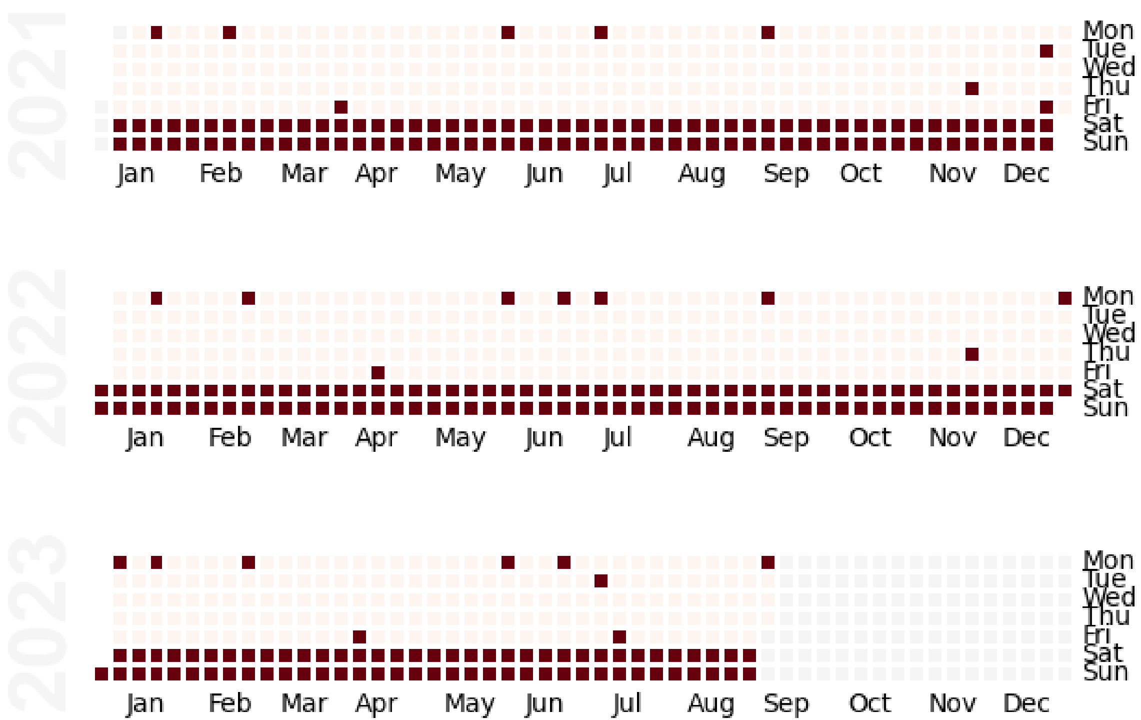
VADER

Data Processing

Preprocessing, dealing with null values, dimensionality reduction

DATES WITH MISSING DATA

Only weekday data is considered (markets closed on weekends)

