
Kaggle Contest: G-Research Crypto Forecasting with Gradient Boosting and LSTM

MAFS 6010Z Project 3

WONG, Hoi Ming (20641276)

WONG, Sik Tsun (20038819)

Department of Mathematics,

The Hong Kong University of Science and Technology

12-Dec-2021

Abstract

We built forecasting models for the excess returns (ie. returns over markets') of 14 cryptocurrencies via gradient boosting and LSTM. We evaluated and compared our models from both methods by out-of-sample Pearson's correlation, and examined the importance of the multifaceted features we engineered from the time series of prices and volume.

1. Introduction

Crypto-currencies are gaining popularity in recent years as an alternative asset class among investors. Over \$40 billion worth of cryptocurrencies are traded every day. Their price volatility remains high and the underlying dynamics has been intriguing for many researchers. In this project we aim to build machine learning models using gradient boosting and LSTM to forecast the short-term returns of crypto-currencies and explored the features useful in prediction.

2. Dataset Description

The dataset we used to build models contains the minute-level trading data of 14 cryptocurrencies from 1-January, 2018 to 21-Sep, 2021. It includes price data ('Open', 'High', 'Low' and 'Close'), numbers of trades as 'Count', volume in terms of crypto-asset units and 'VWAP', the volume-weighted average price. Each crypto-asset is represented by a unique value in the column 'Asset_ID' and the variable to predict is the column 'Target', defined as the residual asset returns¹ in the coming 15-minute period after removing the market signals.

¹ The crypto-asset's beta to the market is estimated using a 3750-minute rolling window. The market return is calculated using fixed asset weights and the same applies to the evaluation metric - weighted Pearson correlation coefficient. Full computation details can be found on the following website.

3. Data Processing

3.1 Feature Engineering

We created 60 new features from the dataset based on different factors, technical indicators and trading setups commonly adopted by traders. We categorize these features into five groups, namely i) trending following, ii) reversal/contrarian, iii) volatility/breakout, iv) relative strength over market and v) miscellaneous features. We will highlight the rationale and details of a few key predictors below. The full collection features added can be found on Table 1.

i) Trend-following Features

Trend following is one of the favorite trading strategies adopted among investors in various asset classes (eg. stocks, commodities futures, etc), and we therefore computed some trend following indicators and examine whether they help predict the dynamics of crypto returns. We computed moving averages of close prices with 10-/20-/50/200-minute windows and created variables for their slopes and patterns as shown in Table 1. We also incorporated standard technical indicators such as Rate of Change (ROC 1/9/20-minute) and Moving Average Convergence and Divergence² (MACD), which some traders use as entry/exit signals for trend following plays.

ii) Reversal/Contrarian Features

In contrast to trend followers, contrarian traders believe in mean reversion and love to capitalize on trend reversals of asset prices. We included the percentage deviation of current close prices from moving average lines (20-/50-/200-minute) to show the tendency of reversal. When the deviation is large, the number of short-term traders with existing winning positions are more tempted to take profits, which makes potential reversal sooner or more likely to happen.

Fig 1: Snapshot of Bitcoin's one-minute chart on 10-Dec-2021 (Investing.com)



iii) Volatility/Breakout Trading Features

Each crypto asset has its own inherent volatility and even for the same crypto currency its volatility can vary over time. A more volatile crypto asset should exhibit greater degree of movement and thus outperform (or underperform) when the entire market is up (or down). We thus computed the 20-/50-

² Details of MACD computation can be found on the webpage below.

<https://www.kaggle.com/cstein06/tutorial-to-the-g-research-crypto-competition/notebook>

<https://www.investopedia.com/terms/m/macd.asp>

/200-minute rolling standard deviation of the asset return to capture the short-term/ mid-term/ long-term volatility.

The time-varying nature of asset return volatility has also engendered another form of popular trading strategies that profit on patterns of volatility contraction period (VCP)³ and breakout of historical trading ranges. VCP traders believe in mean reversion in volatility and periods of compressed trading ranges or reduced volatility should be followed by comeback of big movement. They see series of higher lows in price patterns as hints of strong underlying demand such that buyers do not wait to take the overhead supply. When near-term overhead supply has been dried up, the asset price tends to exhibit relatively large upside and VCP traders will capitalize on longing the asset at the pivot point with a tight stop-loss (shown below). The similar philosophy also applies to shorting the asset. While this strategy may not have a high winning rate, it tends to have attractive risk-to-reward ratios (or equivalently, a high potential gain-to-loss ratio).

Therefore, we created multiple features to mimic volatility contraction. We computed the ratio of 20-minute volatility to 200-minute volatility, and considered the percentage of time in the past 20-/50-/200-minute periods that the short-term volatility is lower than the long-term one. Longer periods of volatility contraction tend to result in greater movement ahead for some assets. We also computed the rolling High/Low price of the assets in 20-/50-/200-minute period and calculated the percentage deviation of the current close price to those levels. This is used to indicate the relative level of current price to a certain historical trading range, showing the distance to a potential breakout.

