Crypto Investment Recommendations

Problem Identification

Cryptocurrency coins are digital assets designed to work as a medium of exchange where ownership records are transactions stored in a blockchain ledger using strong cryptography. Cryptocurrencies typically use decentralized control as opposed to centralized digital currency and central banking systems. Bitcoin was the first decentralized cryptocurrency formed in 2009 and dominates the market with $903.5B market capitalization as of October 2021. According to coinlore.com there are 6235 active coins trading today.[1]

I first learned about cryptocurrencies in 2017 from a former co-worker. If I would have purchased one bitcoin in January 2017 for $1000 it would be worth 36K as of today. Honestly, I did not take it serious at that time and still have skepticism. However, I am bullish on the long-term role of cryptocurrencies in the financial market. The goal of this project is to utilize technical analysis and social media metrics to identify buy, hold and sell opportunities for short-term holding periods. For this project to limit the scope of the analysis, I will be focusing on Bitcoin and Dogecoin.

Data Source

LunarCRUSH was founded in 2018 and is a cryptocurrency focused platform that delivers community insights to investors, funds, and exchanges. "Through resource applications and API's, get real-time insights that help make informed crypto investment decisions."[2] I will be using the assets API endpoint and technical analysis metrics/indicators from the TA-Lib[3] python wrapper.

Data Wrangling

The primary goal of the data wrangling stage was to utilize the LunarCRUSH assets data for the relevant coins and evaluate each metric for cleanliness and usability. The assets endpoint is limited to 720 historical data points, and I will be using a daily interval. After completion of this project, I will archive older data and then merge with the most recent 720 data points to provide a longer history for model training for real-world usage.

Data Cleaning

Based on the timeframe that LunarCRUSH has been in existence and the vast number of metrics available in the asset's endpoint, I had suspicion that some metrics may not be very dependable for my use case. Therefore, I conducted a thorough analysis of missing values specifically when and how many values were missing.

Several columns were no longer being populated so I removed those from further analysis. In addition, multiple columns were missing values at the beginning of the time series. This is consistent with new

---

[1] www.coinlore.com
[2] lunarCRUSH About
[3] TA-Lib Python Library

fields added to the endpoint but not populated retroactively. I did not want to just remove data points for a column that I may not even end up using in the modeling phase, so I decided to keep the blanks in the intermediate data files.

Buy/Hold/Sell Decision

The goal of this project is not to be able to accurately predict a future close price for a coin but to predict the trend and recommend an investment decision. I created a new algorithm to determine buy/hold/sell decisions based on a forward-looking window:

Buy

1. Maximum gain during the window is > the desired threshold
2. Change in close price from the start to end of window is higher (window candle is green)
3. Lowest price during the window is >= the low price for the current day

Sell

1. Maximum loss during the window is < the desired threshold
2. Change in close price from the start to end of window is lower (window candle is red)
3. Highest price during the window is <= the high price for the current day

If either the buy or sell logic is not true, then the recommended decision is **Hold**.

Figures 1 and 2 show the close price and respective BHS decision recommendation. The logic is effective at recommending buying at the bottom of the trough or as the price continues to rise and then recommending selling at the peak or just before the peak.

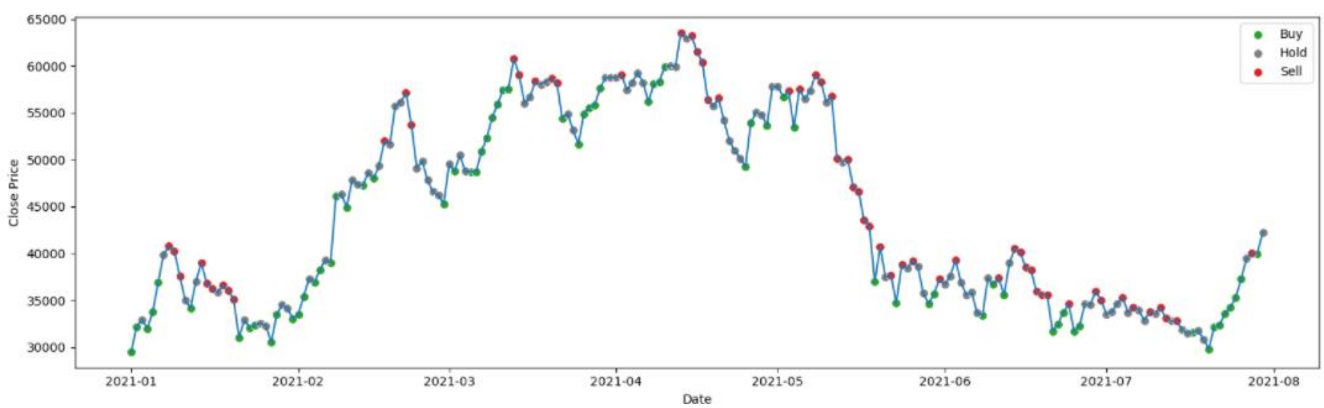


Figure 1 – Bitcoin Buy/Hold/Sell Decision January 2021 – August 2021

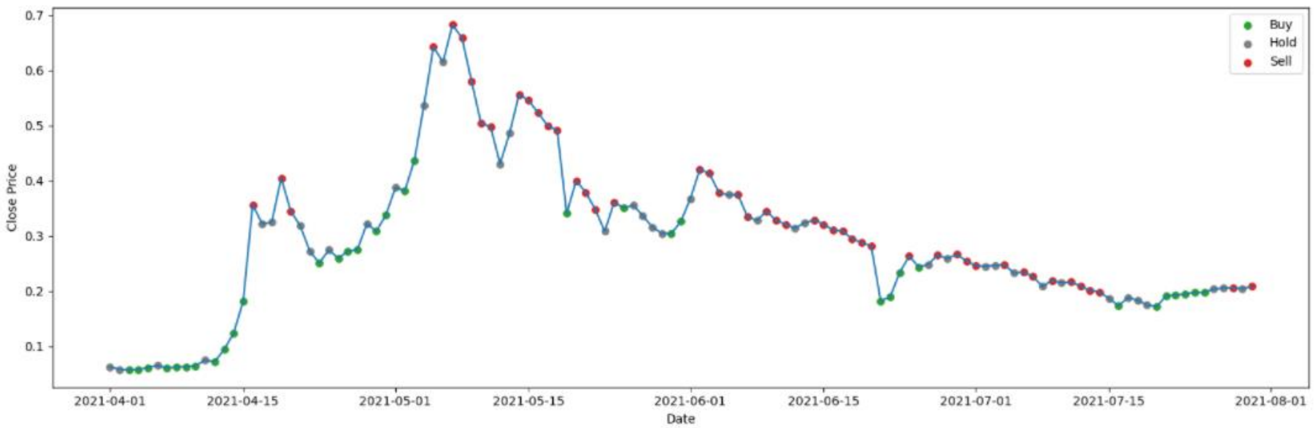


Figure 2 – Dogecoin Buy/Hold/Sell Decision April 2021 – August 2021

Each coin's intermediate dataset consisted of **718** data points for the period 8/20/2019 to 8/6/2021 and **58** features.

Exploratory Data Analysis

Data Profile

In this stage I dropped the blank values due to the rolling 7 day forward window, so this left me with only 711 data points for each coin.

Technical Analysis

Technical analysis is the term used to evaluate investments and identify trading opportunities based on trends and patterns. I will be using the TA-Lib library to supplement the coin price and social media metrics from the LunarCRUSH assets endpoint. In the subsequent sections I will review some key metrics and indicators that I identified during this phase.

Bollinger Bands

Bollinger Bands show the relationship between a simple moving average and two standard deviations above and below.[4] In Figure 3 for Bitcoin you can see that a bull market can drive the price up to the upper band. When the Close price is straddling the upper band there eventually will be a selloff and the price drops. Vice versa when the price rides along the lower band the coin is oversold, and people reenter the market and the price rises.

