Quantitative cryptocurrency trading: exploring the use of machine learning techniques

Quantitative cryptocurrency trading: exploring the use of machine learning techniques

Giuseppe Attanasio
Politecnico di Torino
Torino, Italy
giuseppe.attanasio@polito.it

Luca Cagliero
Politecnico di Torino
Torino, Italy
luca.cagliero@polito.it

Paolo Garza
Politecnico di Torino
Torino, Italy
paolo.garza@polito.it

Elena Baralis
Politecnico di Torino
Torino, Italy
elena.baralis@polito.it

ABSTRACT

Machine learning techniques have found application in the study and development of quantitative trading systems. These systems usually exploit supervised models trained on historical data in order to automatically generate buy/sell signals on the financial markets. Although in this context a deep exploration of the Stock, Forex, and Future exchange markets has already been made, a more limited effort has been devoted to the application of machine learning techniques to the emerging cryptocurrency exchange market. This paper explores the potential of the most established classification and time series forecasting models in cryptocurrency trading by backtesting model performance over a eight year period. The results show that, due to the heterogeneity and volatility of the underlying financial instruments, prediction models based on series forecasting perform better than classification techniques. Furthermore, trading multiple cryptocurrencies at the same time significantly increases the overall returns compared to baseline strategies exclusively based on Bitcoin trading.

KEYWORDS

Cryptocurrencies, Machine learning, Quantitative trading, Classification

1 INTRODUCTION

Traditional online trading platforms allow users to exchange stocks, currencies, and derivatives on the financial markets. Discretionary trading systems rely on signals that are manually generated by domain experts (e.g., buy or sell a given financial instrument at given price). Conversely, quantitative trading systems automatically generate signals based on decision rules which have previously been validated through backtesting over historical data. In the last two decades, the machine learning community has deeply explored the use of machine learning techniques (e.g., classification, regression, time series forecasting) to automatically generate profitable trading signals [9]. However, with the advent of the Bitcoin network in 2009, new trading platforms for the exchange of digital cryptocurrencies have born. Due to the lack of seasonality and the high volatility of the Bitcoin exchange market, the effectiveness of exiting prediction models is not guaranteed. This paper explores the applicability of machine learning techniques to predict the next-day prices of a variety of cryptocurrencies. Furthermore, it presents a trading system that combines the signals generated by various cryptocurrencies and tests its performance over a eight year period with different prediction models and under heterogeneous market conditions.

Most of the recent studies on quantitative cryptocurrency trading aimed at forecasting the price of Bitcoin, which is the leading and most capitalized cryptocurrency (around 72 Billions of USD in March 2019). For example, in [3, 5, 7, 12, 14] the authors addressed the prediction of the next-day direction (up or down) of Bitcoin using binary classification models trained on historical data. They considered, amongst other, MultiLayer Perceptron and genetic algorithms [12], Logistic Regression, Random Forest, Support Vector Machines [5], Bayesian Neural Networks [3], Long Short Term memories and Recurrent Neural Networks [7, 14]. Parallel attempts to perform intraday predictions of Bitcoin prices have also been made (e.g., [11, 13]). Since Bitcoin is also a distributed network that enables users to store and transfer digital currency, a particular attention has been paid to the enrichment of time series data with ad hoc features related to Bitcoin trading and the Bitcoin network (e.g., the average number of transactions per block). The feature engineering process is aimed at including in the prediction models new variables that describe potentially discriminating factors, such as the user activities, the level of attractiveness for investors, and the global macro-financial factors [2, 10]. Furthermore, the correlation between the spread of the Bitcoin's price and the volumes of the related tweets or media published on the Web has been investigated as well [6].

The main limitations of the existing studies can be summarized as follows: (i) The majority of the previous works focused on Bitcoin, while disregarding the other (potentially profitable) cryptocurrencies. (ii) To the best of our knowledge, the approaches relying on multiple cryptocurrencies have not explored the advantages of spreading trading positions over multiple cryptocurrencies. (iii) Backtesting has sometimes been performed over a limited time span. This paper aims at overcoming the aforesaid limitations by proposing a trading strategy relying on multiple cryptocurrencies. The proposed strategy integrates various prediction models, including some of the most established classification techniques as well as some popular time series forecasting methods. To validate model performance, we considered a eight year time period characterized by heterogeneous market conditions. For each prediction model we explored its ability to automatically generate profitable trading signals under multiple aspects.