Fig 2: Volatility Contraction Pattern & Rationale

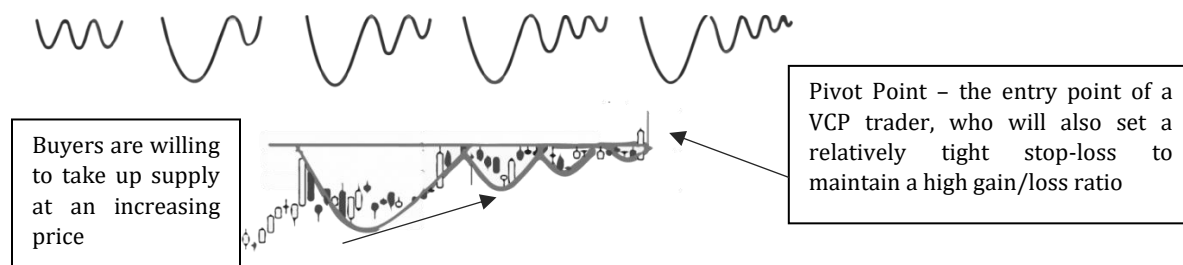
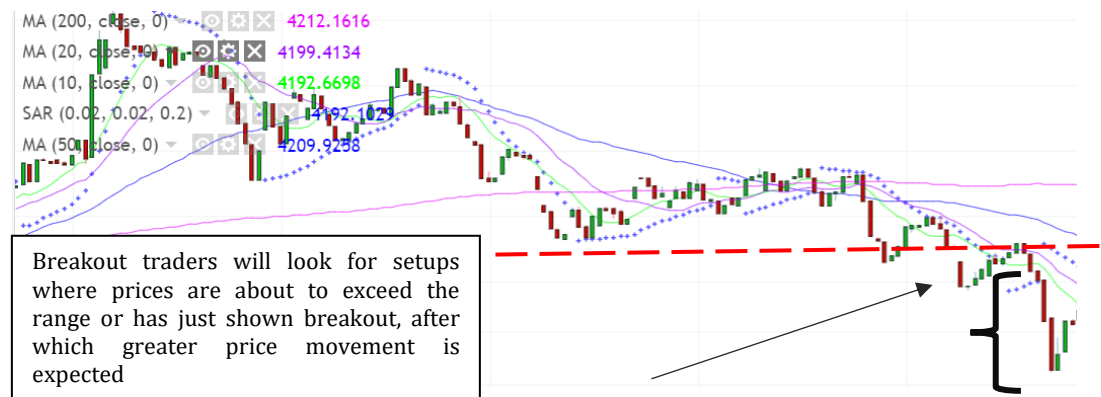


Fig 3: Breakouts - Snapshot of Ethereum's one-minute chart on 10-Dec-2021 (investing.com)



³ The term 'VCP' is coined by Mark Minervini, a former verified US investing champion. More details on trading on VCP can be found in the book *"Trade Like a Stock Market Wizard: How to Achieve Super Performance in Any Market"* (2013) by Mark Minervini

Table 1: Key Features Engineered

Category	Features Added	Variable Name(s)
i) Trend-following Features	<p><u><i>Moving Averages</i></u></p> <ul style="list-style-type: none"> - Simple 10/20-min Moving Average (SMA) on Close - Simple 50/200-min Moving Average (SMA) on Close - 10-min SMA above/below 20-min SMA (1 or 0) - 20-min SMA above/below 50-min SMA (1 or 0) - Slope of 10/20-min SMA <p><u><i>Momentum Factor</i></u></p> <ul style="list-style-type: none"> - Return of 1-min/9-min/20-min <p><u><i>Moving Average Convergence Divergence (MACD)</i></u></p> <ul style="list-style-type: none"> - Exponentially weighted average on Close 12-/26-min - MACD/ MACD Signal Line/ MACD Histogram 	<p>SMA_10MA/ SMA_20MA SMA_50MA/ SMA_200MA MA_up_order1/ MA_down_order1 MA_up_order2/ MA_down_order2 SMA_Slope1/ SMA_Slope2</p> <p>RET/ ROC9/ ROC20</p> <p>EMA12/ EMA26 MACD/ MACD_signal/ MACD_h MACD_crx_above/ MACD_crx_below</p>
ii) Reversal Features	<p><u><i>Deviation from Moving averages</i></u></p> <ul style="list-style-type: none"> - % Deviation of close price from 20-/50-/200-min SMA 	Dev_20MA/ Dev_50MA/ Dev_200MA
iii) Volatility/ Breakout Features	<p><u><i>Rolling Volatility Measure</i></u></p> <ul style="list-style-type: none"> - 20-/50-/200-min volatility - Ratio of 20-min volatility to 200-min volatility <p><u><i>Volatility Contraction Measure</i></u></p> <ul style="list-style-type: none"> - % of time in the past 20-/50-/200-minutes that short-term volatility is less than long-term volatility <p><u><i>Range & Breakout Measure</i></u></p> <ul style="list-style-type: none"> - High price in the past 20-/50-/200 minutes - Low price in the past 20-/50-/200 minutes - % Deviation of from the 20-/50-/200-minute High - % Deviation of from the 20-/50-/200-minute Low 	<p>Vol_20min/ Vol_50min/ Vol_200min Vol_SLT_ratio</p> <p>Pct_vol_compr_20min/Pct_vol_compr_50min Pct_vol_compr_200min</p> <p>High_20min/ High_50min/ High_200min Low_20min/ Low_50min/ Low_200min Dev_to_High20m/ Dev_to_High50m/Dev_to_High200m Dev_to_Low20m/ Dev_to_Low50m/Dev_to_Low200m</p>
iv) Relative Strength vs Markets	<p><u><i>Relative Strength over market</i></u></p> <ul style="list-style-type: none"> - Proxy of crypto-market return - Excess returns of crypto asset against crypto market <p><u><i>Cumulative Relative Strength</i></u></p> <ul style="list-style-type: none"> - Sum of RS in the past 10-/20-/50-/200-min 	<p>RET_M RS</p> <p>RS_10MIN/ RS_20MIN RS_50MIN/ RS_200MIN</p>
v) Miscellaneous	<ul style="list-style-type: none"> - 14-minute RSI - Candlesticks up-shadow/down-shadow - % bar-size of Candlestick - Deviation of Close price to VWAP - 20-minute moving average (MA) of Volume - Ratio of this minute's Volume to 20-minute MA - weekday of the timestamp (one-hot encoded) 	<p>RSI</p> <p>up_shadow/ down_shadow Bar_szie</p> <p>Dev_VWAP</p> <p>VMA_20MA V_vs_V20MA</p> <p>Weekday_0/1/2/3/4/5/6</p>