[4] Bollinger Bands Investopedia

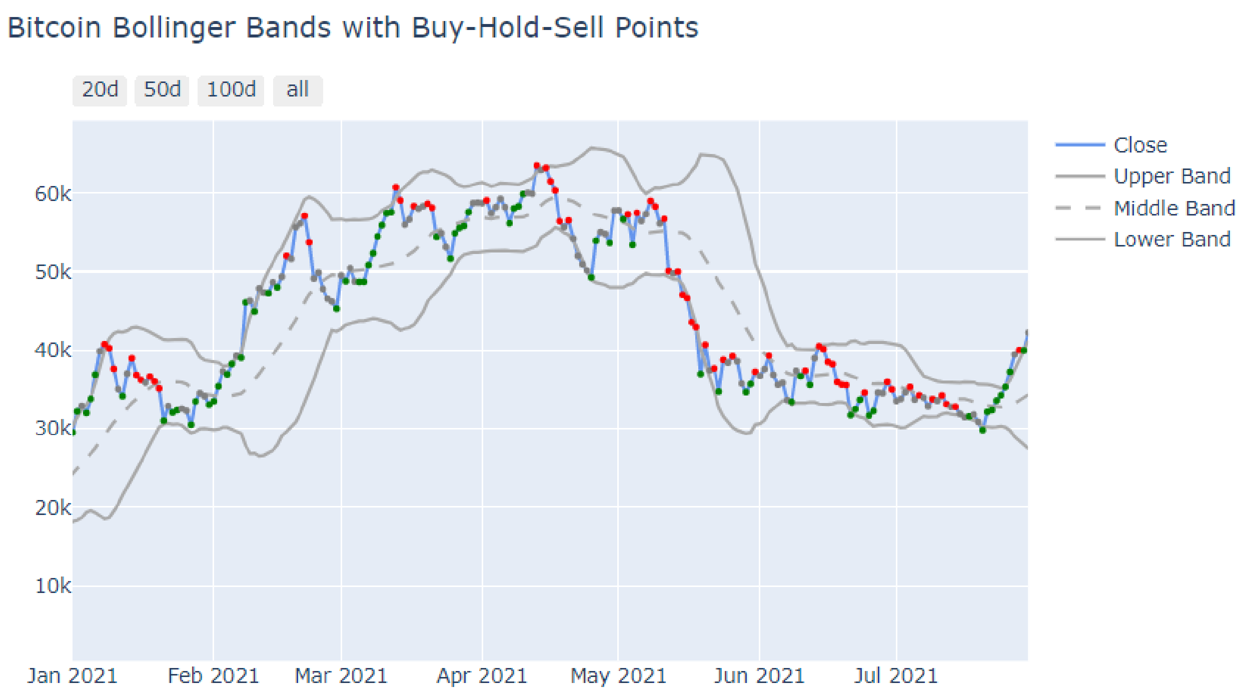


Figure 3 – Bitcoin Bollinger Bands

Moving Averages

Moving averages are one of the most common metrics used in technical analysis and there are a variety of calculations such as simple, weighted and exponential. This method smooths out the volatility of the price and can identify trend direction.[5] In Figure 4 for Dogecoin, the weighted moving average crossing above the simple moving average is a bullish sign. Buy at the support levels when below the weighted moving average and sell at the resistance levels when above the weighted moving average.

[5] Moving Average Investopedia

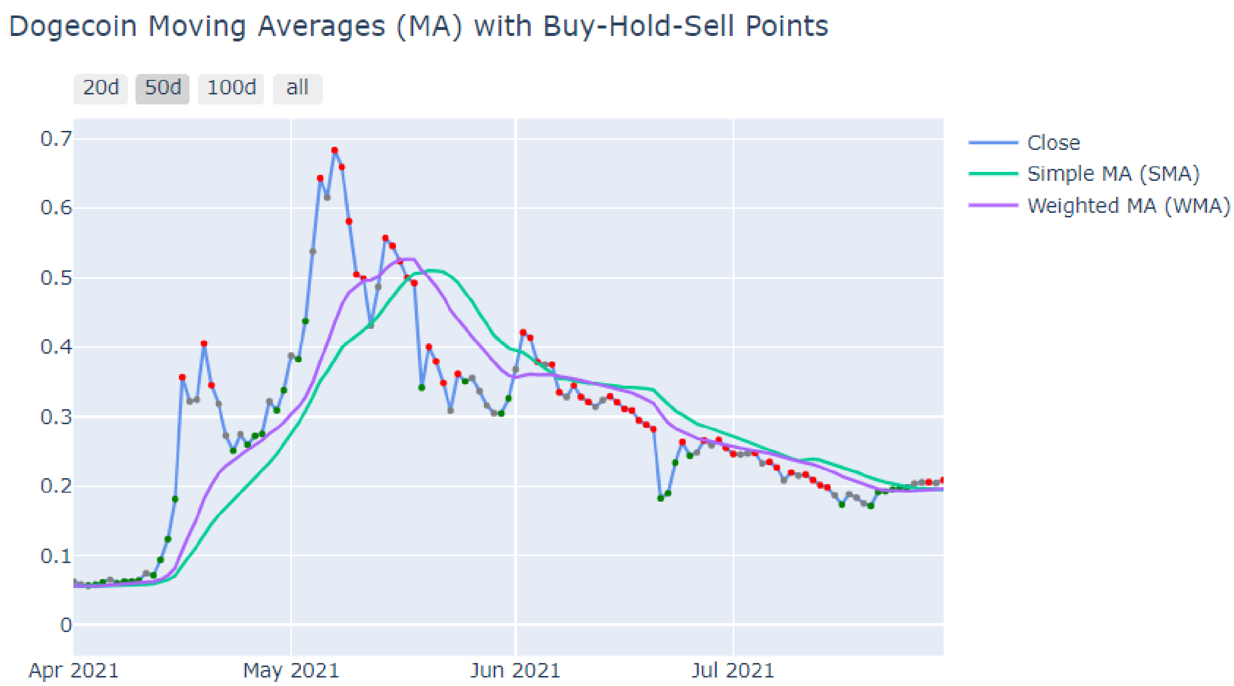


Figure 4 – Dogecoin Simple/Weighted Moving Average

Moving Average Convergence/Divergence (MACD)

Moving average convergence divergence (MACD) is a trend following momentum indicator based on moving averages. MACD is calculated by subtracting the 26-period exponential moving average from the 12-period exponential moving average.[6] The MACD histogram shows when the MACD line is above (green) or below (red) the MACD signal line. In Figure 5 for Bitcoin, the MACD line crosses above the MACD Signal line is a bullish sign with potential buy opportunities. The market will eventually go bearish and drive the MACD back below the signal from a selloff.

[6] MACD Investopedia

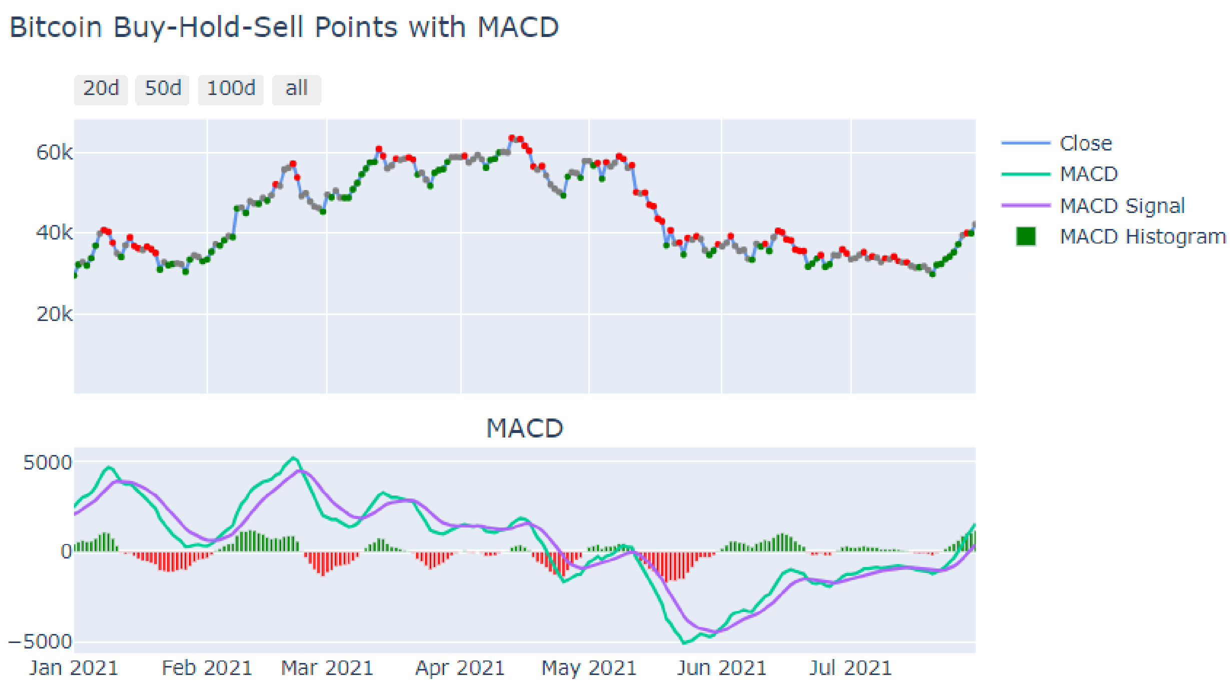


Figure 5 – Bitcoin MACD

Relative Strength Index (RSI)

The relative strength index (RSI) is another momentum indicator that measures the magnitude of recent price changes to identify overbought or oversold conditions of a coin's price. RSI values range between 0 and 100 and above 70 is a sign of overbought and below 30 is a sign of oversold.[7] In Figure 6 for Bitcoin this pattern holds well although there are troughs where the market rises well before the RSI reaches 30.

