Predictive Analysis of Bitcoin Returns

A Comparative Study of Time Series Models

BY HYOJUN KIM

Agenda

1. PROBLEM STATEMENT
2. ASSUMPTIONS AND HYPOTHESES
3. DATA PREPARATION
4. DATA PROPERTIES AND EXPLORATORY DATA ANALYSIS (EDA)
5. DATA PROCESSING AND TRANSFORMATION
6. FEATURE ENGINEERING
7. PROPOSED MODELING APPROACHES
8. MODEL RESULTS AND LEARNING
9. FUTURE WORK
10. GITHUB LINK

Problem Statement:
Bitcoin Return Predictions

The goal is to explore, develop and compare the predictive accuracy of various time-series models in forecasting future Bitcoin returns.

Bitcoin:

- A DECENTRALIZED DIGITAL CURRENCY WITH SIGNIFICANT MARKET CAP AND LIQUIDITY.

Significance:

- FORECASTING BITCOIN RETURNS CAN GUIDE INVESTORS, IMPROVE RISK MANAGEMENT, AND AID IN STRATEGIC PLANNING

Forecasting Importance:

- CRUCIAL FOR INFORMED DECISION-MAKING IN THE VOLATILE BITCOIN MARKET.

Methodology:

- APPLICATION OF TIME-SERIES MODELS (ARIMA, ARFIMA, ETS, SARIMA)

Assumptions and Hypotheses

Assumptions

- Market Efficiency:

  Bitcoin markets are efficient, meaning that current prices reflect all available information.

- Data Integrity:

  The historical data used for analysis is assumed to be accurate and reliable.

- No Influence of External Factors:

  price and return are not significantly influenced by external events, such as regulatory changes or macroeconomic factors, that are not captured in the data.

Hypotheses

- Predictive Power of Past Data:

  The past trading data of Bitcoin has predictive power on its future returns.

- Influence of Technical Indicators:

  Technical indicators derived from the trade data, like trade volume and OHLC prices, have a significant impact on Bitcoin returns.

- Model Accuracy:

  Time-series models such as ARIMA, ARFIMA, ETS, and SARIMA can accurately predict Bitcoin returns.

**BY UNDERSTANDING THE ASSUMPTIONS AND HYPOTHESES,
THE BEST MODEL FOR FORECASTING BITCOIN RETURNS IS ACCURATE AND ROBUST.**

Data Preparation

DATA LOADING

Imported Bitcoin historical trading data from a CSV file into R.

TIMESTAMP CONVERSION

Converted timestamps to a standardized datetime format.

DATA STRUCTURE ANALYSIS

Assessed the data structure and basic summary statistics for each variable.

MISSING VALUES

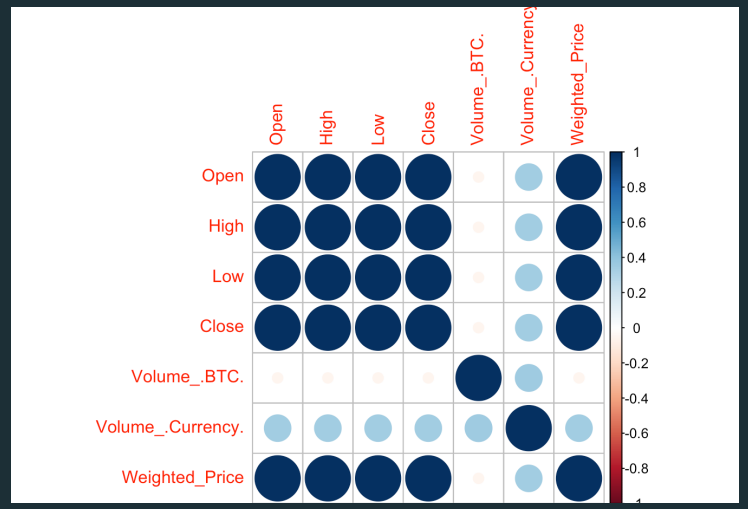
Identified and quantified missing values within the dataset.