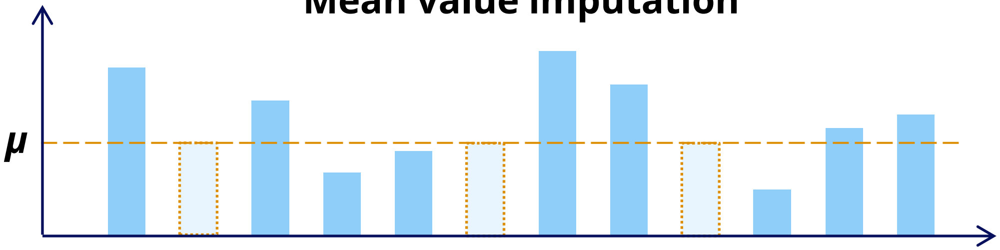


DATA IMPUTATION

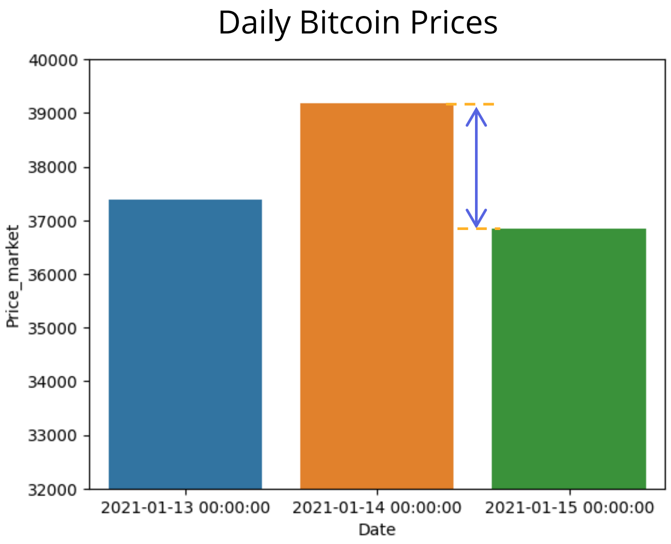
VADER

Futures Market

Mean value imputation



TARGET FEATURES



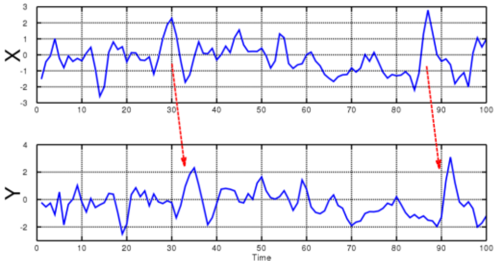
Calculate different forms of price change:

- ***Change*** *= -1793.5*
- ***Fractional change*** *= -0.0479*
- ***Percentage change*** *= -4.79%*

DIMENSIONALITY REDUCTION

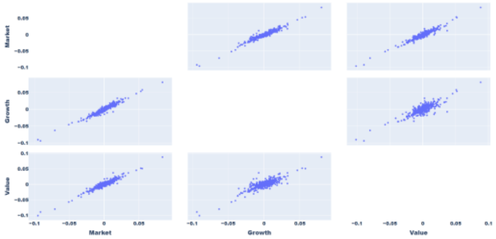
Granger Causality

Statistical Test for **time-based causality** between two features



Variance Inflation Factor

Measure of **multicollinearity** between multiple regression variables



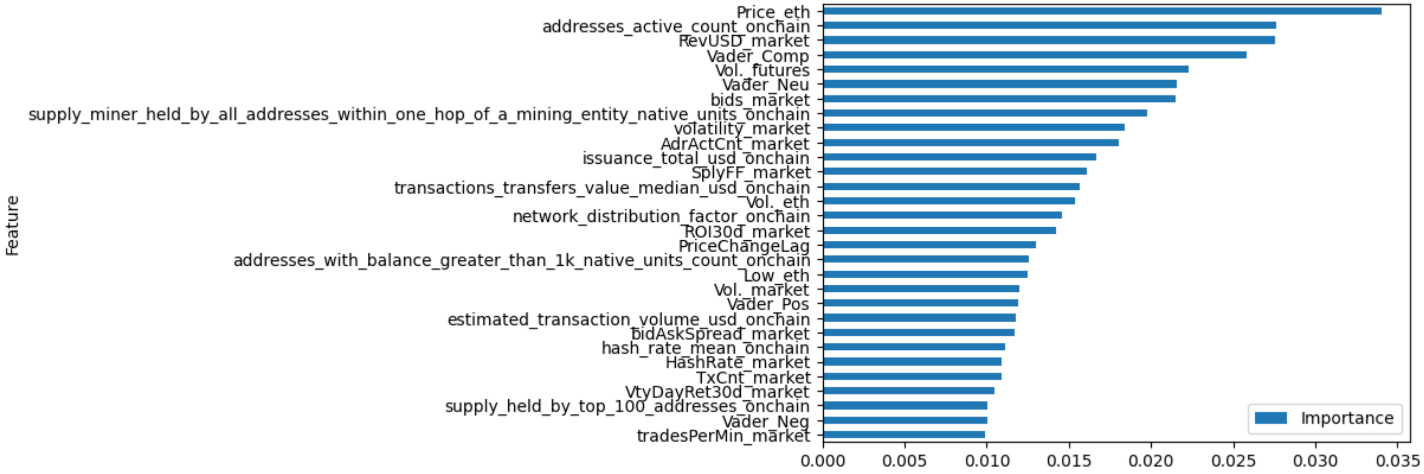
FEATURE IMPORTANCE

Instead of UMAP and PCA, we use the **Gini Impurity** measure calculated by decision tree models.

Through this, we can select top features and reduce dimensionality.