[7] RSI Investopedia

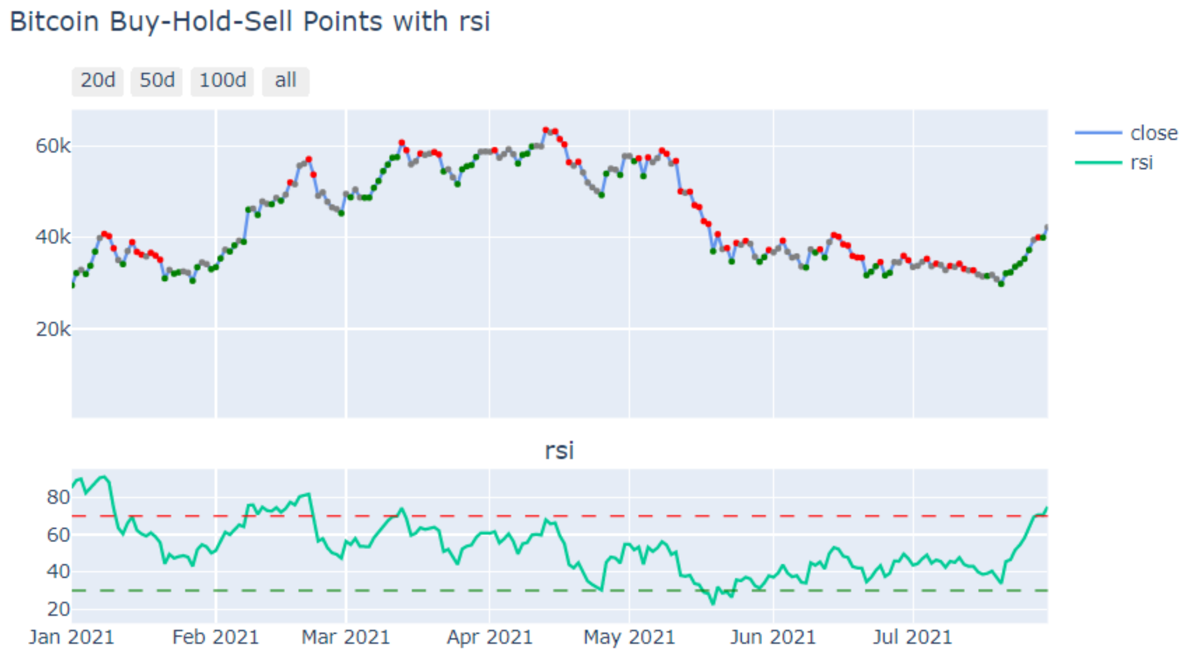


Figure 6 – Bitcoin RSI

Social Analysis

Social influence on cryptocurrency coins was the primary motivation for this project and research in this space landed me on the LunarCRUSH product. Twitter data was my primary interest but building out an entire data feed for tweets really was not feasible for the timeline of the project. LunarCRUSH is a paid product but based on the wide variety of features it offers it made sense for the timeline.

The assets endpoint has 62 columns including open, high, low, and close (OHLC) price data for over 2817 coins, tweet metrics, overall social market activity metrics and other proprietary score metrics.

Twitter

Twitter data dominates the assets endpoint with 19 tweet specific related metrics. Figure 7 summarizes all these metrics.

Column	Description
tweets	Number of tweets collected
tweets_spam	Number of tweets classified as spam
tweets_followers	Sum of follower count for every tweet collected
tweet_quotes	Sum of the number of times all collected tweets were quoted
tweet_retweets	Sum of the number of times all collected tweets were retweeted
tweet_replies	Sum of the number of times all collected tweet reply counts
tweet_favorites	Sum of the number of times all collected tweet likes
tweet_sentiment1	Sum of tweets classified as sentiment 1 (Very Bearish)
tweet_sentiment2	Sum of tweets classified as sentiment 2 (Bearish)
tweet_sentiment3	Sum of tweets classified as sentiment 3 (Neutral)

tweet_sentiment4	Sum of tweets classified as sentiment 4 (Bullish)
tweet_sentiment5	Sum of tweets classified as sentiment 5 (Very Bullish)
tweet_sentiment_impact1	Sum of social score (engagement) of all tweets classified as sentiment 1 (Very Bearish)
tweet_sentiment_impact2	Sum of social score (engagement) of all tweets classified as sentiment 2 (Bearish)
tweet_sentiment_impact3	Sum of social score (engagement) of all tweets classified as sentiment 3 (Neutral)
tweet_sentiment_impact4	Sum of social score (engagement) of all tweets classified as sentiment 4 (Bullish)
tweet_sentiment_impact5	Sum of social score (engagement) of all tweets classified as sentiment 5 (Very Bullish)
sentiment_absolute	Percent of bullish or very bullish tweets
sentiment_relative	Percent tweets that are bullish (excluding neutral in the count)

Figure 7 – Twitter Metrics

Dogecoin was a penny coin prior to 2021 and exploded all the way to over 60 cents in May 2021 due to social influence. In Figure 8 for Dogecoin you can see large spikes in tweets coinciding with large price increases. However, the tweet data does not appear to be leading indicator but follows the large price swings.

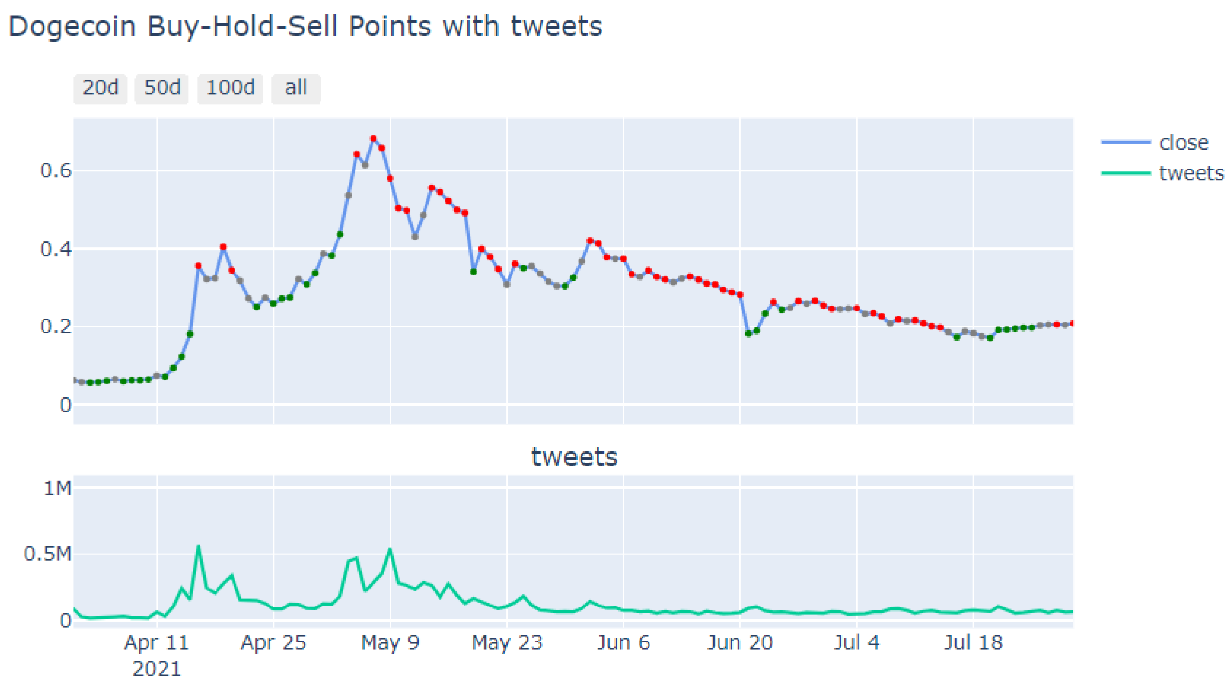


Figure 8 – Dogecoin Tweets

I transformed all tweet metrics using rolling mean and standard deviation to smooth out the volatility to identify patterns in the data. In Figure 9 for Bitcoin, I explored the bearish sentiment impact on price using a rolling standard deviation transformation. As the social market turns bearish with higher standard deviation values this coincides with sell opportunities.