Data Properties and Exploratory Data Analysis

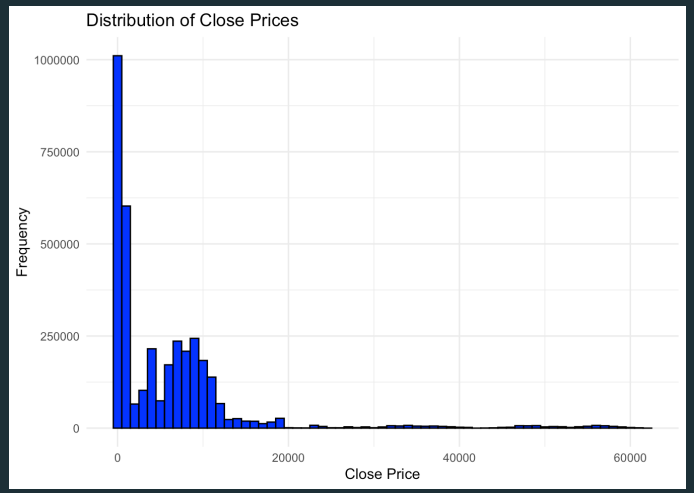
DATA PROPERTIES:

```
##    Timestamp                       Open             High
##  Min.   :2011-12-31 07:52:00.00   Min.   :    3.8   Min.   :    3.8
##  1st Qu.:2014-04-22 14:56:00.00   1st Qu.:  443.9   1st Qu.:  444.0
##  Median :2016-08-17 09:52:00.00   Median : 3597.0   Median : 3598.2
##  Mean   :2016-08-15 22:39:26.50   Mean   : 6009.0   Mean   : 6013.4
##  3rd Qu.:2018-12-08 16:56:00.00   3rd Qu.: 8627.3   3rd Qu.: 8633.0
##  Max.   :2021-03-31 00:00:00.00   Max.   :61763.6   Max.   :61781.8
##                                   NA's   :1243608   NA's   :1243608
##       Low               Close          Volume_.BTC.      Volume_.Currency.
##  Min.   :    1.5   Min.   :    1.5   Min.   :   0.0   Min.   :       0
##  1st Qu.:  443.5   1st Qu.:  443.9   1st Qu.:   0.4   1st Qu.:     452
##  Median : 3595.6   Median : 3597.0   Median :   2.0   Median :    3810
##  Mean   : 6004.5   Mean   : 6009.0   Mean   :   9.3   Mean   :   41763
##  3rd Qu.: 8621.1   3rd Qu.: 8627.2   3rd Qu.:   7.3   3rd Qu.:   25698
##  Max.   :61673.6   Max.   :61781.8   Max.   :5853.9   Max.   :13900672
##  NA's   :1243608   NA's   :1243608   NA's   :1243608   NA's   :1243608
##  Weighted_Price
##  Min.   :    3.8
##  1st Qu.:  443.8
##  Median : 3596.8
##  Mean   : 6008.9
##  3rd Qu.: 8627.6
##  Max.   :61716.2
##  NA's   :1243608
```

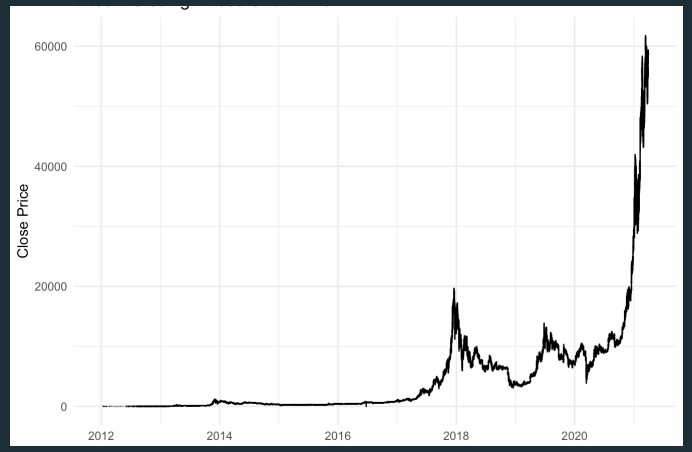
CORRELATION PLOT:



CLOSING PRICE HISTOGRAM:



BITCOIN CLOSING PRICE:



STATIONARITY:

```
##
##  Augmented Dickey-Fuller Test
##
## data:  sample_data
## Dickey-Fuller = -9.6449, Lag order = 9, p-value = 0.01
## alternative hypothesis: stationary
```

- THE SAMPLE SIZE IS 1,000.
- SINCE P-VALUE < 0.05, THIS SHOWS STATIONARITY.

