

# A TIME SERIES PREDICTOR FOR STOCK MARKET PRICES: BITCOIN CRYPTOCURRENCY

Michael J. Van Slyke

# The problem

## Context

The stock market is a volatile entity where private investments done wisely can be for one's profit or foolishly to destitution. It is subject to a multitude of variables:

- Market Forces
- New technologies
- Human Nature, etc...

## Problem Statement

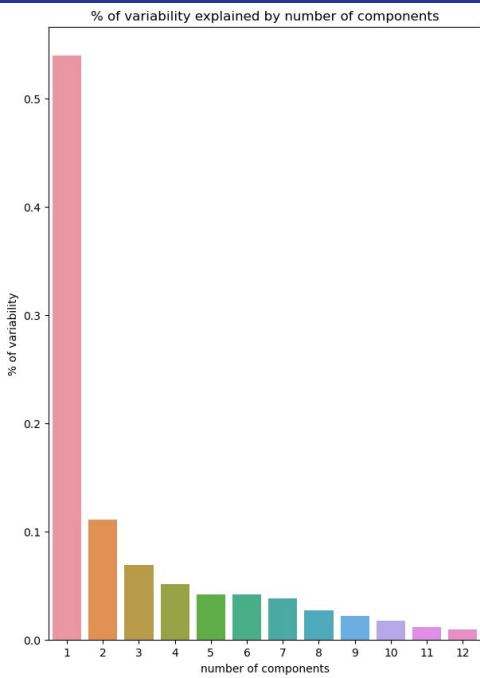
Train a time series model which predicts stock prices with an %85 - %95 accuracy by April 10, 2024 as a piece for a larger model, ultimately directed towards automating ideal trading actions depending on pre-defined investing goals.

## Scope

As a piece of a much larger project in the long term, we will endeavor to achieve the best reasonable accuracy and precision possible with this model, between %85-%95 at least, for proof of concept, in predicting stock market values. Measured most likely using MAE, and RMSE values for the average distance between predicted and true values.

# Data Wrangling

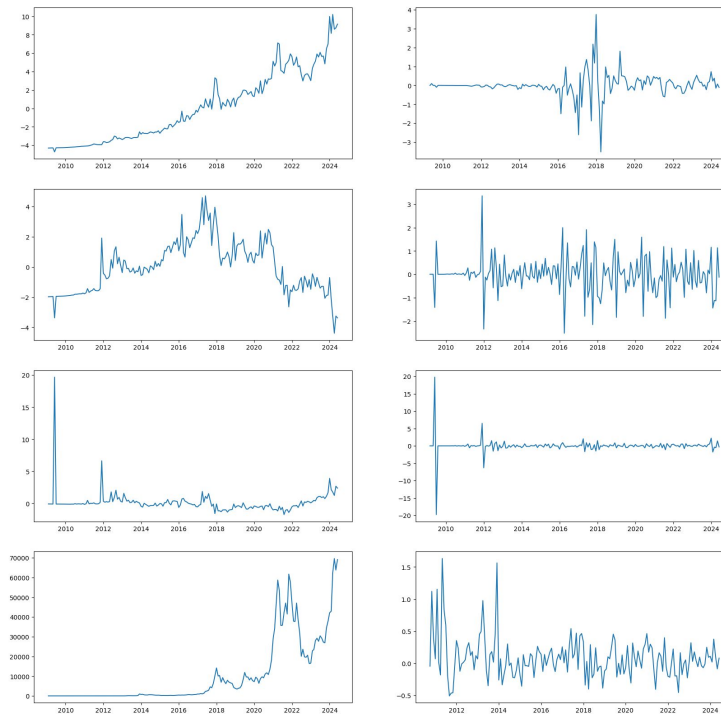
From 32 to 12 dimensions




- Rotated Data Stream from NASDAQ API
- Renamed Codes to Significant names for columns
- Dropped columns with only data from 2010 - 2016
- Examined Correlated Data
- Reduced Dimensionality using PCA.

# Feature Engineering

- Determined via Correlogram that PCAs had substantially less correlations compared to Eigenvectors.
- From Autocorrelation, Partial Autocorrelations and Adfuller tests, it was determined that PCAs 1, 2, 6 and Market Price weren't sufficiently stationary to make forecasts using ARIMA and SARIMAX modeling techniques and so were transformed. (Shown right.)
- Preserved original data for LSTM model since LSTM don't care about stationarity of Time-Series.