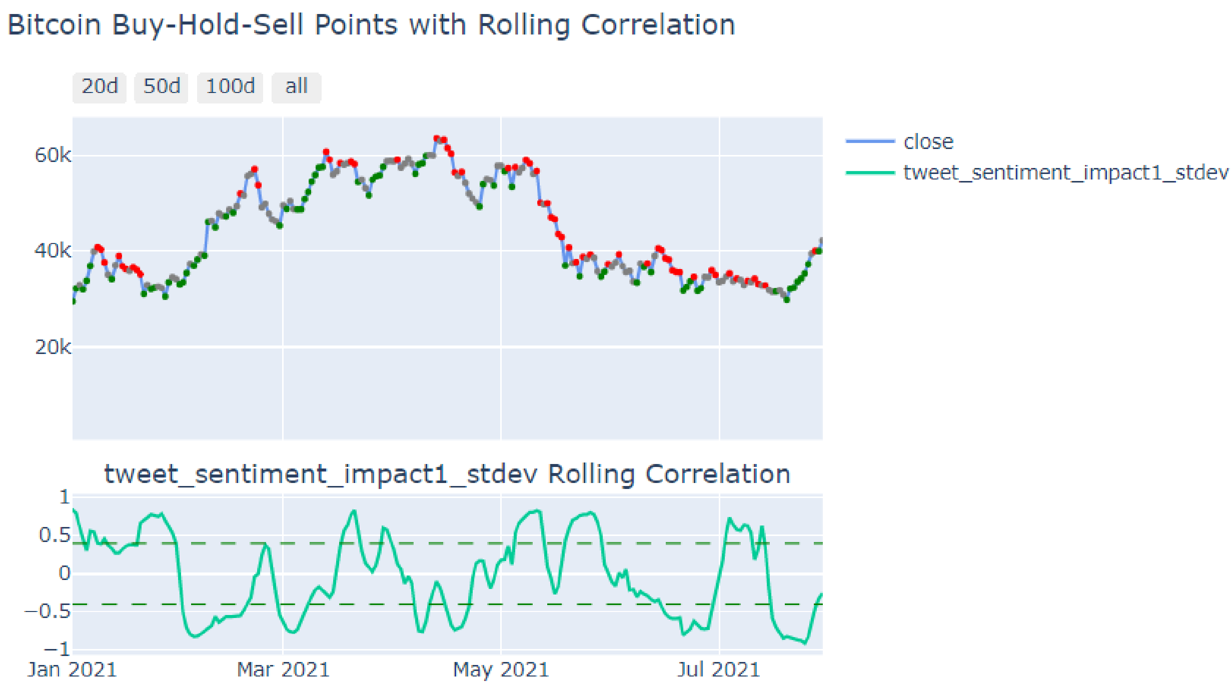


Figure 9 – Bitcoin Tweet Sentiment Impact 1 Rolling Standard Deviation

Other Metrics

The assets endpoint also includes a variety of other social aggregation metrics, rank values and proprietary scores. Figure 10 summarizes all these metrics.

Column	Description
market_cap	Total available supply multiplied by the current price in USD
url_shares	Number of urls shared and collected on social
unique_url_shares	Number of unique url shares posted and collected on social
social_score	Sum of followers, retweets, likes, reddit karma etc. of social posts collected
average_sentiment	Average sentiment of collected social posts
news	Number of news articles published
price_score	A proprietary score based mostly on the change in MACD over time
social_impact_score	A proprietary score based on the relative trend of social_score
correlation_rank	A score based on how the assets social metrics correlate with price and volume
galaxy_score	A proprietary score based on technical indicators of price, average social sentiment, relative social activity, and a factor of how closely social indicators correlate with price and volume
volatility	Degree of variation of a trading price series over time as measured by the standard deviation of logarithmic returns

alt_rank	A proprietary score based on how an asset is performing relative to all other assets supported
alt_rank_30d	AltRank™ but using 30-day metrics instead of 24-hour metrics
market_cap_rank	Position/rank of the asset relative to all other supported assets, lower is better
percent_change_24h_rank	Position/rank of the asset's percent change in 24 hours, lower is better (positive percent change)
volume_24h_rank	Position/rank of the assets 24-hour volume in USD relative to all other supported assets, lower is more volume
social_volume_24h_rank	Position/rank of the assets 24-hour social volume relative to all other supported assets, lower is most volume
social_score_24h_rank	Position/rank of the assets 24-hour social score relative to all other supported assets, lower is best/highest social score
social_contributors	The number of unique accounts posting on social
social_volume	Number of social posts
market_cap_global	Total available supply of all coins multiplied by the current price in USD
market_dominance	Market cap dominance value specific to the coin.

Figure 10 – Other Metrics

In Figure 11 for Dogecoin, I explored the Pearson correlation coefficient values globally across the entire time series. Volume, social_score, social_contributors, social_volume and market_dominance have a strong positive correlation with close price.

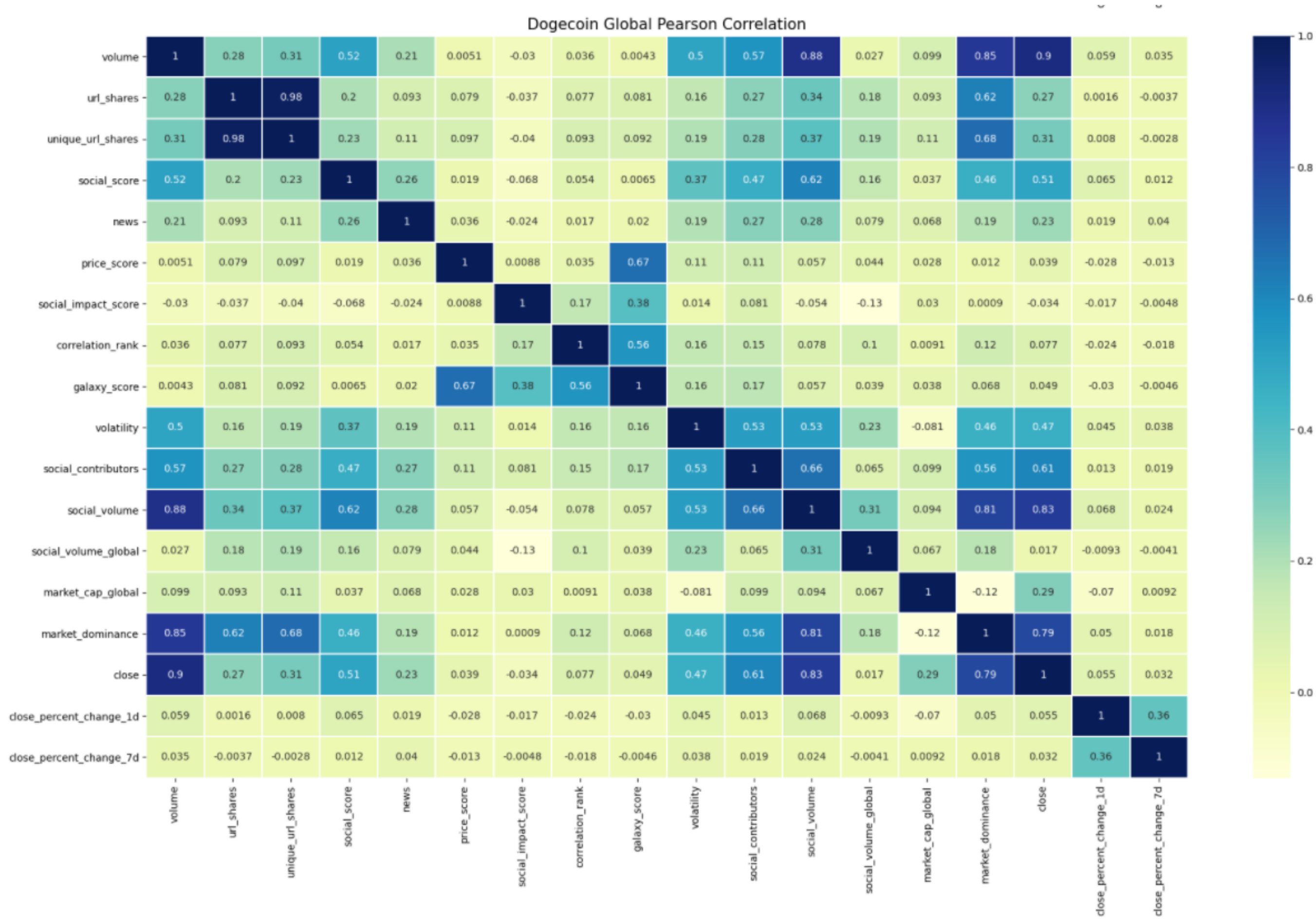


Figure 11 – Dogecoin Other Metrics Pearson Correlation Matrix

Social_volume had the strongest relationship with close price for Dogecoin and spikes in global social media activity coincides very closely with Dogecoin price increases as shown in Figure 12.