Impurity	Task	Formula	Description
Gini impurity	Classification	$\sum_{i=1}^{C} f_i(1 - f_i)$	f_i is the frequency of label i at a node and C is the number of unique labels.
Entropy	Classification	$\sum_{i=1}^{C} -f_i \log(f_i)$	f_i is the frequency of label i at a node and C is the number of unique labels.
Variance / Mean Square Error (MSE)	Regression	$\frac{1}{N}\sum_{i=1}^{N}(y_i - \mu)^2$	y_i is label for an instance, N is the number of instances and μ is the mean given by $\frac{1}{N}\sum_{i=1}^{N} y_i$
Variance / Mean Absolute Error (MAE) (Scikit-learn only)	Regression	$\frac{1}{N}\sum_{i=1}^{N}\lvert y_i - \mu\rvert$	y_i is label for an instance, N is the number of instances and μ is the mean given by $\frac{1}{N}\sum_{i=1}^{N} y_i$

FEATURE SELECTION



Using the arbitrary threshold of **0.01**, we filter features to keep the most important ones. We are left with **29 features** in our training dataset!

Prediction Methods

How do we effectively predict the daily price changes of Bitcoin?

NEURAL NETWORKS

INPUT SIZE

670 samples, **14** features
Look back **1** day

MODEL LAYERS

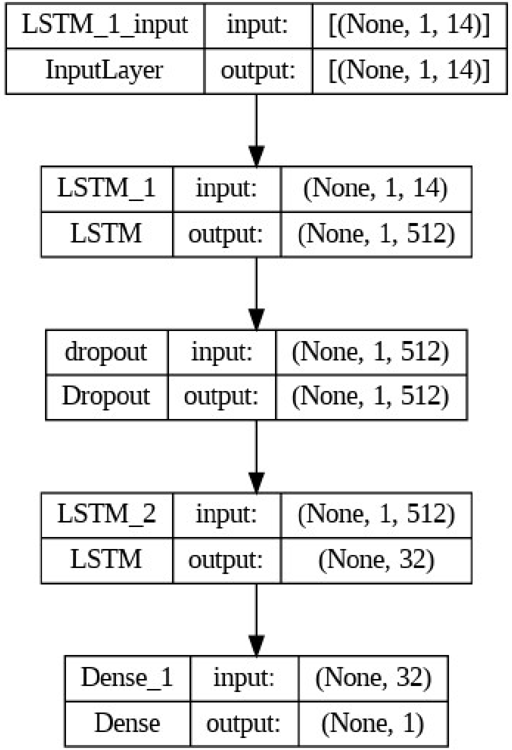
2 LSTM layers
with **64** & **32** neurons respectively

REGULARIZER

Dropout rate: **50%**

Hyperparameters

300 epochs, **32** batch size
Adam Optimizer
Learning Rate: **0.1**



TIME SERIES FORECASTING

ARIMA

PROPHET

To statistically model the time-series in terms of moving averages and seasonality

Apply lagged version of the time series, by making predictions with lag = 1, differentiation = 0

Decompose the time series data

Combine the level, trend, seasonality and noise **additively**
Set the seasonal period to be **50** days

Fit the data into the ARIMA model

Order of autoregressive model = **1**
Degree of differencing = **0**
Order of moving-average model = **0**

Forecast with univariate features

Input historical figures

Forecast the price change and the uncertainty interval (the upper & lower bounds of predicted price change)

GRADIENT BOOSTED TREE

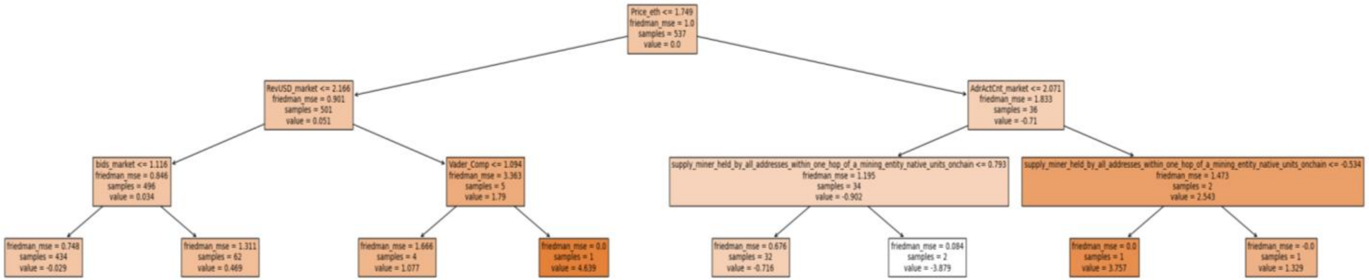
A generally powerful **domain-agnostic model**