iv) Relative Strength versus the market

As the target variable is the residual return instead of the absolute return of a crypto-asset, it is pivotal to include not only the features of a cryptocurrency k , but also information about the broad crypto-market. We therefore created a proxy of the market return by computing the weighted average of 1-minute returns of the 14 crypto-currencies. We applied the weights provided by G-Research, the contest organizer in the dataset 'asset_details.csv', which will also be used in model evaluation when computing the weighted Pearson's correlation. The weights are based on turnover and market capitalization of the crypto-assets, and an equally-weighted mean could be considered but it could be biased towards the small coins. In case of missing values for certain cryptos at some timestamps, we omitted those assets at that timestamp and computed the market return using the normalized weights of the remaining available assets.

We also introduced the feature relative strength ' RS ', defined by the return of asset k over the proxy of market return we computed at any timestamp. We computed the cumulative relative strength of a crypto-asset over the market in the past 10-/20-/50-/200-minute period, which will be used for exploring the momentum factor of crypto assets in the sense of relative returns (ie. whether outperforming cryptocurrencies will continue to outperform)

e) Other Miscellaneous Features

We defined some supplementary features for our prediction, but they did not belong to the above categories. We extracted the weekday from timestamps, and computed RSI, metrics on candlesticks, moving averages of volume and deviation to VWAP, all of which are tabulated in table 1.

3.2 Missing values handling

In model training and evaluation, we dropped the observations where the target variables are missing. For the asset Maker (with id = 10) with >30% missing values in the target in the first quarter, we started our model training from the second quarter. For observations with missing value in the features, the GBRT model can explicitly handle such situation, while for the LSTM model we ignored those samples during the training process.

4. Model Specification

4.1 Hierarchical Model Structure – Quarterly Models for each Crypto-asset

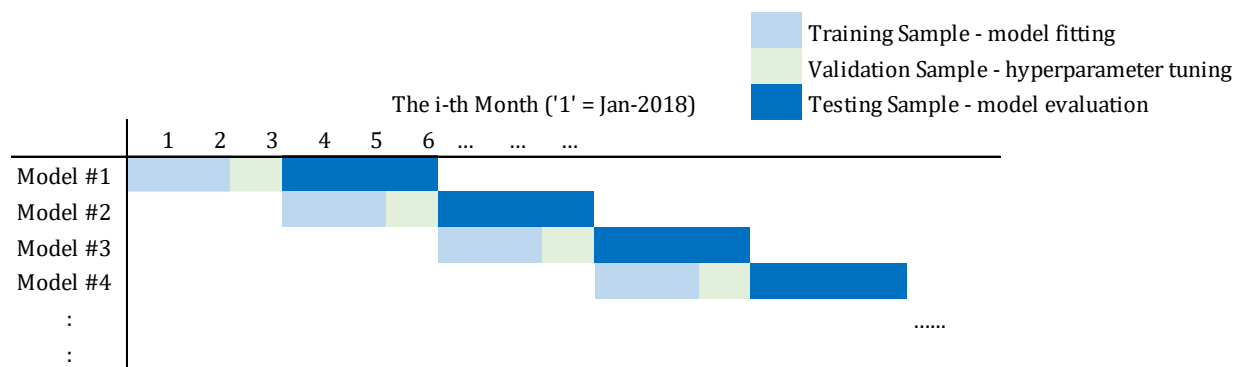
We separately built machine learning models for each of the crypto assets. For each of the k -th cryptocurrency, we fitted quarterly models for prediction, and in each quarter, we will train our model using the data of the k -th asset in the first two months of the past quarter (as illustrated in Fig. 5). The third month of the past quarter was only kept for validation and hyperparameter tuning, which will not enter into the model parameter estimation.