This paper is organized as follows. Section 2 presents the trading strategy based on Machine Learning techniques. Section 3 validates the performance of ML models on real data. Finally, Section 4 draws conclusions and presents the future extensions of this work.

2 METHODOLOGY

Cryptocurrency markets allow traders to operate on the exchanges between the digital, decentralized coins and the most renowned real currencies (typically USD or Euro). Since cryptocurrencies are characterized by non-stationary, highly volatile, and non-seasonal price series, predicting their future price is particularly challenging. The method presented in this study aims at applying supervised machine learning models, including classification algorithms and time series forecasting techniques, to predict the next-day prices of a variety of cryptocurrencies. The models trained on historical data are exploited to generate ad hoc trading signals (e.g., *buy Bitcoin because the price is likely to significantly increase in the next 24 hours*).[1]

The main steps of the presented methodology can be summarized as follows:

- **Data collection and preparation phase**: For each cryptocurrency, the series of the historical prices are stored and enriched with additional features.
- **Training phase**: Separately for each cryptocurrency, prediction models are trained every day over historical data using an expanding window approach.
- **Trading signal generation**: The trained models are applied to predict the next-day cryptocurrency price variations. When the expected price variations are sufficiently large, ad hoc trading signals (buy/sell) are generated.
- **Trade and money management**: The trading signals generated based on the machine learning models are used to drive multi-day trades. Trade management relies on the prediction outcomes generated in the considered days.

2.1 Data collection and preparation

The historical price time series of the main cryptocurrencies are collected and stored into a unique repository. Based on the assumptions that prices incorporate all the underlying information about the considered asset and that past series trends are likely to occur again in the future, we train supervised multivariate models on historical data.

Data modelling. To apply classification techniques, historical price time series data are stored in relational datasets (separately for each cryptocurrency). Each record is associated with time stamp t. A record consists of (i) a target feature whose value corresponds to the relative price variation between time stamp t and $t+1$, (ii) a feature for each preceding value in the time series ($W-1$ features), (iii) a set of features representing statistical and technical indicators and oscillators.

The value of the target feature (i) at an arbitrary time stamp t is set as follows: $V_t = 100 \cdot \frac{P_{t+1}-P_t}{P_t}$, where, in our context, time stamps t and $t+1$ represent consecutive dates and P_t and P_{t+1} are the closing prices on dates t and $t+1$, respectively.

Features of type (ii) are the historical prices, which are commonly used to identify the most recurrent trends in the time series. To better characterize the daily price variations, we include not only the daily closing price (at midnight), but also the daily opening, maximum, and minimum prices.

Features (iii) indicate the values of the most renowned statistical indicators and oscillators. Technical indicators and oscillators are commonly used in technical analysis to drive trend following and mean reversion trading systems [8]. Since their use in quantitative trading approaches is established [9], we deemed such features as relevant to predict next-day cryptocurrency prices. Notice that since the values of the moving averages are not normalized, we used, as previously done in many quantitative trading strategies [8], the relative difference between the moving averages computed on different periods (e.g., $\frac{MA(20)-MA(50)}{MA(50)}$). Due to the lack of space, the full list of dataset features derived from technical indicators and oscillators is given in Appendix. Notice also that since time series forecasting algorithms do not support the relational data representation, their input data does not include features of type (iii).

Data cleaning. To avoid introducing bias in the training phase, we removed the records including missing values and we corrected small errors in the data.

Discretization. To tackle the next-day price classification problem, we transform target feature values in discrete bins whose values were recommended by domain experts. Specifically, percentage daily variations above 1% are labeled as *Increase*, because the daily variations is significant. Conversely, variations below -1% are labeled as *Decrease*. Finally, in-between values are labeled as *Stationary*, meaning that the price variation is not significant.