EDA:

- MISSING VALUES: 8,705,256
- OBSERVATION: 4,857,377
- VARIABLES: 8
- CLOSE IS HIGHLY CORRELATED WITH OPEN, HIGH, LOW, AND WEIGHTED_PRICE
- AS CLOSING PRICE DECREASES, FREQUENCY INCREASES

Data Processing and Transformations

Data Processing:

- Data cleaning:

  The dataset was examined to identify and address inaccuracies.

- Feature selection:

  Careful selection of relevant features/variables (e.g. trade volume, opening/closing prices, technical indicators) to capture info for predicting Bitcoin returns.

- Splitting the data:

  The dataset was split into training and testing sets; the former used to build models, the latter to evaluate them.

Anomaly Detection, Cleansing, Imputations, & Transformations:

- Anomaly Detection:

  Techniques were employed to identify and address anomalies in the dataset using statistical methods and domain knowledge-based approaches.

- Data Cleansing:

  Data quality issues addressed through validation, filtering, and removal of problematic observations to ensure dataset reliability.

- Imputations:

  Missing values handled using spline interpolation, filling in based on overall trend and pattern of available data.

- Transformations:

  Various techniques such as rolling st. dev., SMA, RSI, and MACD calculation were applied for feature engineering to improve data quality and meet modeling assumptions. Also, rolling mean was calculated.

Feature Engineering

- **ROLLING STANDARD DEVIATION:**

Calculated the rolling standard deviation for the 'Close' price with a window size of 50. This captures the volatility over time.

- **SIMPLE MOVING AVERAGE (SMA20):**

Computed the 20-period simple moving average for the 'Close' price, giving an understanding of the average price over the past 20 periods.

- **RELATIVE STRENGTH INDEX (RSI14):**

Calculated the 14-period Relative Strength Index, which is a momentum oscillator that measures the speed and change of price movements.

- **MOVING AVERAGE CONVERGENCE DIVERGENCE (MACD):**

The MACD is a trend-following momentum indicator that shows the relationship between two moving averages of a security's price.

- **7-DAY MOVING AVERAGE (CLOSEMA7):**

Calculated the 7-day moving average for the 'Close' price, providing insights into the average price over the past week.

Proposed Modeling Approaches

Autoregressive Integrated Moving Average (ARIMA):

ARIMA models enable predictions based on past values, taking into account trends, seasonality, and noise in time-series data.

Fractionally Differenced ARIMA (ARFIMA):

ARFIMA models are suitable for long/short memory data series and can generalize ARIMA for slowly decaying autocorrelations in residuals.

Exponential Smoothing State Space Model (ETS):

This model provides a comprehensive framework to detect level, trend and cyclical patterns in time series data.

Seasonal ARIMA (SARIMA):

This model extends ARIMA by adding a seasonal component, useful for seasonal data.