The data of each quarter were used for training only one model to ensure time relevance of information for predicting very short-term returns. The training data for each asset k does not contain the features of the other assets j at all timestamps. Instead, the interaction of asset k with other crypto-assets is captured by the market return ' RET_M ' and k -th asset's excess return ' RS ' which we defined earlier.

This hierarchical model structure applies to all quarters and to 13 of the cryptocurrencies in the data, with the exception of Maker with asset ID equal to 10. For Maker, the length of data available is <700,000

minutes and there exist >30% missing target variables in the first quarter. To avoid having too few sets of training samples, we built monthly models for Maker, as compared to quarterly models for others.

Fig. 4 Illustrations of model training, hyperparameter tuning and evaluation, for each of the asset k (except for Maker)



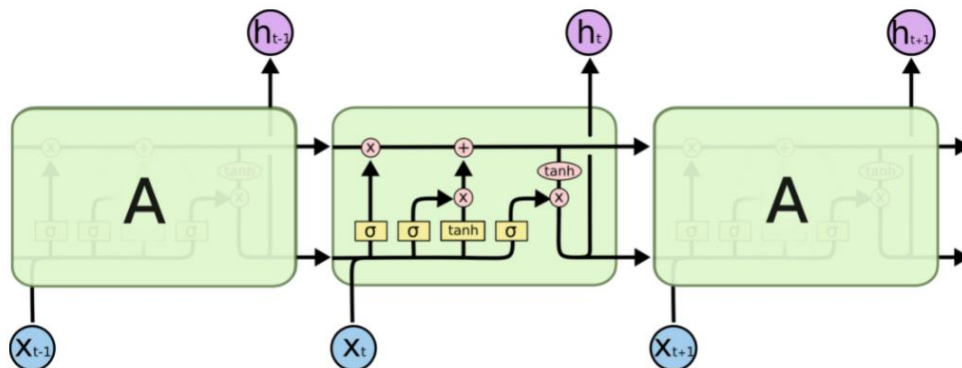
4.2 Model Introduction - Gradient Boosting Regression Tree (GBRT)

Boosting involves combining a collection of weaker learners to produce a stronger learner. Gradient boosting constructs additive regression models by sequentially fitting a weaker learner to current pseudo-residuals at each iteration, which are the gradient of the loss functional being minimized. At each iteration a subsample of the training data is random selected from the full training data set. GBRT provides allows flexibility in optimization on different loss functions and supports missing values as well as different data types. To avoid over-fitting, we introduced an L2 shrinkage parameter to regularize our boosting trees.

4.3 Model Introduction - Long-Short-Term Memory (LSTM)

LSTM is a type of Recurrent Neural Network (RNN) proposed by Hochreiter and Schmidhuber in 1997 as a solution to the vanishing gradient problem. In LSTM model, there are hidden states and cell states, and three gates are introduced to control which information is erased, written, and read. An input gate unit is introduced to protect the memory contents stored from perturbation by irrelevant inputs; and an output gate unit is introduced to protect other units from perturbation by currently irrelevant memory contents stored; a forget gate is introduced to control what is kept or forgotten from previous cell state.

Fig. 5 Illustrations of the repeating module in an LSTM model



LSTM is composed of short-term and long-term memory components which are powerful tools for machine learning on sequential data, such as asset price time series. However, LSTM requires a large amount of computation resources and time for model training. It can also be prone to overfitting, and difficult for result interpretation. As such, in this project, we adopted a two-layer LSTM model with dropout features in each layer to avoid overfitting.

4.4 Hyperparameter Tuning

For GBRT, as mentioned in the section 4.1, we adopted recursive performance evaluation scheme to grid search the hyperparameters for each cryptocurrency. For every fitted model, the data in the first two months is used for training, while those in the third month are for the out-of-sample validation and hyperparameter tuning. MSE loss was used for evaluation in the validation set.

For LSTM, due to the limitation on the computing power, we did not adopt a recursive performance evaluation scheme but instead we performed a grid search by training the data in the first two months with different sets of parameters, and then evaluate the performance based on the out-of-sample testing data in the next month.

Table 2: Values of Hyperparameters Grids of Models

LSTM	GBRT
Number of steps: [30, 60]	Max depth: 5 to 12
Number of units (dimensionality of the output space): [20, 40]	Learning rate: [0.02, 0.05, 0.1, 0.2]
Number of epochs: 35	Number of estimators: 100
Batch size: 64	Loss function: root mean square error
Dropout fraction 0.2	Lambda: [0.1, 1, 10, 100, 1000]
Optimizer: Adam algorithm	
Loss function: mean squared error	