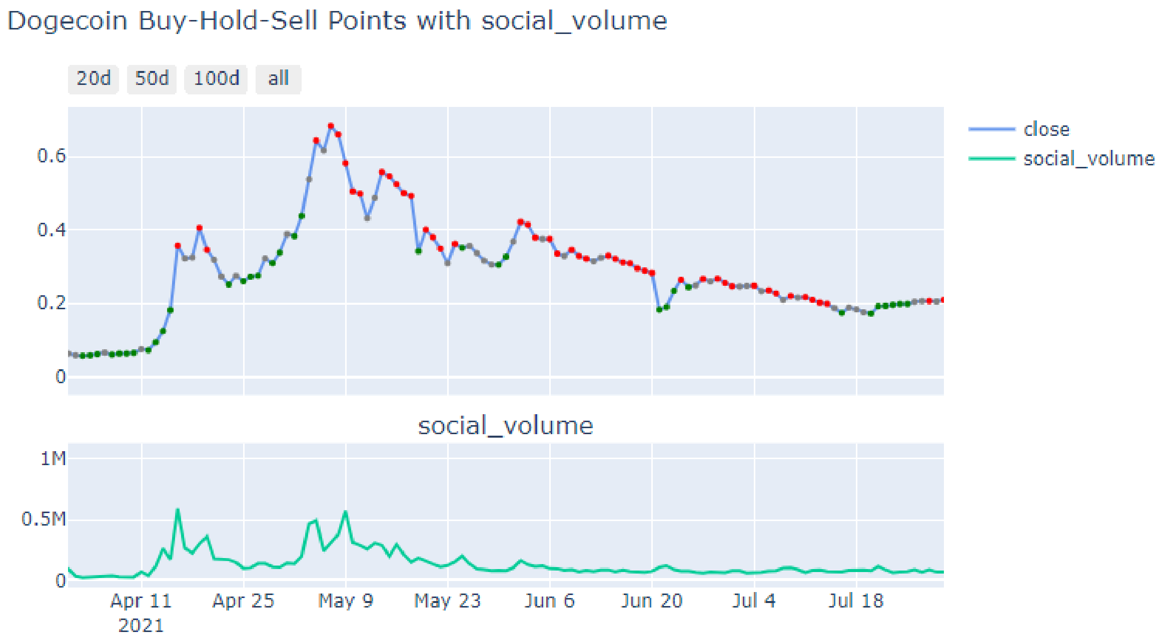


Figure 12 – Dogecoin Social Volume

Preprocessing and Modeling

Phase 1 of this project will focus on time series predictions using a LSTM recurrent neural network (RNN) architecture. The code based on the modeling phase was influenced by the repository and [blog post](#) by Jakob Aungiers.[8] However, I have refactored several aspects of the code to meet my specific requirements such as evaluating multiple preprocessing techniques.

Data Transformation

The first step in the data transformation process was to add all the single and multi-period technical analysis metrics. In addition, based on the exploratory data analysis phase, I added in custom transformations for rolling mean and standard deviation.

The next step was to normalize the features to ensure convergence during model training. In the Jakob's project he implemented a custom % Change method as shown in Figure 13.

n = normalized list [window] of price changes
p = raw list [window] of adjusted daily return prices

Normalization: $n_i = \left(\frac{p_i}{p_0}\right) - 1$

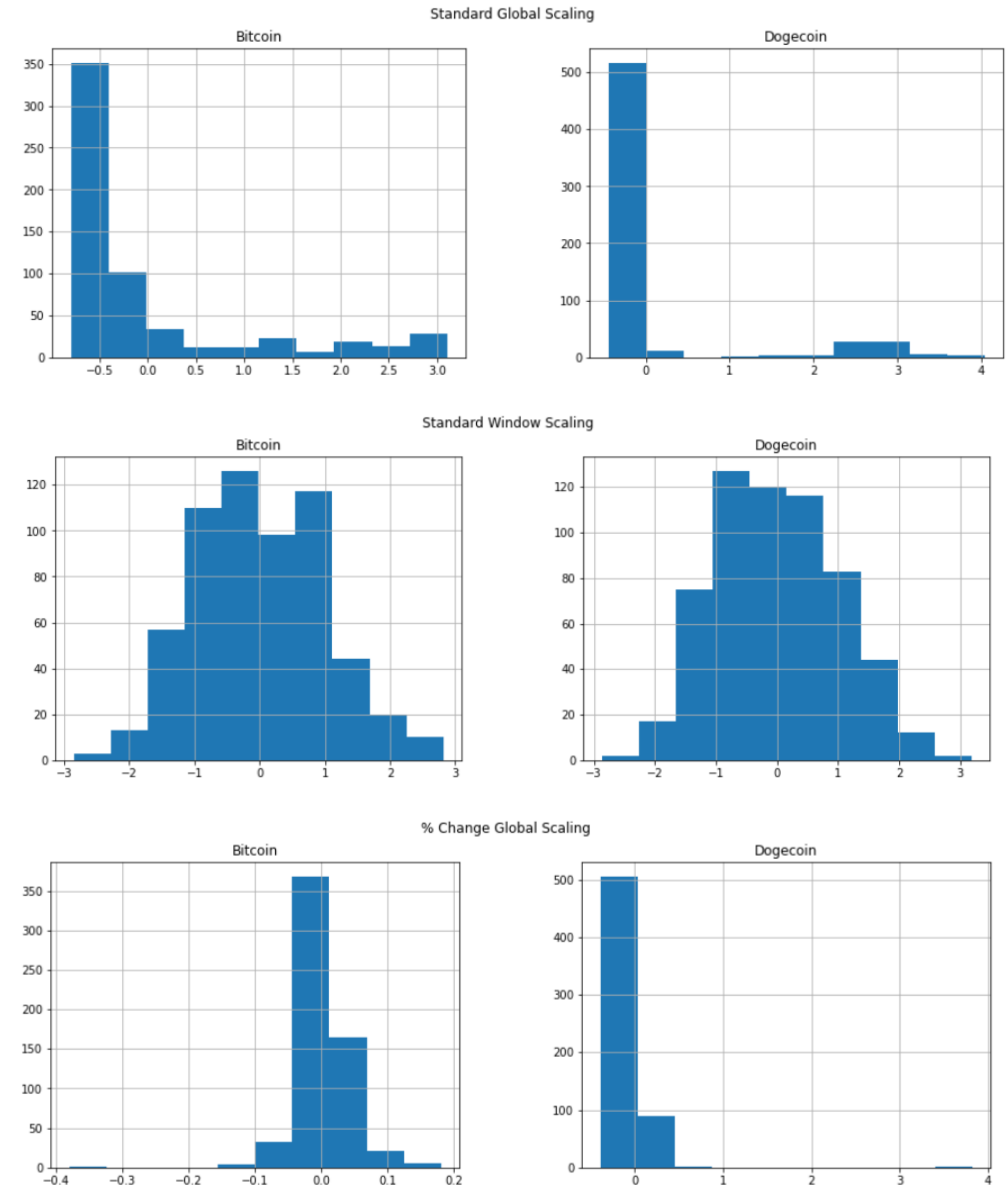
Figure 13 – % Change Window Normalization

[8] Time Series Prediction Using LSTM Deep Neural Networks

I did not want to rely solely on this method and explored multiple options including:

1. Min-Max Normalization (Global)
2. Standardization by removing mean and scaling to unit variance (Global and Window)
3. % Change normalization (Global and Window)

As shown in Figure 14, Standard window scaling normalizes the data the best for both coins. % Change window scaling normalizes the data very well for Bitcoin but since Dogecoin did not really have substantial changes in price until 2021 this scaling method is not as effective for Dogecoin. In the modeling phase I will consider only the window scaling methods.