Model Results and Learning

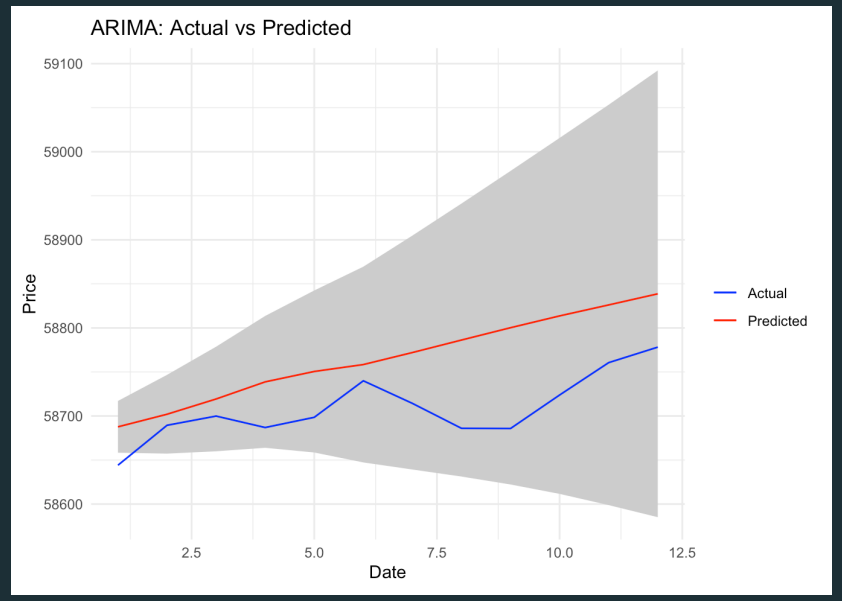
Model Results:

ARFIMA	RMSE	MAE	MAPE
Training Set	31.07	11.09	15.24%
Test Set	409.30	384.59	0.65%
ARIMA	**RMSE**	**MAE**	**MAPE**
Training Set	14.98	4.51	0.11%
Test Set	64.97	57.17	0.097%
ETS	**RMSE**	**MAE**	**MAPE**
Training Set	13.84	4.09	0.11%
Test Set	46.80	36.37	0.062%
SARIMA	**RMSE**	**MAE**	**MAPE**
Training Set	14.98	4.51	0.11%
Test Set	64.97	57.17	0.097%

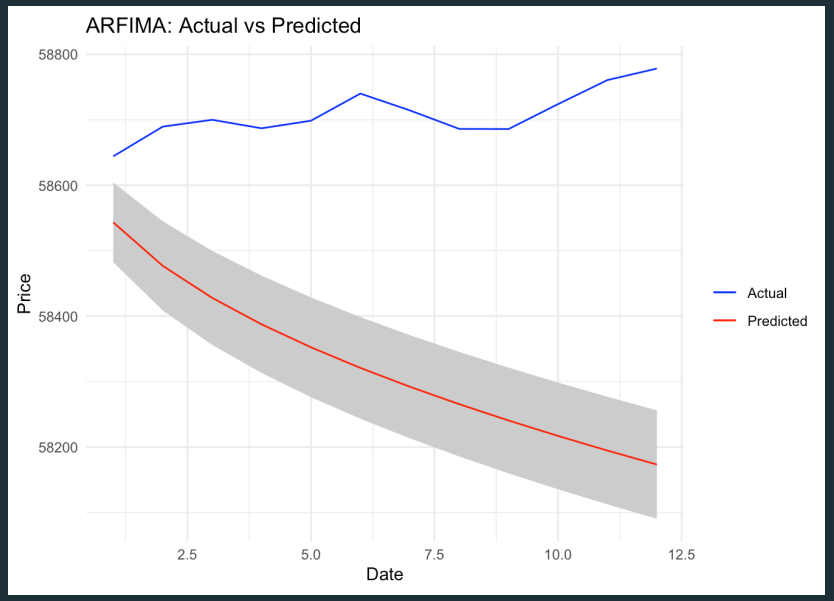
Key Learnings:

- ARFIMA MODEL HAD A HIGH ERROR RATE IN THE TEST SET, INDICATING POTENTIAL OVERFITTING ISSUES.
- ARIMA AND SARIMA MODELS PERFORMED SIMILARLY ON BOTH THE TRAINING AND TEST SETS, SUGGESTING THEY MIGHT BE APPROPRIATE FOR OUR DATA.
- THE ETS MODEL HAD THE LOWEST RMSE AND MAE ON THE TEST SET, INDICATING IT MIGHT BE THE MOST ACCURATE MODEL FOR FUTURE FORECASTING.
- THERE MAY BE ROOM FOR FURTHER PARAMETER TUNING OR FEATURE ENGINEERING TO IMPROVE MODEL PERFORMANCE.