5. Results & Discussion

5.1 Comparison of the correlation coefficients















The values of evaluation metrics of GBT and LSTM for each of the crypto currencies are tabulated in Fig 6. Both modelling approaches showed positive correlation of predictions with the target variables and better than baseline models without adding any new features under the same model set-up. Both models performed well in Litecoin and EOS.IO, while their prediction accuracies diverged in other crypto assets like Bitcoin, Ethereum and Stellar. Machine learning models did improved prediction over baseline, but the absolute level of correlations is still low, probably because the signal-to-noise ratio of cryptos in short time frames (eg. minute levels) is low. The price dynamic of cryptos may also be highly non-stationary such that it could changes in a varying fashion (ie. the change point may not occur at quarters, but in certain months or even days)

Gradient boosting outperformed LSTM on an overall basis, probably because the hyperparameters in LSTM are not tuned for every quarterly model. Due to limitations of computing power, the number of epochs in the LSTM model is set to be 35. If the number is lower than the optimal value, it will lead to underfitting. Similarly, the number of steps is set to be within 60, which means each training and testing















sample contains at most 60-time steps. Thus LSTM's characteristics in retaining useful long-term information may not be fully utilized.

Fig.6 Pearson's Correlation Coefficients for Gradient Boosting Trees & LSTM models

Gradient Boosting

Asset_ID	Weight	Asset_Name		Avg_Corr	Max_Corr	Min_Corr
0	4.304	Binance Coin		0.0182	0.0566	-0.0145
1	6.780	Bitcoin		0.0415	0.1059	-0.0157
2	2.398	Bitcoin Cash		0.0377	0.1201	-0.0074
3	4.407	Cardano		0.0259	0.0549	-0.0013
4	3.555	Dogecoin		0.0298	0.1123	-0.0436
5	1.386	EOS.IO		0.0380	0.0996	-0.0045
6	5.894	Ethereum		0.0388	0.1035	0.0031
7	2.079	Ethereum Classic		0.0362	0.0886	-0.0063
8	1.099	IOTA		0.0276	0.0545	0.0064
9	2.398	Litecoin		0.0419	0.1073	-0.0481
10	1.099	Maker		0.0239	0.0506	-0.0043
11	1.609	Monero		0.0342	0.1018	-0.0283
12	2.079	Stellar		0.0424	0.0813	0.0027
13	1.792	TRON		0.0330	0.0789	-0.0073
Overall				0.0339	0.0898	-0.0126

LSTM

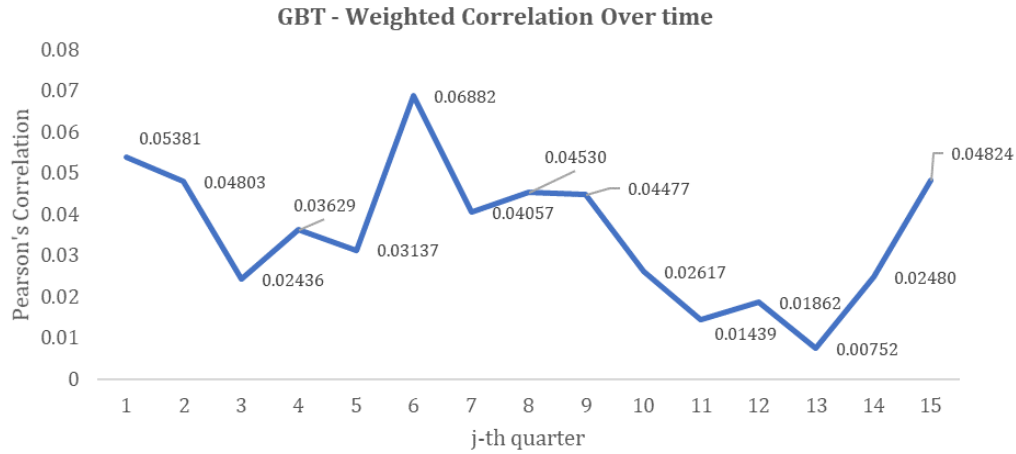
Asset_ID	Weight	Asset_Name		Avg_Corr	Max_Corr	Min_Corr
0	4.304	Binance Coin		0.0110	0.0562	-0.0273
1	6.780	Bitcoin		0.0180	0.0630	-0.0077
2	2.398	Bitcoin Cash		0.0196	0.0790	-0.0151
3	4.407	Cardano		0.0152	0.0633	-0.0090
4	3.555	Dogecoin		0.0344	0.1909	-0.0284
5	1.386	EOS.IO		0.0324	0.1023	-0.0095
6	5.894	Ethereum		0.0162	0.0463	-0.0315
7	2.079	Ethereum Classic		0.0127	0.0453	-0.0253
8	1.099	IOTA		0.0184	0.0346	-0.0051
9	2.398	Litecoin		0.0431	0.0807	0.0128
10	1.099	Maker		0.0162	0.0635	-0.0383
11	1.609	Monero		0.0168	0.0357	0.0011
12	2.079	Stellar		0.0087	0.0358	-0.0419
13	1.792	TRON		0.0161	0.0733	-0.0310
Overall				0.0193	0.0707	-0.0185

5.2 Model Accuracies across Quarters

The overall performance of GBT models was largely steady as we moved along the timeline (shown in Fig. 7). The accuracies varied due to the non-stationarity of asset returns. In particular, the correlation dipped in the 10-12th quarter, the period near mid-2020 when Covid-19 were sweeping and global central banks

have implemented a new set of ultra-loose monetary policies. Price dynamics could have undergone changes under a relatively noisy macro-economic environment, which our models could not well capture.