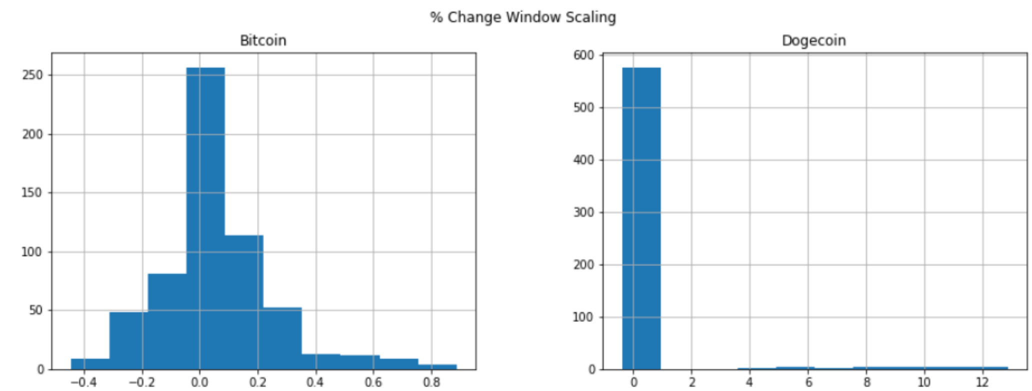


Figure 14 – Normalized Data Histograms

Modeling

In the modeling phase I am focusing on the ending daily close price for each coin. I am not as concerned with being able to accurately predict the future close price but to be able to identify the directional trend. In the first stage I will create one-dimensional LSTM models for Bitcoin and Dogecoin to determine the most effective normalization method.

The 4 evaluation metrics I will be using are:

1. mean directional accuracy - MDA utilizes the change in value (positive or negative) and averages if the predictions had the same directional change. Higher is better.
2. mean absolute error - MAE averages the absolute value of the prediction error. I will use this as the loss metric in the model. Lower is better.
3. mean absolute scaled error - MASE uses MAE but scales the result by dividing by the MAE of actual data and a shifted forward naive forecast. This metric will be scale invariant and allow me to compare the normalization methods across models. Lower is better.
4. accuracy - This will be a sequence accuracy metric using a custom evaluation metric based on directional trend analysis. The objective is to be able to predict the correct directional change while also being able to identify large % increase or decreases.

Baseline Model

Min-Max scaling globally across the time series is poor. As shown in Figure 15, the predictions are well off the mark and the trend accuracy is only 30%.

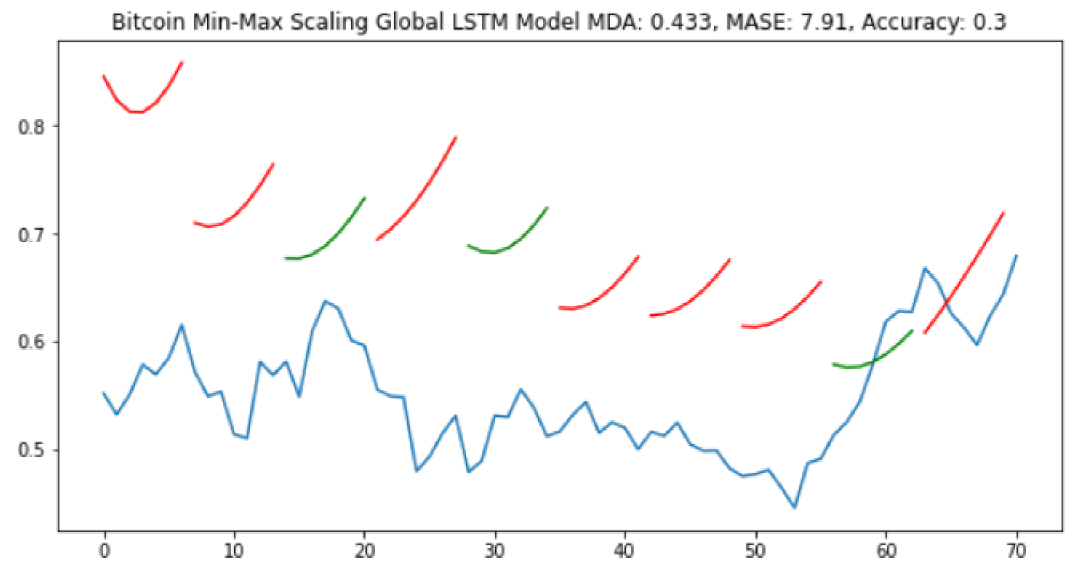


Figure 15 – Bitcoin Min-Max Scaling Global

Standard scaling globally is quite like min max although MASE improves since the predictions are closely to actual close prices.

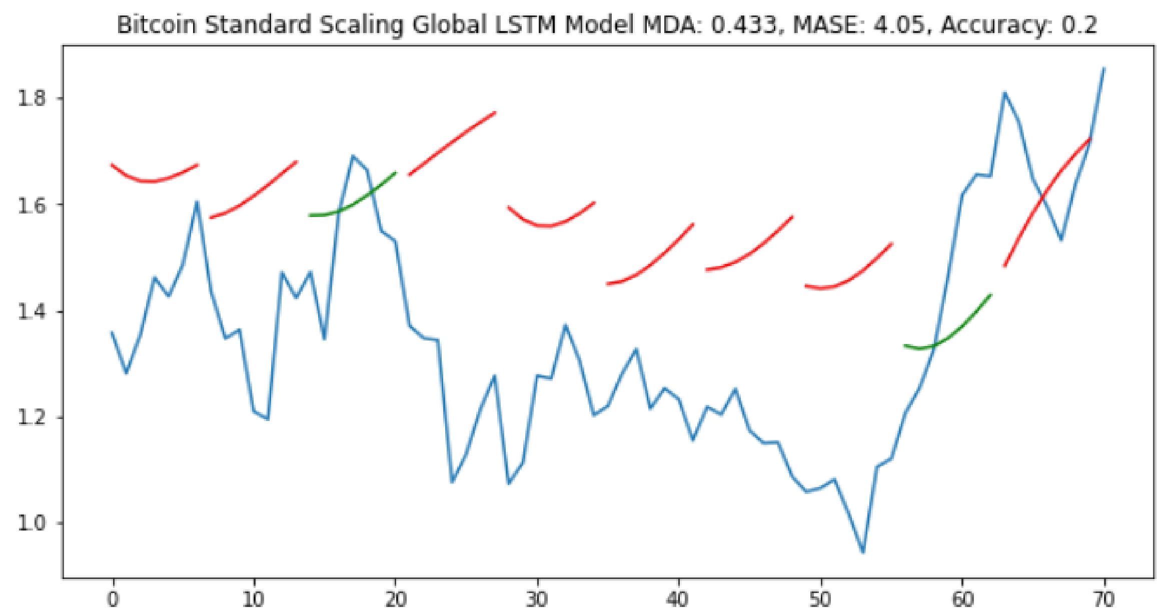


Figure 16 – Bitcoin Standard Scaling Global

Standard scaling across a window performs much better than the prior two methods. The predictions are more accurate than prior scaling methods and this method had the highest sequence trend accuracy so far.