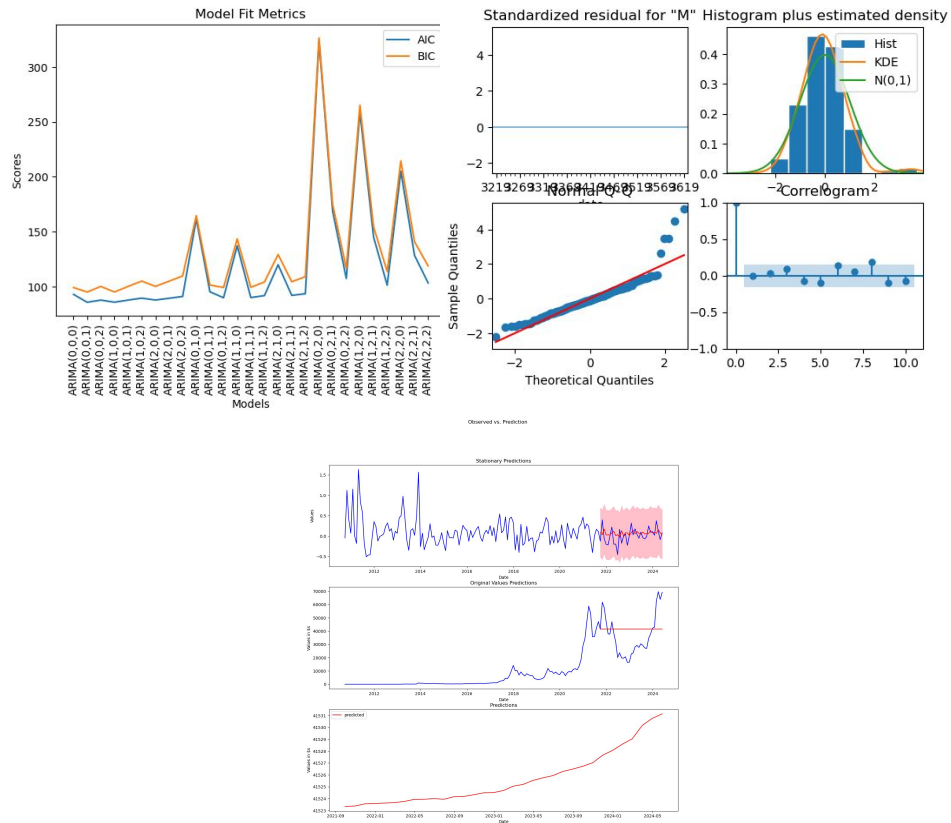


# Model Pre-Processing

- Pre-selected at the start of project to model Time-Series data with ARIMA, SARIMAX, and LSTM
  - Though we assumed we would need exogenous variables (other time-series data than what's being predicted) to make truly accurate predictions we wished to see if a simple univariate Time-Series model would be sufficient.
  - Thus we had to curate 3 unique X, y inputs for our models,
    - ARIMA: X: A single Time-Series 'Market Price' Stationary
    - SARIMAX: X: Exogenous variables PCA1-12 Stationary, y: 'Market Price' Stationary
    - LSTM X: Exogenous variables PCA1-12 non-Stationary, y: 'Market Price' non-Stationary
  - ARIMA and SARIMAX were divided into train and test sets, while LSTM was divided into, train, validation, & test sets.
  - All data was scaled by either the standard scaler, or min-max scaler prior to being submitted to the models.
- 