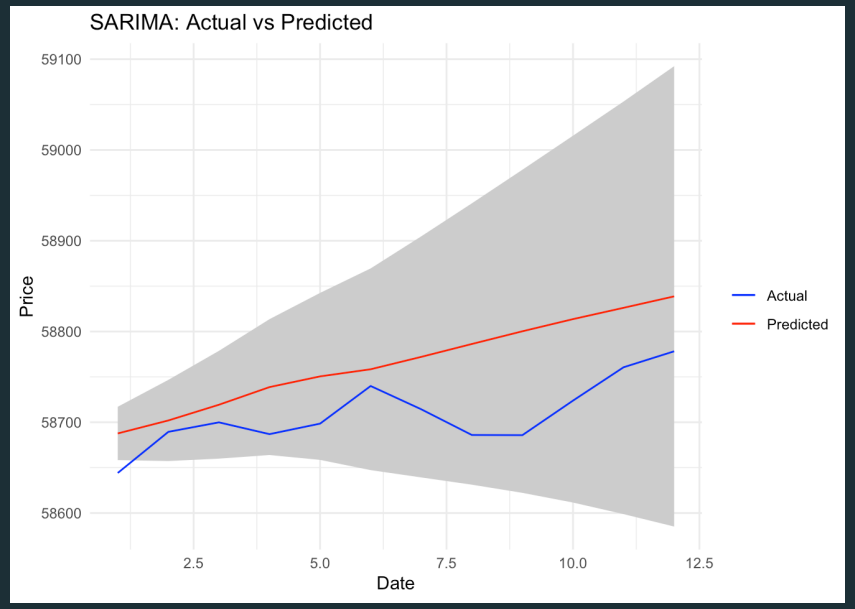
Autoregressive Integrated Moving Average (ARIMA):



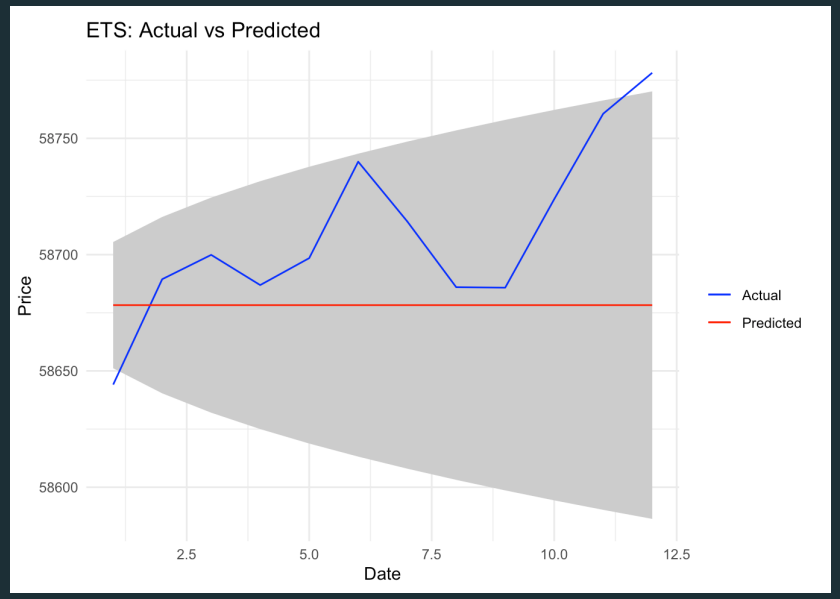
Fractionally Differenced ARIMA (ARFIMA):



Seasonal ARIMA (SARIMA):



Exponential Smoothing State Space Model (ETS):



Future Work

EXPLORING MACHINE LEARNING ALGORITHMS

Investigate ML algorithms (e.g., Neural Net) to identify most accurate model for forecasting Bitcoin returns.