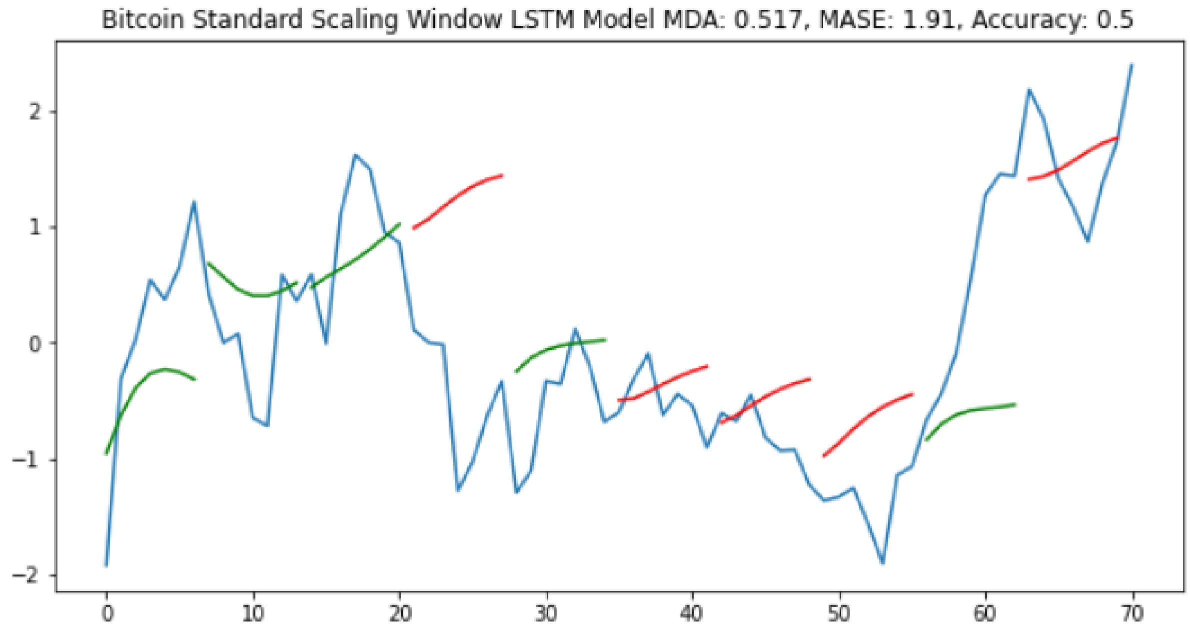


Figure 17 – Bitcoin Standard Scaling Window

% Change globally is clearly the worst performing. The sequence predictions are completely flat. This also highlights the weakness with all three metrics and acts like a naive baseline model to just predict there will be no change in price.

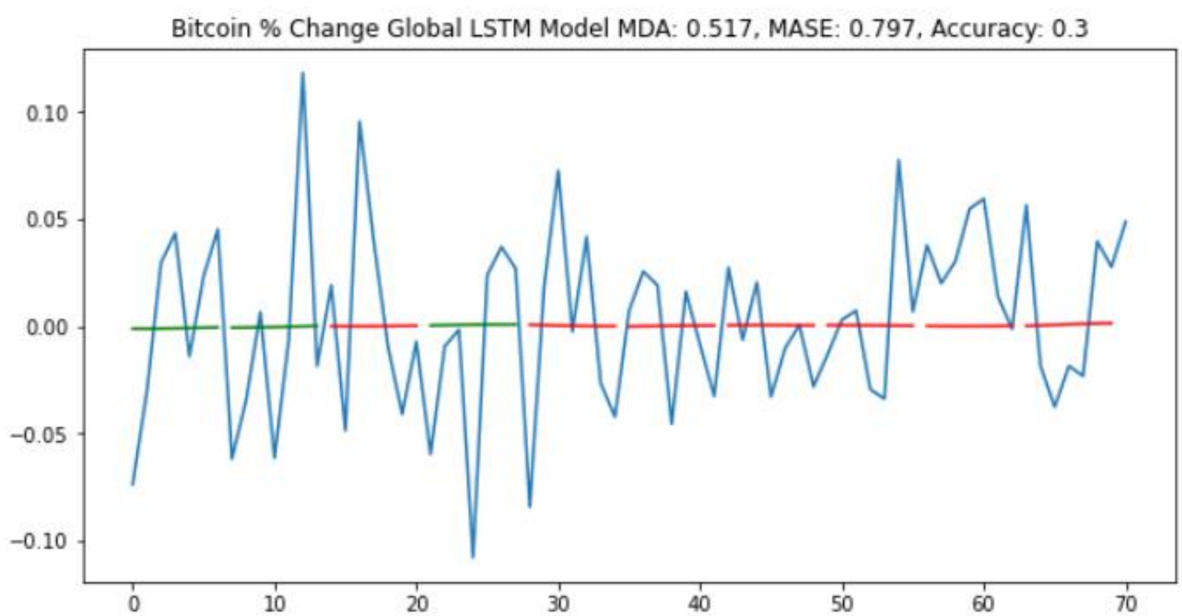


Figure 18 – Bitcoin % Change Global

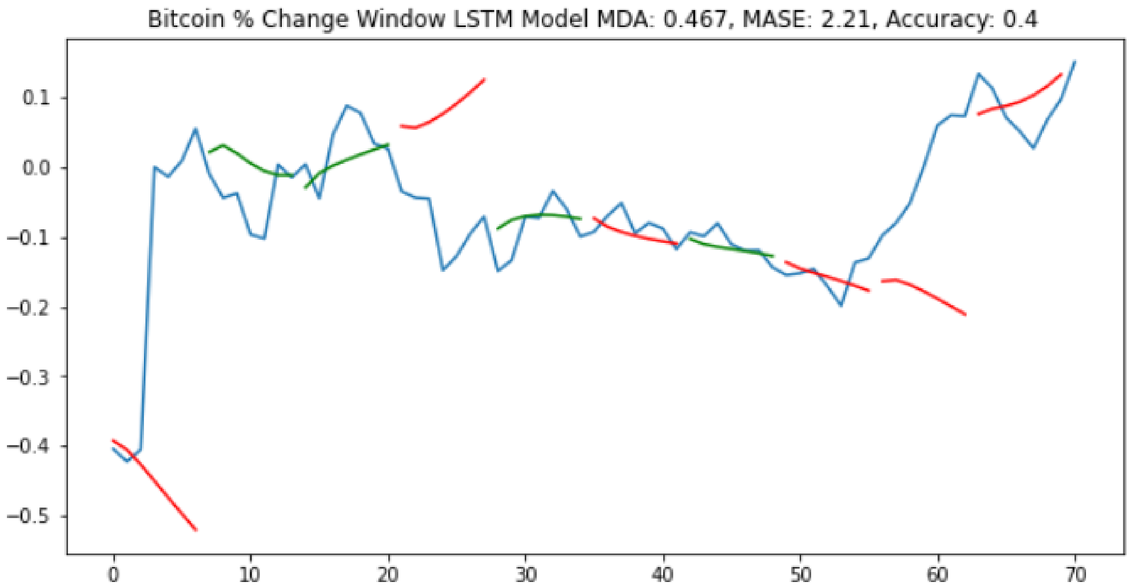


Figure 19 – Bitcoin % Change Window

Like standard scaling over a window, % change also performs quite well. Due to stochastic nature of RNN modeling training, the results can vary for each so in the next phase I will train multiple models for Standard and % Change window scaling and then average the evaluation metrics.

Coin	Method	MDA	MASE	Accuracy
Bitcoin	Standard Scaling Window	0.537	1.94	0.48
Bitcoin	% Change Scaling Window	0.537	2.09	0.46
Dogecoin	Standard Scaling Window	0.53	3.97	0.59
Dogecoin	% Change Scaling Window	0.502	8.66	0.59

Figure 20 – Baseline Metrics

Standard scaling over a window performs the better across all metrics for both coins. This also establishes a baseline to determine if we can improve the model using a multi-dimensional model.

Model Optimization

I implemented an exhaustive search process to identify additional features for each model. The first step was to incrementally add a feature to the one-dimensional model to determine if improved accuracy on one training run. I then narrowed down the features based on the initial results and conducted multiple training runs to identify the features that consistently improved the model from the baseline. Figure 21 for Bitcoin shows the results for all training runs.

```
LSTM model trained with "linreg". 5 training run(s) trend accuracy of 0.58.
LSTM model trained with "dx". 5 training run(s) trend accuracy of 0.52.
LSTM model trained with "market_cap_global". 5 training run(s) trend accuracy of 0.52.
LSTM model trained with "market_cap_global_pct_change". 5 training run(s) trend accuracy of 0.5.
LSTM model trained with "market_dominance". 5 training run(s) trend accuracy of 0.44.
LSTM model trained with "market_dominance_stdev". 5 training run(s) trend accuracy of 0.58.
LSTM model trained with "percent_change_24h_rank". 5 training run(s) trend accuracy of 0.64.
LSTM model trained with "social_volume_24h_rank". 5 training run(s) trend accuracy of 0.54.
LSTM model trained with "volume_24h_rank". 5 training run(s) trend accuracy of 0.56.
LSTM model trained with "social_score_24h_rank". 5 training run(s) trend accuracy of 0.5.
LSTM model trained with "social_volume_global". 5 training run(s) trend accuracy of 0.58.
LSTM model trained with "social_contributors". 5 training run(s) trend accuracy of 0.48.
LSTM model trained with "news_mean". 5 training run(s) trend accuracy of 0.44.
LSTM model trained with "volatility_mean". 5 training run(s) trend accuracy of 0.56.
LSTM model trained with "tweets_mean". 5 training run(s) trend accuracy of 0.42.
LSTM model trained with "tweet_sentiment_impact_bearish". 5 training run(s) trend accuracy of 0.52.
LSTM model trained with "tweet_sentiment_impact_bullish". 5 training run(s) trend accuracy of 0.42.
LSTM model trained with "tweet_sentiment_net". 5 training run(s) trend accuracy of 0.44.
LSTM model trained with "tweet_sentiment_bearish". 5 training run(s) trend accuracy of 0.5.
LSTM model trained with "tweet_sentiment_bullish". 5 training run(s) trend accuracy of 0.56.
```