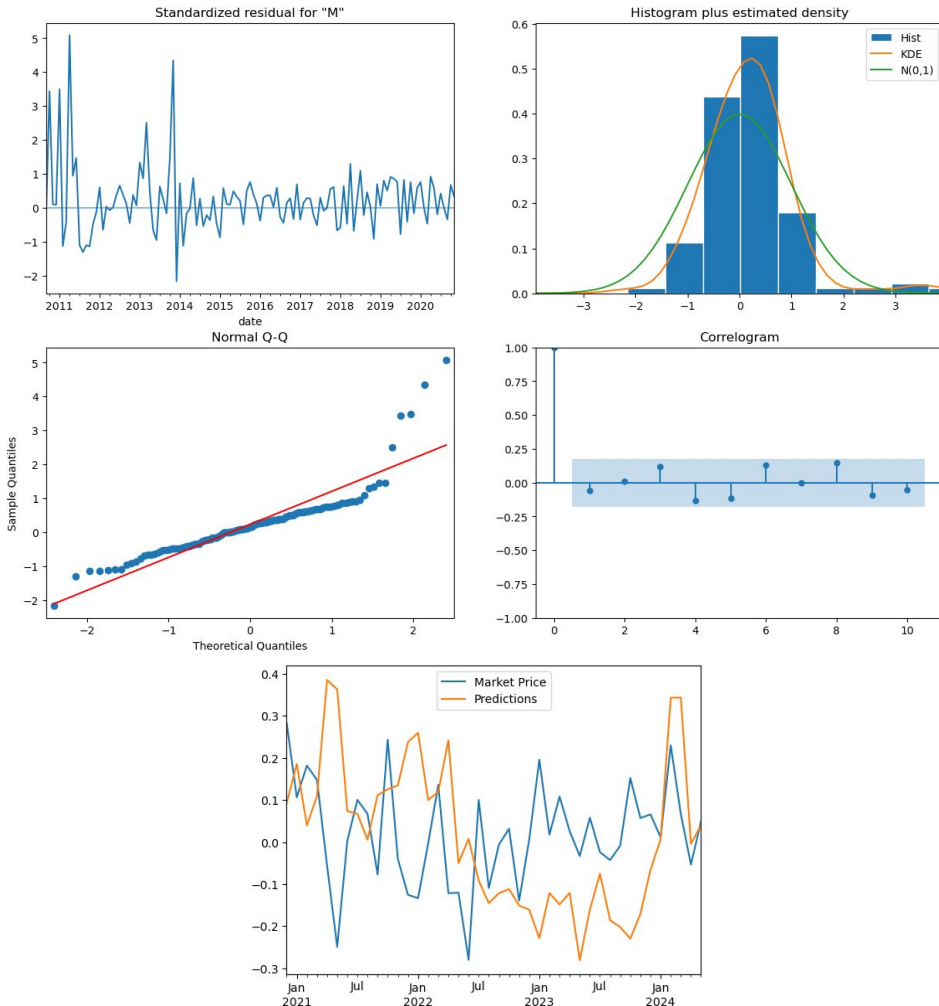
# Modeling: ARIMA

- Both AIC and BIC scores moved in parallel and conveniently the best BIC/AIC score rested with the same model the (0,0,1) in short a Moving Average (MA) model.
- The Metrics appeared decent except for the collection of outliers in the flow of data points
- Stationary data falls within margin of error.
- Transformed to original data appears to be a line but in actuality is an exponential line.
- Not a sufficient model for our purpose as it is effectively only prediction the mean of the previous values.