2.2 Training phase

We generated the daily training models using a variety of algorithms:

- **Time series forecasting algorithms**: Auto Regressive and Auto Regressive Integrated Moving Average models (ARIMA), Simple, Double Exponential Smoothing and Holt-Winter's models (EXPSMOOTH), Linear regression (LINREG) [1].
- **Classification algorithms**: MultiLayer Perceptron (MLP), Support Vector Classifier (SVC), Multinomial Naive Bayes (MNB), Random Forest Classifier (RFC) [4].

To predict the (discretized) next-day price variation of a given cryptocurrency, we apply the expanding window approach [4] to pick the records in the training dataset, i.e., we consider all the records associated with all the current and preceding dates and we build a prediction model on them. This allows us to consider not only the most recent price trends but also long- and medium-term recurrences in the data. Notice that a description of short-term trends is, to some extent, incorporated also in the values of some of the main technical indicators (e.g., the moving averages at 5 periods).

[1]Since cryptocurrency markets are open 24/7, we conventionally open/close trades at midnight.

2.3 Trading signal generation

The models are applied to each test record corresponding to the next trading day in order to generate potentially profitable trading signals (e.g., *buy the Bitcoin cryptocurrency today because its price is likely to significantly increase in the next 24 hours*).

According to the prediction outcomes, trading signals are managed as follows: (i) if the predicted class label is *Increase* then generate a *Buy* signal for the corresponding cryptocurrency. (ii) if the predicted class label is *Decrease* then generate a *Sell* signal for the corresponding cryptocurrency. (iii) if the predicted class label is *Stationary* then generate a *Hold* signal for the corresponding cryptocurrency.

2.4 Trade management

The trading system opens new multi-day long positions of equal size every time a *Buy* signal is generated for a given cryptocurrency. When a *Sell* signal is generated, a long position for the corresponding cryptocurrency is closed (if any)[2]. When a *Hold* signal is generated, long positions for the corresponding cryptocurrency are maintained (without opening any new position). To preserve the equity against excessive losses, a trailing stop signal is automatically generated when the cryptocurrency price moves in the unfavourable direction by more than 0.5% ($\frac{reward}{risk}$ >2).

3 EXPERIMENTAL RESULTS

We conducted an empirical evaluation of the proposed method on data referring to a 8-year time period (i.e., from January 2011 to December 2018). To assess the performance of the cryptocurrency trading system in different scenarios, we separately analyzed system performance (in terms of overall return, number of trades, and distribution of the achieved returns over the considered cryptocurrencies) in different years. Hereafter, we will denote as *year X* the dataset used to test system performance over year X. To gain insight into the characteristics of the analyzed periods, Table1 reports for each cryptocurrency the observed maximum and minimum variations of the daily price (expressed in percentage). For example, *year 2018* corresponds to a bearish market condition (Bitcoin maximum daily variation +14%, minimum -17%, yearly variation -73%), while *year 2017* corresponds to a bullish one (Bitcoin maximum daily variation +26%, minimum -16%, yearly variation +1338%). Notice that since some of the considered cryptocurrencies have been introduced quite recently, some datasets include only a subset of them. Specifically, in years 2011 and 2012 only the Bitcoin cryptocurrency was available, while Bitcoin and Litecoin were available in 2013 and 2014. Then, the number of cryptocurrencies significantly increases (6 in 2015, 11 in 2016, 24 in 2017, 23 in 2018).

To simulate real investments, we initially trained each model over one third of the days and tested it the day after. Then, to evaluate the performance achieved on the subsequent dates, we expanded the training window using the strategy described in Section 2.2. Daily percentage returns were computed by summing up the profits and losses generated by the positions closed on each day. To take transaction costs into account, we decreased all the daily percentage returns achieved on each operation by 0.15%.

[2]By construction, we do not open short selling positions, because many brokers still do not offer short selling option

The experiments were all performed on a local workstation running Ubuntu 18.04.2 LTS, with 16 GB of RAM and an i7 8700 CPU.