Combine multiple weak learners into a **strong predictive model**

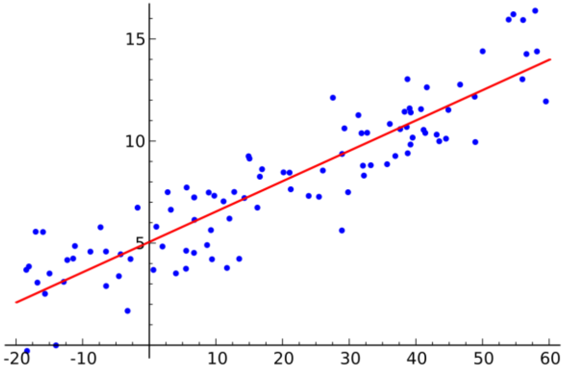
Hyperparameters

3
MAXIMUM DEPTH

100
NUMBER OF ESTIMATORS



LINEAR REGRESSION



Simple **multiple linear regression model**

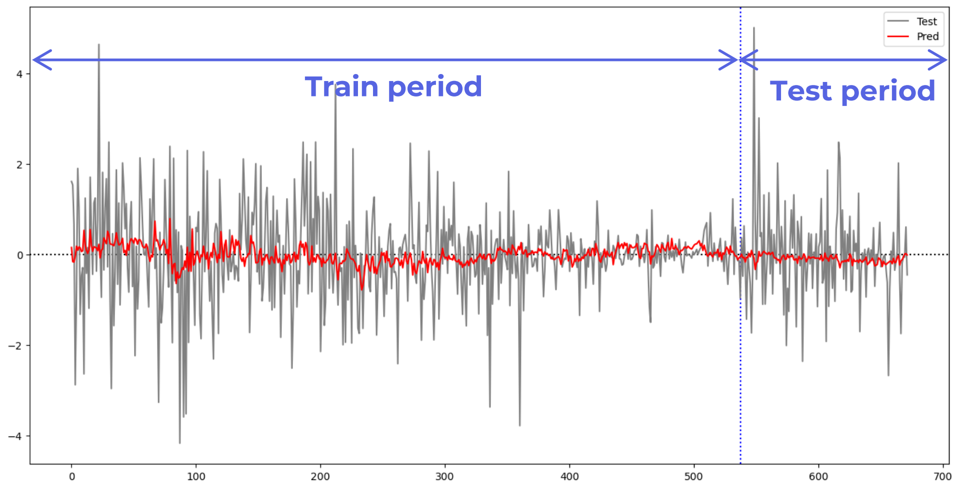
No extra optimizations done

Serves as the **baseline model** to compare all other models against!

Prediction Metrics

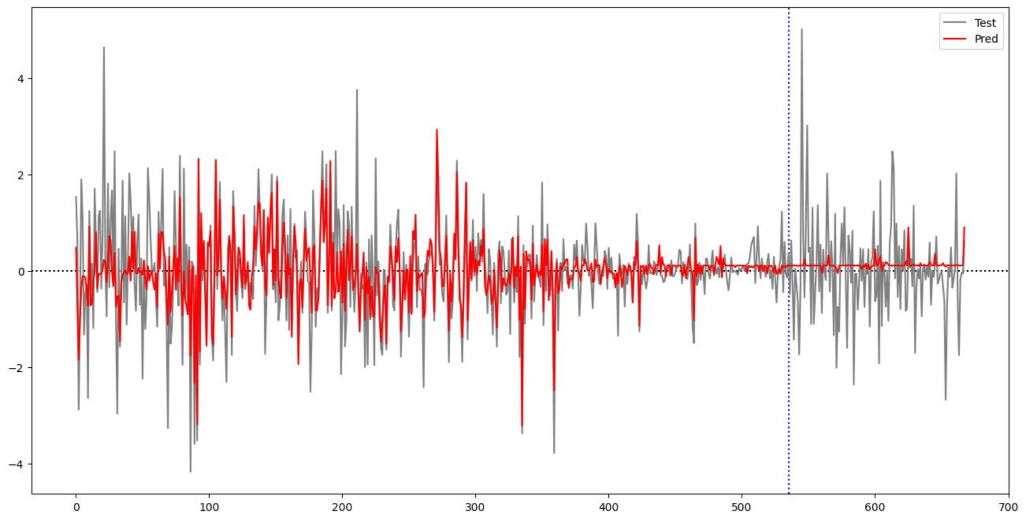
How did our models perform?