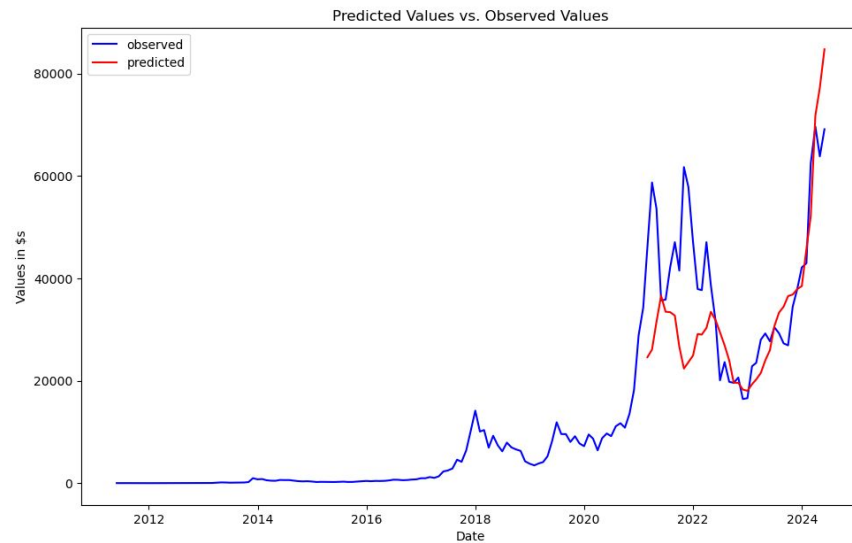
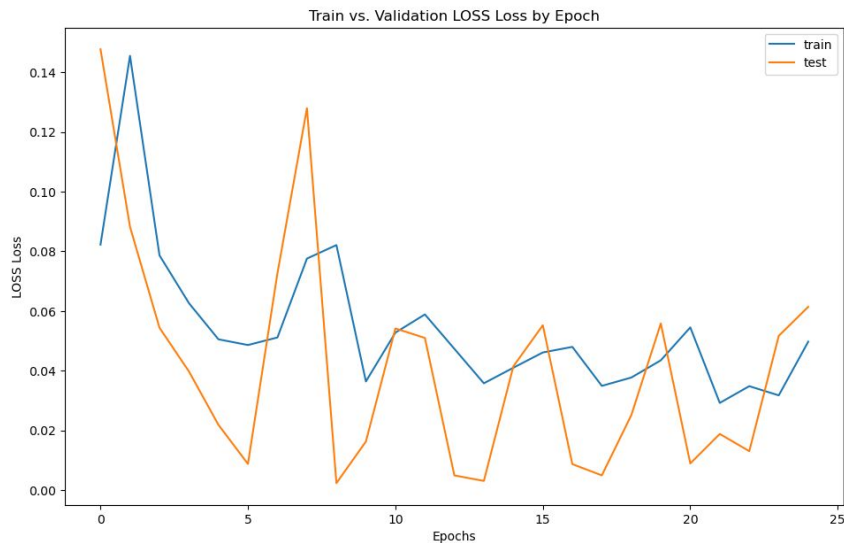
# Modeling: SARIMAX

- Both AIC and BIC scores moved in parallel and conveniently the best BIC/AIC score rested with the same model the (0,0,1) in short a Moving Average with Exogenous Variables (MAX) model.
- The Metrics appeared decent except for once again the collection of outliers.
- Not a sufficient model for our purpose as it is effectively predicting at random with this model.
- Likely the cause of both these models' failure is due to only using the previous data point to predict the next; it lacks a window
- In the last model we added a window and the results were amazing!



# Modeling: LSTM with Windowing

Model36 Performance Visualization



**Final Selected Model:**

**Metrics: MAE: 0.002, RMSE trainset score: 0.045, MAE testset score: 0.008, RMSE testset score: 0.09**

**LSTM: Model layers: num\_layers: 1, num\_nodes: 125, dropout: 0.1, loss: MAE, optimizer: ADAM, Window: 8**



# Solution

LSTM Model 36

0.09 RMSE

0.008 MAE

- There were other more performant models in the +1500 models tested for LSTM on this dataset.
  - However those were substantially heavier and in production where lines of code can cost you in cloud deployments, if you can achieve virtually the same result with less, DO IT.
  - The metrics were substantially better then what we were hoping to achieve with this model.
-

# Insights

The Final graph of the stationary data from the testing data frame clearly indicates that the model is really just guessing. A "Your guess is as good as mine" paradigm for our purposes is absolutely not sufficient. Therefore per this analysis, the best model to follow hands down is the LSTM, specifically LSTM Model 36. With that said however, there are likely better ways to go about doing it or better refinements we can enact on the deep learning hyper parameters. we could consider GRUs or CNNs as potential replacements for the LSTM due to them being lighter and plausibly more accurate still. As it stands, Neither the ARIMA, nor the SARIMAX modeling came even remotely close and the biggest concern with the LSTM is how it began to overestimate the value of Bitcoin, which could result in misguided suggestions to investors or in an investing AI, which is the grand purpose of this group of projects.

## Further Investigations:

I should like to investigate other currencies beyond just crypto to see if they behave similarly and can therefore be generally forecast with a single currency value model. Also, since the market has many interdynamics I will need to investigate other stocks (AMD, Intel, Apple, Microsoft, Google, Tesla, etc) which are close to cryptocurrency in the sense they provide technological services necessary for cryptocurrency minting. Pending these analyses, next steps would include a more general investigation of stocks loosely grouped into business sectors and repeat the process of modeling their data. Alongside these investigations will include the prediction of import indicators when making and considering investments to affect the decision making paradigm of our potential future AI.