The rest of this section answers to the following research questions:

- Is the system able to beat the market under different market conditions?
- What is the best performing class of algorithms (classification, time series forecasting) on the analyzed dataset?
- Which is the algorithm that achieved, on average, best performance over the considered time period?
- Is the proposed trading strategy robust to market drawbacks?
- What is the distribution of the trade returns over the different cryptocurrencies?

3.1 Comparison between machine learning approaches and benchmarks

We evaluated the overall returns achieved by our methodology on all the datasets by varying the prediction algorithm. Tables 2-5 report (i) the overall net return (expressed in percentage), (ii) the number of opened trades, and (iii) the average net return per trade (%), separately for each year between 2015 and 2018. Due to the lack of space, we omitted similar results for the preceding years. However, in Table 6 we summarized the overall profits achieved by the best algorithm for years 2011-2014. Hereafter, years between 2015 and 2018 will be considered as representatives since (i) they show rather opposite market conditions (as discussed above), and (ii) they include the highest number of cryptocurrencies (many of them were introduced in the last 3-4 years). To configure the algorithm parameters, for each algorithm we performed a grid search separately on each dataset. The selected configuration settings are reported in Tables 2-6.

On each dataset we compared the performance of our system with that of a benchmark buy-and-hold strategy. The benchmark strategy entails buying at the beginning of the testing period the leading Bitcoin cryptocurrency (which is the only one available in all the datasets) and closing the long position at the end of the testing period. Based on the achieved results, all the machine learning algorithms beat the benchmark on every tested datasets (except for LINREG in year 2017). The achieved overall returns range from from 116% to 4840% depending on the considered algorithm and year.

3.2 Comparison between different classes of prediction models

As shown in Tables 2-5, the strategies based on time series forecasting achieved, on average, higher returns that those based on classification models. By ranking the algorithms by decreasing overall return separately for each dataset, ARIMA, EXPSMOOTH, and LINREG placed in the first three positions in 2 cases out of 8, while in 3 other cases at least two or them ranked first, second, or third. Due to the high volatility, considering very short-term variations (occurring in the very few past days) appears to be more effective than relying on medium- and long-term terms. Therefore, the presence of medium- and long-term technical indicators may introduce a bias in the classification models.

Table 1: Maximum and minimum daily variations of each cryptocurrency over the years.

	2018		2017		2016		2015	
Crypto	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**
ADA	44%	-25%	137%	-25%	-	-	-	-
BCH	52%	-26%	55%	-38%	-	-	-	-
BNB	62%	-31%	58%	-27%	-	-	-	-
BTC	14%	-17%	26%	-16%	13%	-15%	27%	-25%
BTG	48%	-31%	106%	-36%	-	-	-	-
DASH	33%	-20%	46%	-20%	43%	-37%	488%	-75%
DOGE	53%	-33%	77%	-45%	411%	-80%	1349%	-93%
EOS	42%	-22%	186%	-30%	-	-	-	-
ETC	24%	-30%	70%	-28%	28%	-21%	-	-
ETH	20%	-20%	30%	-24%	47%	-27%	46%	-60%
IOT	25%	-27%	48%	-33%	-	-	-	-
LINK	53%	-28%	38%	-25%	-	-	-	-
LTC	33%	-21%	74%	-27%	20%	-20%	41%	-44%
NEO	29%	-25%	66%	-29%	-	-	-	-
QTUM	50%	-33%	1364%	-40%	-	-	-	-
TRX	96%	-35%	94%	-28%	-	-	-	-
USDT	12%	-8%	112%	-50%	-	-	-	-
VEN	-	-	212%	-30%	-	-	-	-
WAVES	48%	-26%	39%	-25%	23%	-22%	-	-
XEM	55%	-30%	126%	-32%	66%	-26%	46%	-32%
XMR	20%	-24%	53%	-24%	90%	-30%	216%	-74%
XRP	37%	-31%	180%	-48%	50%	-26%	156%	-63%
ZEC	24%	-19%	68%	-26%	146%	-84%	-	-
ZRX	38%	-28%	213%	-36%	-	-	-	-

3.3 Comparison between different algorithms

Among the considered time series forecasting methods, Exponential Smoothing performed best in 3 out of 8 dataset in terms of overall return, while it ranked second in 2 cases. Unlike Linear Regression and ARIMA, it relies on a non-linear model, which appears to fit better than linear ones the underlying series. By comparing the tested classification algorithms in terms of overall return per year, their outcomes do not show significant differences. This confirms the hypothesis that classifiers' performance is negatively influenced by features describing technical variables, independently of the context under analysis.