Figure 21 – Bitcoin Multi-Dimensional Training Run Accuracy

This was a subjective process but the final model for Bitcoin included the features:

1. close
2. linreg
3. percent_change_24h_rank
4. social_volume_24h_rank
5. tweet_sentiment_bullish

I was able to improve the baseline model accuracy to 70%.

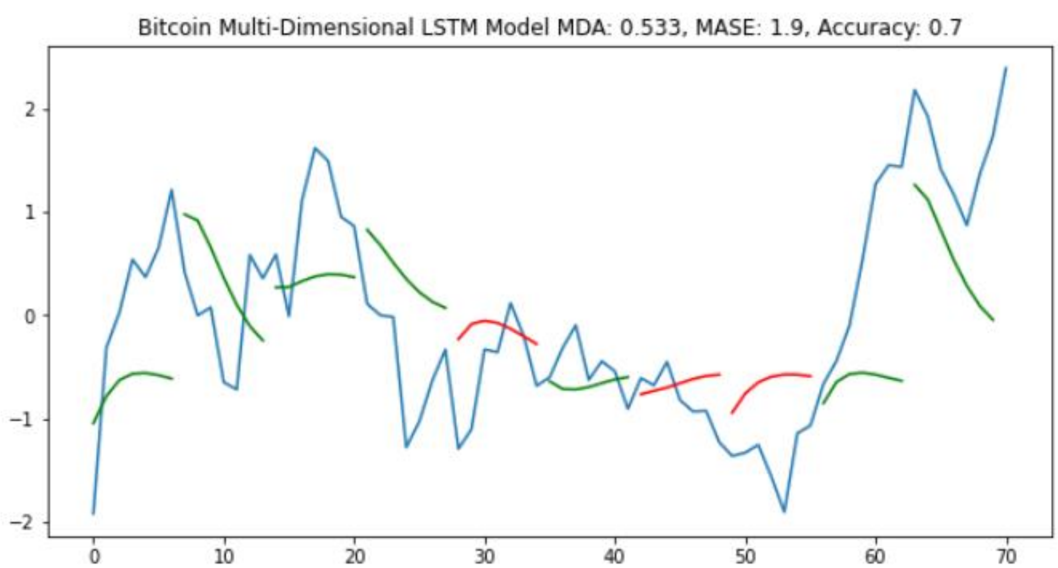


Figure 22 – Bitcoin Final Multi-Dimensional Model Predictions

Figure 23 shows the multiple training runs for Dogecoin.

```
LSTM model trained with "tsf". 5 training run(s) trend accuracy of 0.62.
LSTM model trained with "linreg". 5 training run(s) trend accuracy of 0.72.
LSTM model trained with "roc". 5 training run(s) trend accuracy of 0.64.
LSTM model trained with "cci". 5 training run(s) trend accuracy of 0.6.
LSTM model trained with "macd". 5 training run(s) trend accuracy of 0.68.
LSTM model trained with "market_cap_global". 5 training run(s) trend accuracy of 0.7.
LSTM model trained with "market_cap_global_pct_change". 5 training run(s) trend accuracy of 0.74.
LSTM model trained with "social_impact_score". 5 training run(s) trend accuracy of 0.64.
LSTM model trained with "price_score". 5 training run(s) trend accuracy of 0.7.
LSTM model trained with "social_volume_global". 5 training run(s) trend accuracy of 0.68.
LSTM model trained with "social_volume_global_mean". 5 training run(s) trend accuracy of 0.76.
LSTM model trained with "url_shares". 5 training run(s) trend accuracy of 0.76.
LSTM model trained with "galaxy_score". 5 training run(s) trend accuracy of 0.68.
LSTM model trained with "tweet_sentiment_bearish". 5 training run(s) trend accuracy of 0.66.
LSTM model trained with "tweet_sentiment_bullish". 5 training run(s) trend accuracy of 0.7.
LSTM model trained with "tweet_sentiment_impact3". 5 training run(s) trend accuracy of 0.76.
LSTM model trained with "tweet_sentiment_impact_bullish". 5 training run(s) trend accuracy of 0.66.
LSTM model trained with "tweet_sentiment_impact_net". 5 training run(s) trend accuracy of 0.74.
LSTM model trained with "tweet_sentiment_impact_net_mean". 5 training run(s) trend accuracy of 0.74.
LSTM model trained with "tweet_sentiment_impact_bearish_stdev". 5 training run(s) trend accuracy of 0.66.
LSTM model trained with "average_sentiment". 5 training run(s) trend accuracy of 0.56.
```

Figure 23 – Dogecoin Multi-Dimensional Training Run Accuracy

The final model for Dogecoin included the following features

1. close
2. linreg
3. market_cap_global_pct_change
4. social_volume_global_mean
5. tweet_sentiment_impact_net_mean

I was able to improve the baseline model accuracy to 90%.

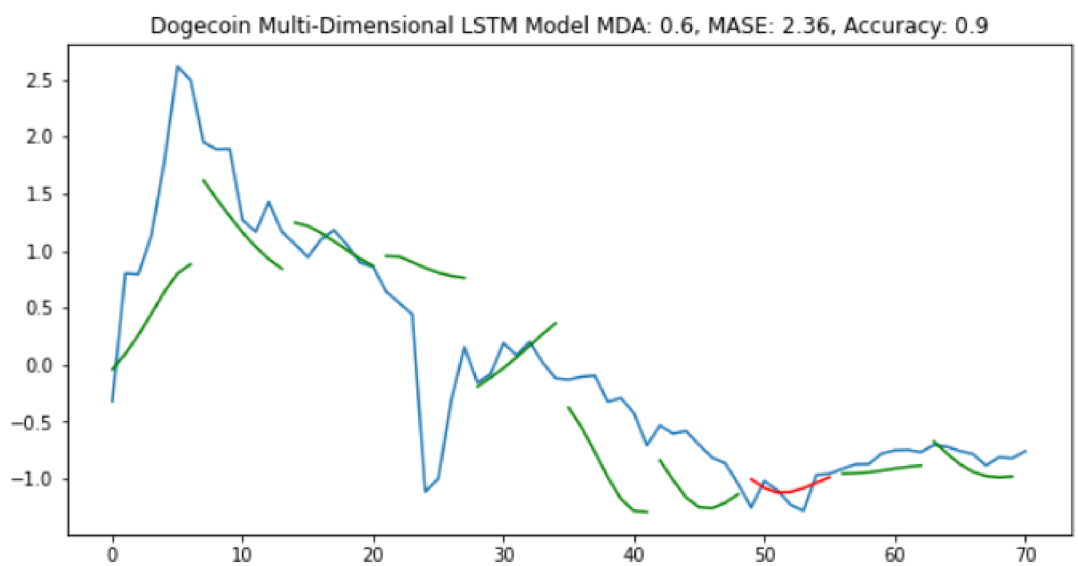


Figure 24 – Dogecoin Final Multi-Dimensional Model Predictions

Future Improvements

In the next phases of this project, I will explore multiple next steps. I want to identify a more effective way of identifying the optimal features that have the most consistent training result. I need to also optimize the RNN parameters including batch size, layers and learning rate.

Finally, the RNN model really did not sufficiently take advantage of all the technical analysis metrics so I will implement a novel approach proposed in this research paper[9] to encode time series data into an image and then make buy/hold/sell decision predictions with a deep convolutional neural network.

---

[9] Algorithmic Financial Trading with Deep Convolutional Neural Networks: Time Series to Image Conversion Approach