LINEAR REGRESSION



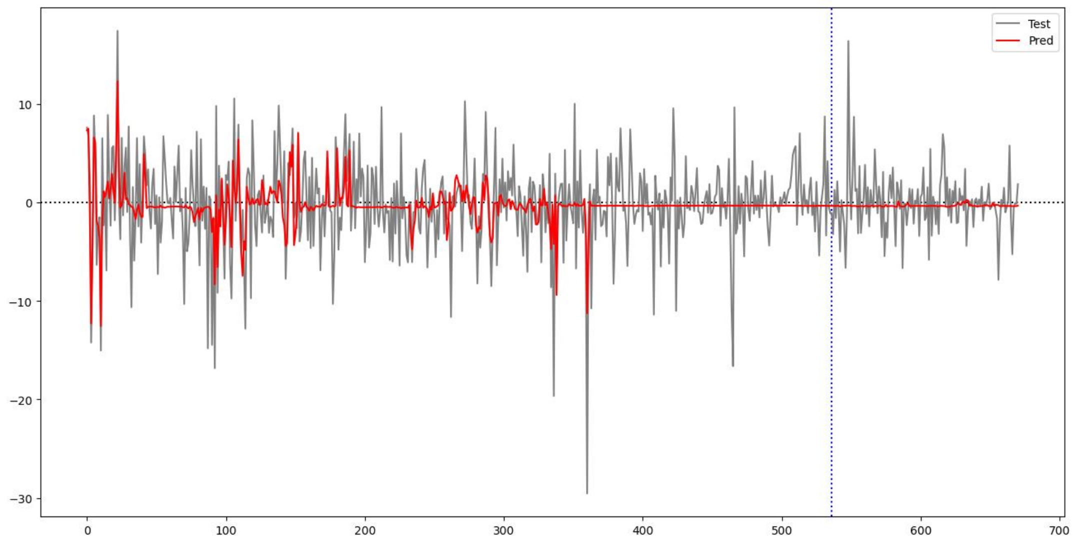
RMSE: 1.0097

LSTM



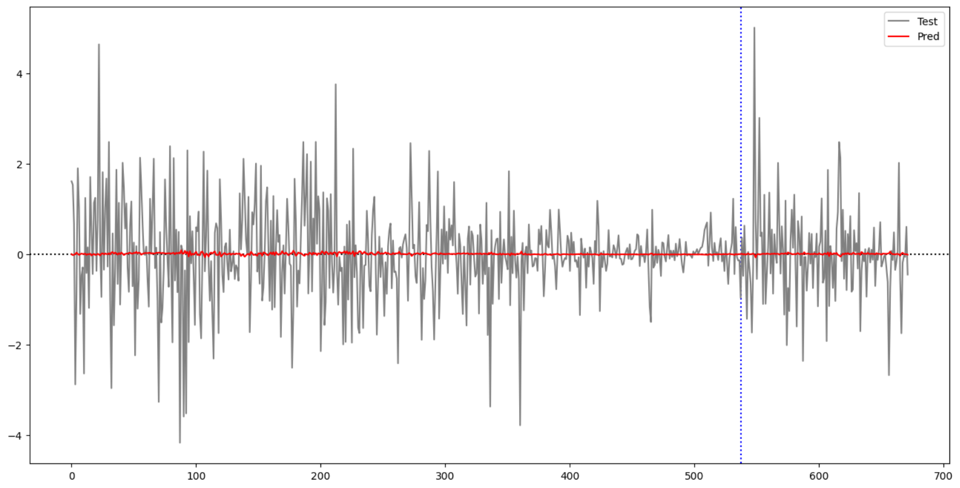
RMSE: 1.0043

LSTM
with Percentage Change



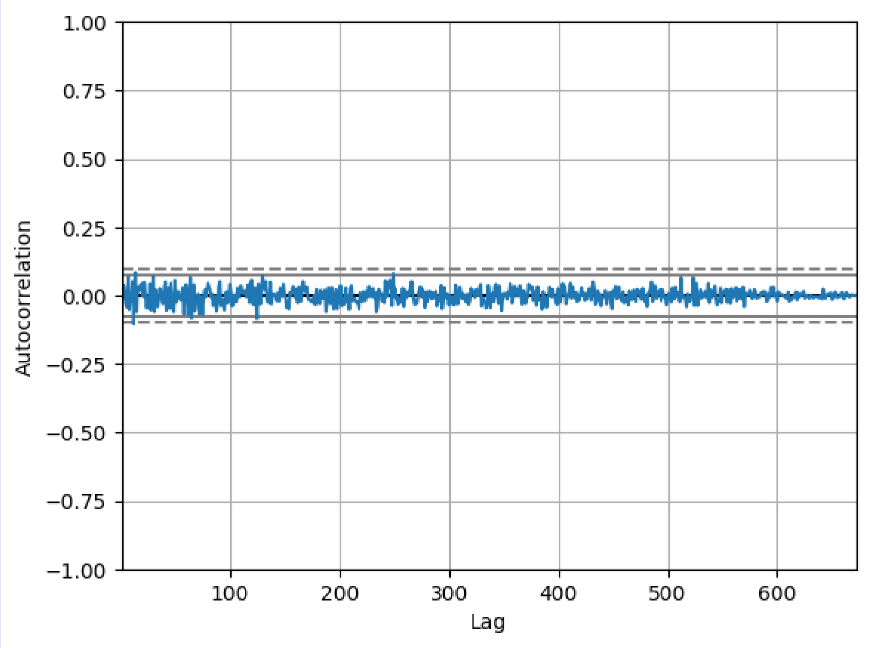
Flat prediction in test period!