Fig. 7 Change of Weighted Average Pearson's Correlation over time



5.3 Features importance

As shown in Fig 8, volatility and breakout features showed relatively stronger predictive power in both gradient boosting trees (GBT) and LSTM. In GBT, mid-to-long-term volatility (50-/200-minute period), deviation to rolling trading ranges (Dev_to_Low200m, Dev_to_High200m) and extent of volatility contraction (Pct_vol_compr_200min, Pct_vol_compr_50min) are among the most important features, while in LSTM, deviation to historical trading ranges seemed to be more dominant.

Relative strength against the market (RS) was also found to be useful in predicting future residual returns in both models. Mid-to-long-term relative strength (50-/200-minute) followed volatility features closely in terms of importance in GBT. In LSTM, RS did add show some values, but it is not as prominent in GBT.

One clearer distinction in the two plots is that while GBT did not seem to appreciate the addition of time features, with the one-hot encoded variable `weekday_k` at the bottom, time features are better captured in LSTM in prediction.

Besides, compared to LSTM, GBT seems to be more democratic in terms of the distribution of features importance. The top 10 features in LSTM accounted for over 50% of the importance, and they mostly belong to volatility/breakout and time features. In GBT, miscellaneous features (such as 20-minute moving average of volume), mid-to-long-term trend following features (namely 50-/200-minute moving average, MACD signal) and reversal features (deviation from 200-minute moving average) are ranked at relatively high positions in the plot.

In Fig 9 and Fig 10, we plotted heatmaps to examine the feature importance for each of the asset. The features that are useful in prediction are consistent across various crypto assets in both GBT and LSTM. Even for the asset Maker on which we treated differently by training monthly models, the important features are similar to those from quarterly models of other cryptos.