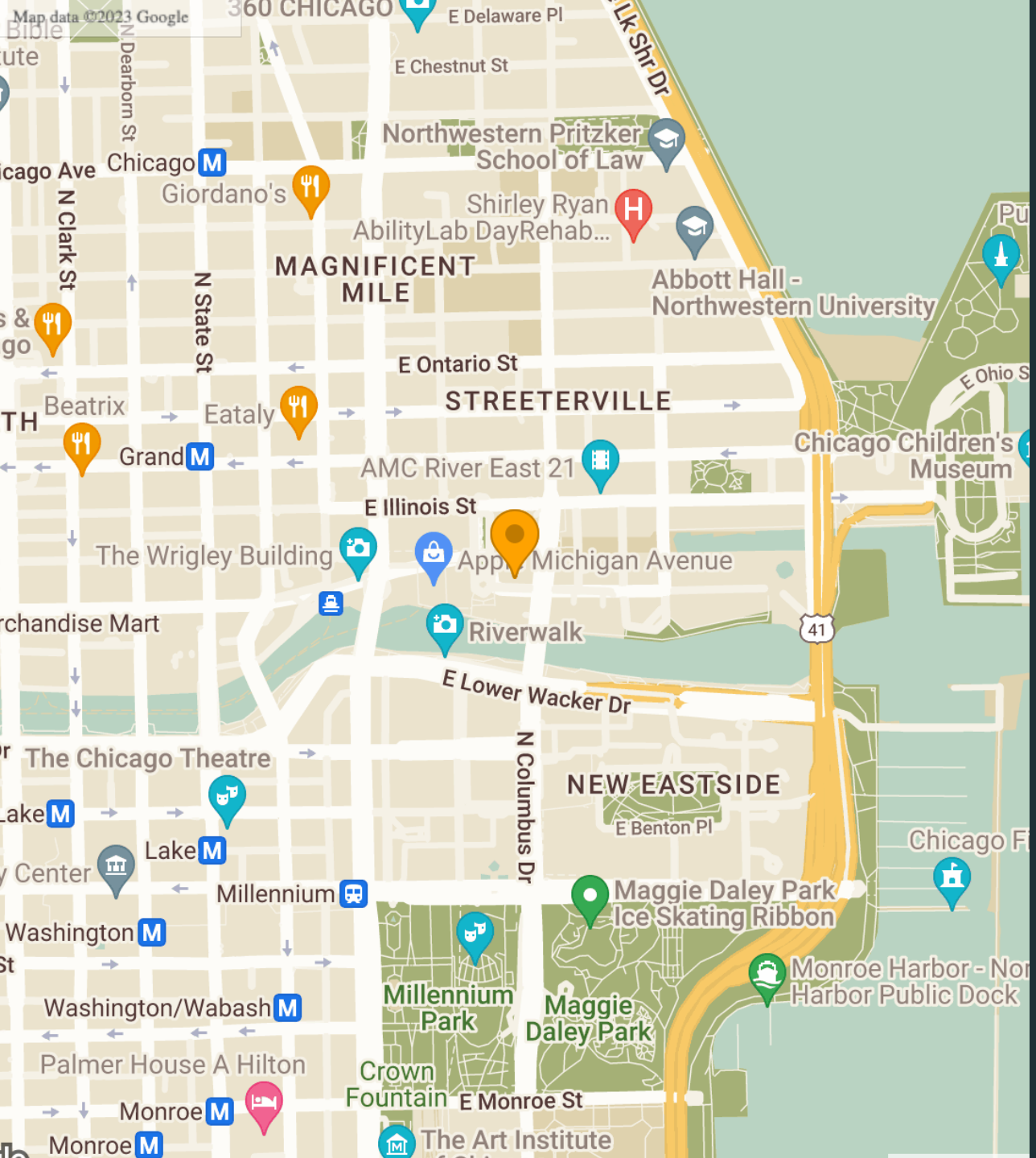
IMPROVING DATA QUALITY

Ensure data accuracy and completeness to improve the accuracy of the model

OPTIMIZING MODEL PARAMETERS

Tune model parameters to maximize the accuracy of the model

BY EXPLORING DIFFERENT MACHINE LEARNING ALGORITHMS, IMPROVING DATA QUALITY, AND OPTIMIZING MODEL PARAMETERS, WE CAN DEVELOP A BETTER ROBUST MODEL FOR FORECASTING BITCOIN RETURNS.



Github Link

https://github.com/hyojun-uchicago/Time-Series-UChicago/blob/main/Hyojun_Kim_TS_Final_Project.Rmd

kimhyojun@uchicago.edu