ARIMA

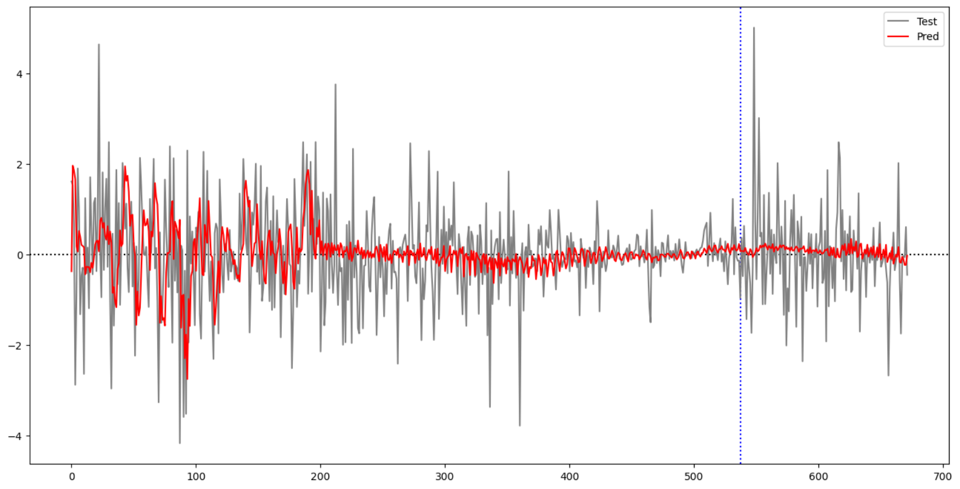


RMSE: 1.0012

Price Change Autocorrelation

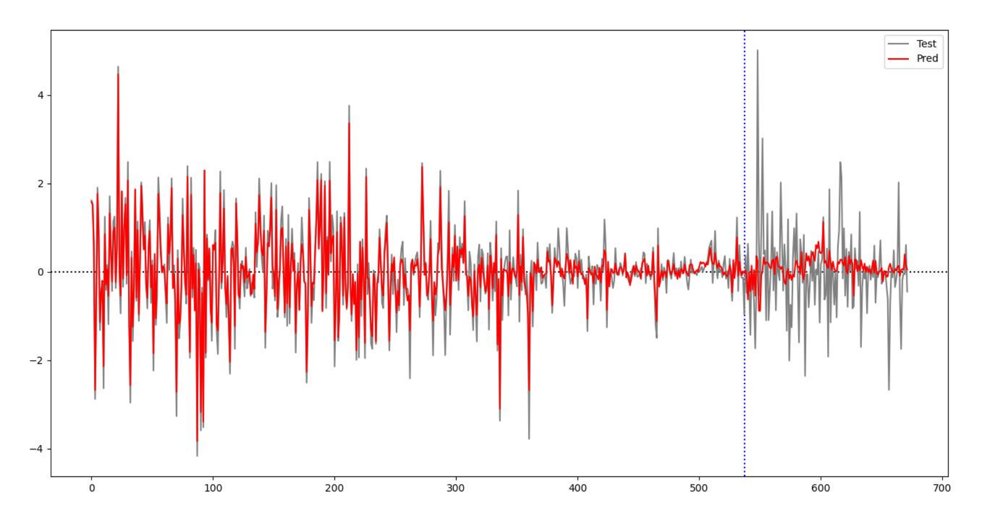


Prophet



RMSE: 1.0089

Gradient Boosting



RMSE: 0.994

Purchasing Simulation

Utilising our model to trade

TRADING STRATEGY

Strategy: Swing Trading
Focuses on taking smaller gains in short term trends and cutting losses quicker

If model indicates...

Positive Price Change: Initiate Buy Order

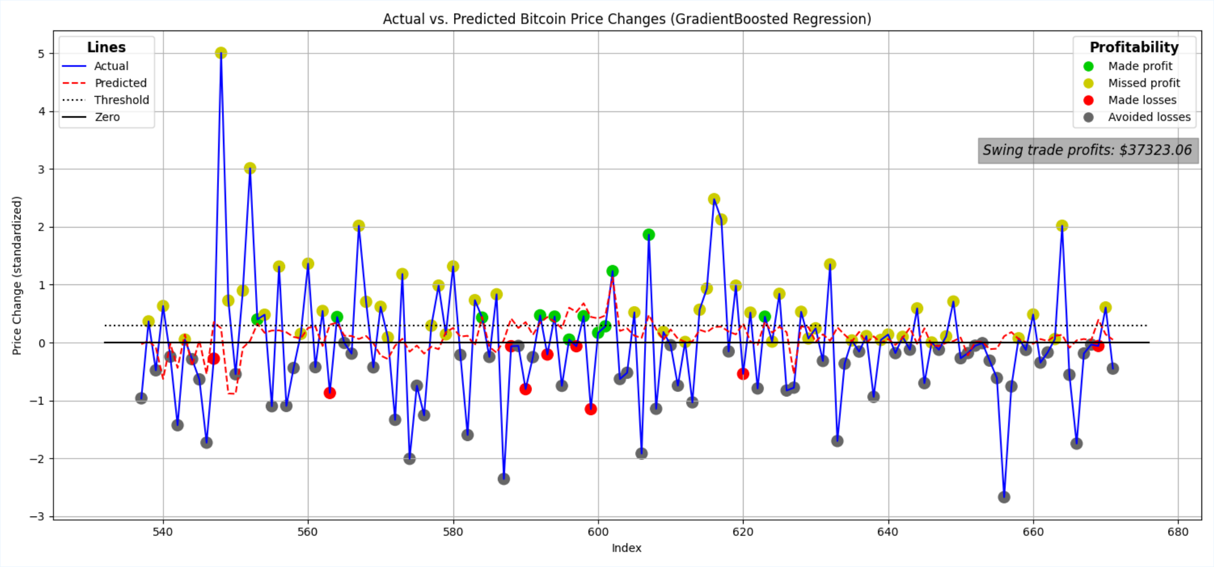
Buy one more Bitcoin before prices rise tomorrow, as long as we have enough money.

Negative Price Change: Execute Sell Order

Sell all our Bitcoin today, before prices drop tomorrow!

TRADING SIMULATION

Initial capital: $1,000,000
Transaction fee: 2%



Trading profit (%): 3.73%

Conclusion

We're nearly at the end!

LEARNING POINTS

Importance of good data and feature selection

Understanding and evaluation of different models

Model must be coupled with market knowledge

AREA FOR IMPROVEMENTS

More reliable data
Feature engineering
Model tuning

THANK YOU

Ang Kai En
Kim Yeontae
Ng Jing Xiang
Sim Jun You
Wang Zihan
Wong Cheuk Wah (Jane)