Fig. 8 Feature Importance of GBT and LSTM – aggregating all crypto assets

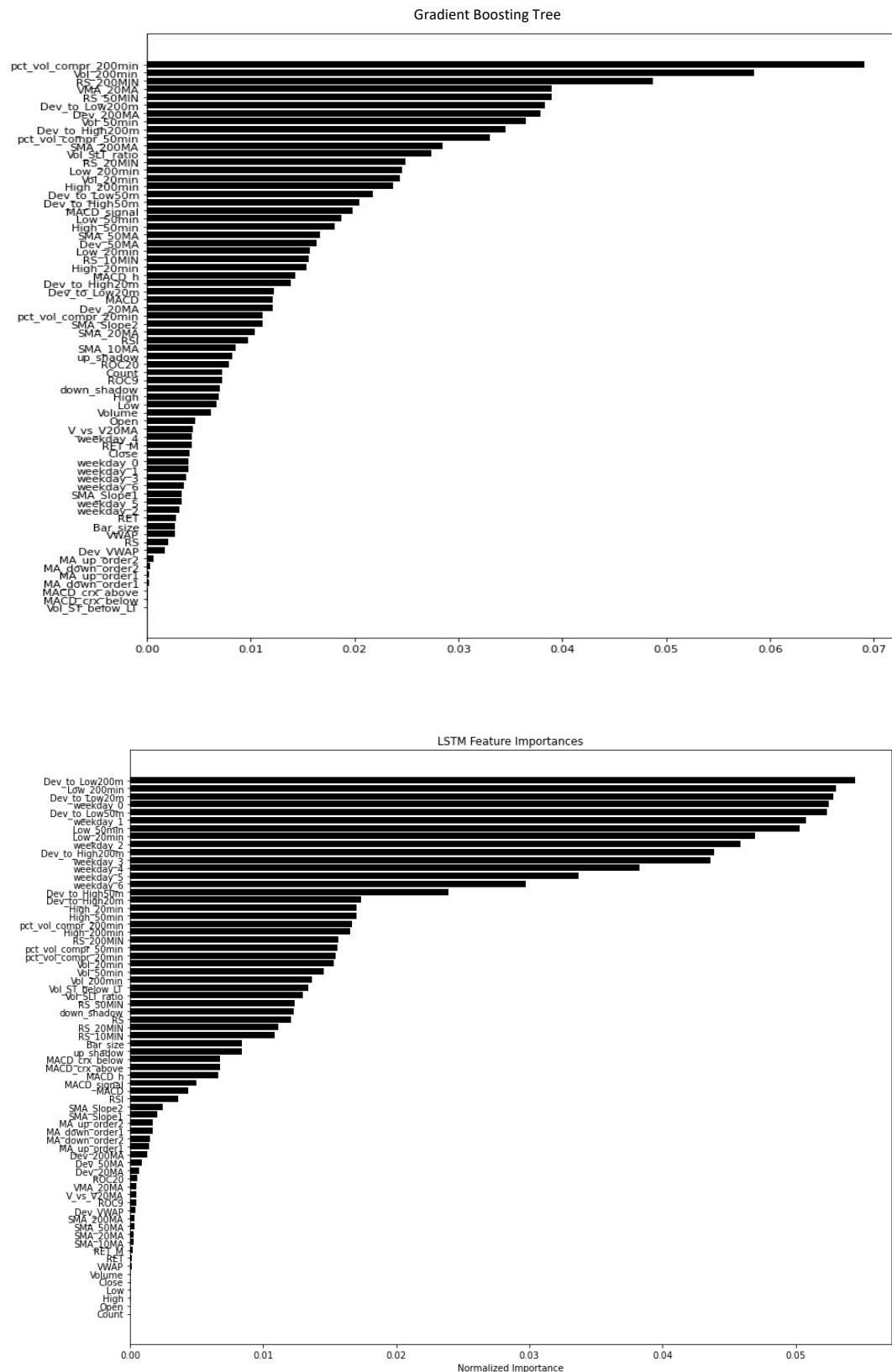


Fig 9 Feature Importance Heatmaps for GBT – Variables are sorted by the aggregate importance of all crypto assets

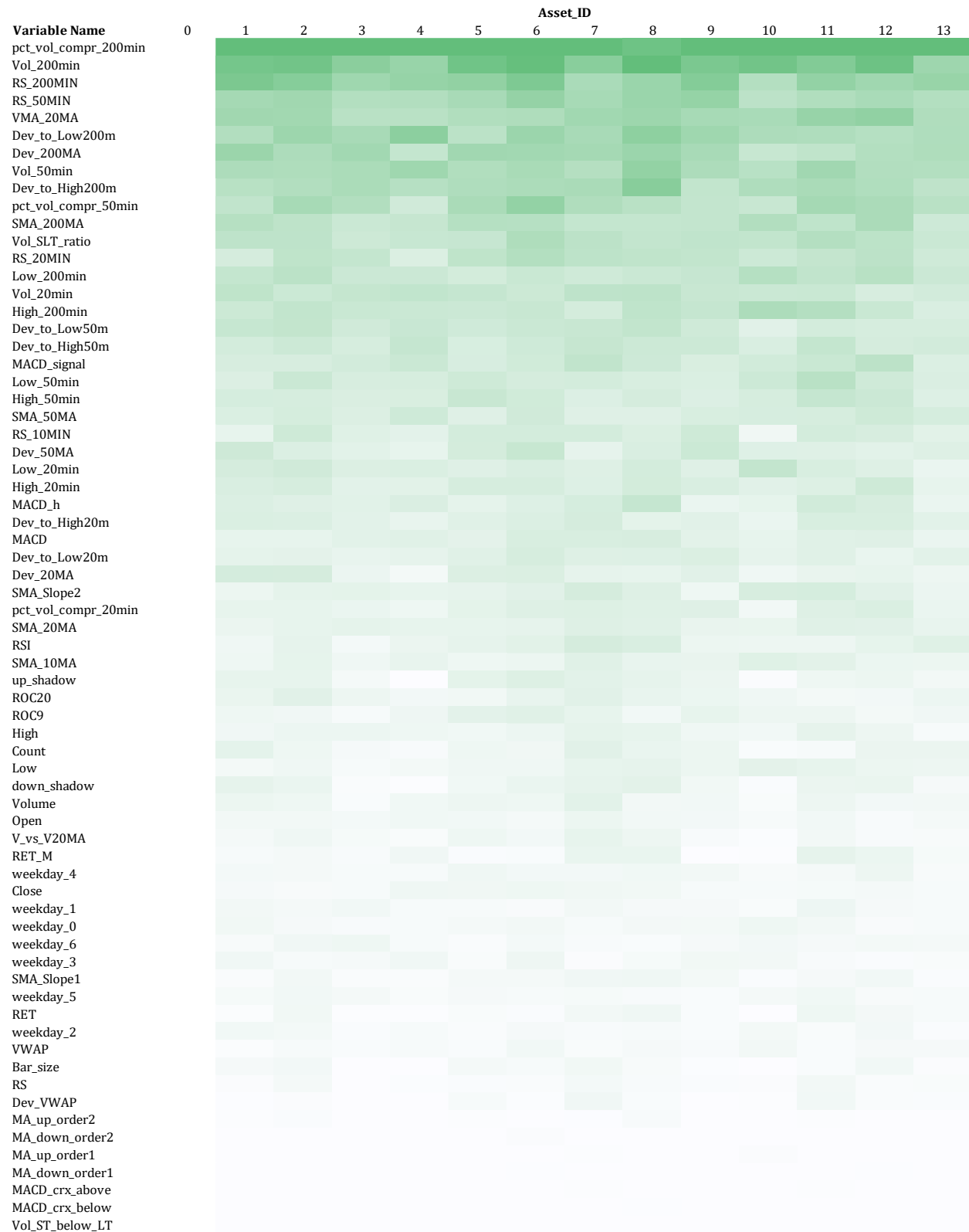
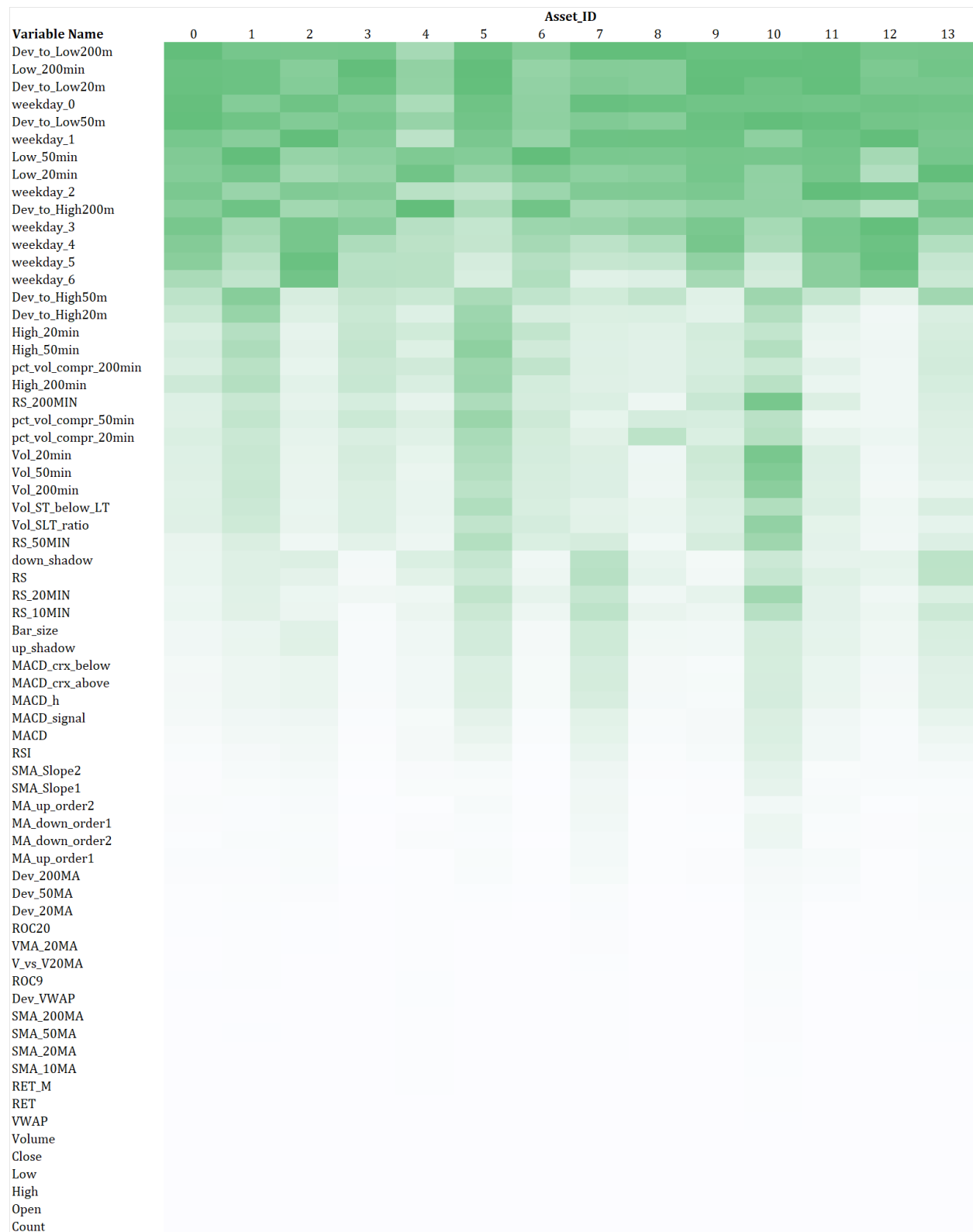


