Crypto Money Machine

Using ML to Identify Short Term Opportunities in Bitcoin

Team members: Charles Brown, Jacob Burnett, Jas Pinglia, Thuy Nguyen

Executive Summary

The Project

Seeks to answer whether machine learning trained with technical indicators could identify a buy and sell period that would generate a profitable return for BTC

The Goals

1. Test the hypothesis that technical indicators could predict short-term price movements.
2. Compare predicted returns against actual returns

FinTech and ML Tools

Accomplishing our goals requires the application of several FinTech tools such as ML models, Python and Pandas coding, and Jupyter Notebooks, to financial trading data.

Data Preparation and Model Training Process

Data Source

Binance

We used 530 days of hourly price data for Bitcoin.

Data Preparation

Data Preparation

- Technical Indicators
- No data gaps
- Scaled Data

Training Process

Training on Multiple Datasets

The training process was complicated by the multiple iterations used.

The Approach

Approach

ML Model Selection

1. Filter classification reports for ML model iterations
   a. F1 > .35
2. Evaluate filtered report and select ML model and buy/sell period

Calculate Algo Returns

1. Using the selected ML model and buy/sell period, calculate algo trading returns on the testing dataset
2. No additional investments before selling
3. No transaction fees

Compare to Actuals

1. Plot actual returns against algo returns for the cryptos
2. Adjust training variables/ML, as needed, to improve model profitability
3. Repeat step 1 until no improvements can be attained

Technical Indicators

Technical Indicators

- RELATIVE STRENGTH INDEX (RSI)
- STOCHASTIC OSCILLATOR %K (STOCHK)
- AVERAGE DIRECTIONAL INDEX (ADX)
- DIRECTIONAL MOVEMENT INDICATOR (DMI)
- MOVING AVERAGE CONVERGENCE DIVERGENCE (MACD)
- SIMPLE MOVING AVERAGE (SMA)
- EXPONENTIAL MOVING AVERAGE (EMA)
- BOLLINGER BANDS (BBANDS)

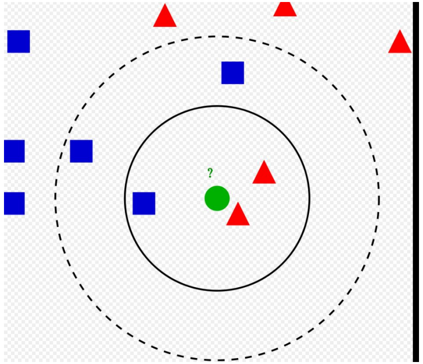
- Relative Strength Index is displayed as an oscillator having a range from 0-100, the traditional usage is that anything above 70 is an indication of being overbought while anything below 30 is being oversold
- Stochastic oscillator is an indicator that measures momentum, by comparing an assets closing price over a range of time. It takes a moving average of that range and generates an overbought or oversold signal utilizing a 0-100 range. The stochastic oscillator uses a 14 day period, and it uses the following to calculate: most recent closing price, lowest price traded in 14 previous days, highest price traded in 14 previous days, current value of stochastic indicator.
- Average directional index works with the next indicator. Directional movement index, or DMI, which is used to observe the strength of ADX. When ADX is above 25 the trend has strength and when ADX is below 25 the trend is considered weak.
- MACD is calculated by taking 26 period EMA and subtracting it by the 12 period EMA which results in MACD signal line. The MACD signal line is triggered when the MACD line crosses over or under the MACD signal line. When it crosses over it is a buy signal and when it crosses under it is a sell signal.
- SMA calculates the number of averages of a range of prices, in our model we use sma for 5, 10, 20, 30, 50, 100, 200 for time in our model.
- EMA which is similar to SMA, with the main difference is that it gives more weight to the recent price data in our model we use 5, 10, 20, 30, 50, 100, 200 for time in our model.
- Bollinger bands are comprised of three lines the upper, middle and lower, the middle represents the moving average, and the upper and lower are set by the standard deviation and are positioned below and above the moving average.

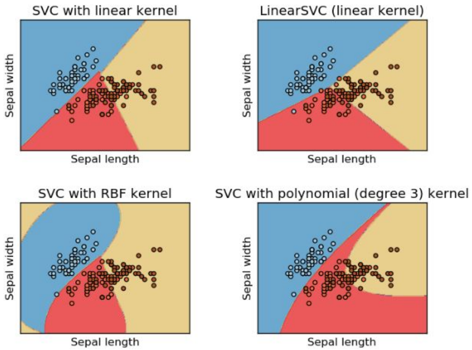
Machine Learning Models

Machine Learning Models

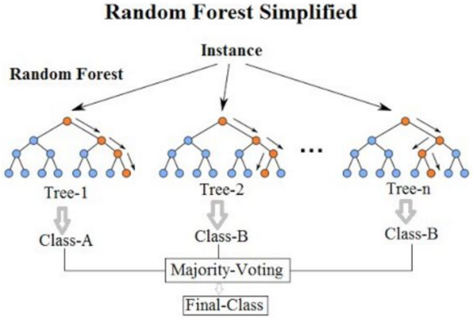
Models Used:

1) K-Nearest-Neighbors Classifier
2) Simple Vector Machine SVC model
3) Random Forest Classifier
4) AdaBoost Classifier





Models in depth



1) K-Nearest-Neighbors Classifier

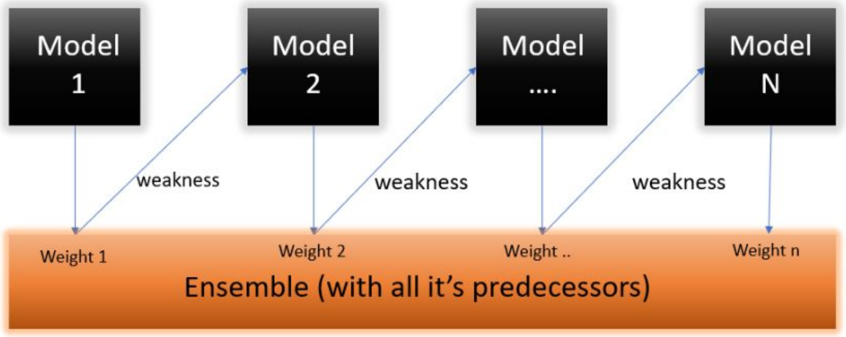
Neighbors are weighted based on distances

2) Simple Vector Machine SVC model

Attempts to find the best hyperplane to separate the different classes.

3) Random Forest Classifier

An ensemble model which creates a multitude of decision trees. A Majority ~~vote~~ is taken and final class picked.



Why adaBoost?

AdaBoost = Adaptive Boosting

This is an ensemble model like a decision tree with only one split called a stump.

The algorithm build a model giving equal weights to all data points. It gives higher weights to data points that are wrongly classified in each iteration (next model).

It will keep training until error is minimized.

Machine Learning Models

After the iteration, we found that Random Forest - 30 period return had the best result and we used that model going forward.

	_0_precision	_1_precision	_0_recall	_1_recall	_0_f1	_1_f1	_0_support	_1_support	_accuracy
column_name									
SVC()23_period_return	0.775839	0.391756	0.135458	0.934330	0.230646	0.552045	8534.0	5086.0	0.433774
SVC()24_period_return	0.766621	0.394819	0.133475	0.932943	0.227364	0.554834	8466.0	5130.0	0.435128
SVC()25_period_return	0.759545	0.392863	0.133932	0.929663	0.227711	0.552322	8467.0	5104.0	0.433203
SVC()26_period_return	0.774806	0.395084	0.130528	0.937378	0.223417	0.555878	8435.0	5110.0	0.434921
SVC()27_period_return	0.771841	0.393472	0.126839	0.937917	0.217874	0.554375	8428.0	5090.0	0.432238
SVC()28_period_return	0.782861	0.394535	0.128150	0.941119	0.220247	0.555989	8412.0	5078.0	0.434173
SVC()29_period_return	0.769874	0.396441	0.132010	0.935269	0.225375	0.556847	8363.0	5098.0	0.436223
SVC()30_period_return	0.780105	0.397222	0.125165	0.942330	0.215719	0.558865	8333.0	5098.0	0.435336
RandomForestClassifier()29_period_return	0.768603	0.388949	0.103558	0.948192	0.182523	0.551622	8179.0	4922.0	0.420884

The Results and Conclusions


actual_cumulative_returns
algo_cumulative_returns
6
5
4
3
2
1
0
−1
2020-07
2020-09
2020-11
2021-01
2021-03
2021-05
2021-07
2021-09
2021-11
2022-01
Date

Results and Conclusions

We were unable to find a combination of time periods and technical indicators that produced a return on investment higher than buy and hold.

In fact, we did not find one with a positive return.

Results and Conclusions

We looked at several training periods, holding durations, and ML models. None were able to achieve proficiency at creating profitable buy signals.

Results and Conclusions

Conclusion 1: Identifying profitable buy signals is hard.

Results and Conclusions

Conclusion 1: Identifying profitable buy signals is hard.

Conclusion 2: Hourly variability and highly volatile underlying stocks/coins complicates the training process. Possibly too much noise.

Results and Conclusions

Conclusion 1: Identifying profitable buy signals is hard.

Conclusion 2: Hourly variability and highly volatile underlying stocks/coins complicates the training process. Possibly too much noise.

Conclusion 3: Using a single ML model might be insufficient.

Next Steps

Next Steps

Next Step 1: Change the underlying and move to daily returns for reduced noise and volatility.

Next Steps

Next Step 1: Change the underlying and move to daily returns for reduced noise and volatility.

Next Step 2: Continue to look for better ML models.

Next Steps

Next Step 1: Change the underlying and move to daily returns for reduced noise and volatility.

Next Step 2: Continue to look for better ML models.

Next Step 3: Consider a multiple model approach to improve results.

Crypto Money Machine

Using ML to Identify Short Term Opportunities in Bitcoin and Alts

Team members: Charles Brown, Jacob Burnett, Jas Pinglia, Thuy Nguyen