Fig 10 Feature Importance Heatmaps for LSTM – Variables are sorted by the aggregate importance of all crypto assets



5.4 Areas of Improvement

The signal-to-noise ratio is low in the current dataset and therefore further feature engineering could be considered, such as data transformation and inclusion of more technical indicators, time-lag variables, aggregation features, market data and even untraditional data (such as sentiment on Twitter posts). Domain knowledge on cryptos (such as understanding of the asset's protocol) could also be applied.

The hyperparameter tuning process and model architecture can be further improved for the LSTM model. The recursive performance evaluation scheme can be adopted for hyperparameter tuning with a larger range for the grid search on the number of steps and epochs. The model architecture can be further adjusted, such as adding more layers or parallel nodes. However, it should be aware that more complicated architecture with larger number of steps and epochs can lead to overfitting, and therefore one should strike a balance by considering validation results and regularization.

Last but not least, other deep learning models can also be considered, like the Gated Recurrent Unit (GRU) model, which is quicker to compute and has fewer parameters than LSTM. Instead of fitting separated models for each cryptocurrency, ensemble methods, in which separated models for each cryptocurrency are combined with another model fitted with all the cryptocurrencies, could also be considered.

6. Reference

- [1] Friedman, J. H. (March 1999). "Stochastic Gradient Boosting"
- [2] Hochreiter, S. & Schmidhuber, J"urgen, (1997). "Long short-term memory. Neural computation"
- [3] <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>