3.4 Robustness of the trading system

A key aspect in trading system evaluation is the robustness against drawbacks (periods in which the market beats the prediction models). Figures 1-4 plot the equity lines related to three representative algorithms (i.e., the best performing algorithm, a fairly good approach, and the worst performing one). We assumed to trade with a starting equity of 100,000 Euros using a fixed amount per trade equal to 10,000 Euro. All the tested strategy show fairly robust equity line trends. In year 2018 LINREG achieved very high returns despite the market conditions were bearish. It was able to take advantage of the temporary pullbacks of the market all over the year. Conversely, in year 2017 EXPSMOOTH and MLP performed significantly better than LINREG in a bullish market condition. Finally, equity lines in years 2015 and 2016 are quite stationary due to a more limited daily variations of the cryptocurrency prices.

3.5 Comparison between number of trades and overall return

We studied also the statistical relationship between the number of trades generated by a strategy and the overall return. As shown in Tables 2-5, the two measures appeared to be almost uncorrelated. Time series forecasting models appeared to generate, on average, more trades than classifiers. However, in many cases the average return per trade (ARPT) achieved by classification models is higher than those achieved by time series forecasting methods. Therefore, classification models often generate more accurate signals, but they miss many profitable trades. As discussed in Section 4, as future work we plan to use ensemble methods to overcome this specific issue.

Table 2: Algorithm performance in year 2018

Classifier	Configuration	Overall return	Num trades	ARPT
LINREG	(None)	1486.99%	1129	1.32%
ARIMA	(5,0,0)	926.86%	1352	0.69%
EXPSMOOTH	(0.1,None)	790.17%	1638	0.48%
MLP	((100,),sgd)	571.84%	1135	0.5%
SVC	(poly)	515.38%	814	0.63%
MNB	(1)	482.14%	266	1.81%
RFC	(10,gini)	122.04%	1152	0.11%
Benchmark		*-52.01%*		

Table 3: Algorithm performance in year 2017

Classifier	Configuration	Overall return	Num trades	ARPT
EXPSMOOTH	(0.9,0.1)	4840.37%	701	6.9%
MNB	(1)	3561.12%	652	5.46%
SVC	(rbf)	3188.08%	622	5.13%
MLP	((100,),sgd)	3112.55%	878	3.55%
ARIMA	(3,1,1)	2703.68%	827	3.27%
RFC	(50,gini)	1939.05%	893	2.17%
LINREG	(None)	924.23%	562	1.64%
Benchmark		*1225.22%*		

Table 4: Algorithm performance in year 2016

Classifier	Configuration	Overall return	Num trades	ARPT
ARIMA	(4,1,1)	913.74%	342	2.67%
EXPSMOOTH	(0.5,0.3)	884.42%	442	2.0%
RFC	(100,gini)	794.92%	367	2.17%
LINREG	(None)	571.38%	340	1.68%
MLP	((1000,),sgd)	555.01%	252	2.2%
SVC	(rbf)	483.37%	183	2.64%
MNB	(0,2)	463.31%	166	2.79%
Benchmark		*126.91%*		

Table 5: Algorithm performance in year 2015

Classifier	Configuration	Overall return	Num trades	ARPT
ARIMA	(5,1,1)	1570.07%	202	7.77%
EXPSMOOTH	(0.7,0.1)	1513.52%	265	5.71%
LINREG	(None)	907.02%	129	7.03%
RFC	(10,gini)	530.05%	229	2.31%
MLP	((500,),adam)	367.08%	199	1.84%
SVC	(rbf)	278.76%	101	2.76%
MNB	(0,2)	116.78%	100	1.17%
Benchmark		*73.92%*		

3.6 Return distribution over different cryptocurrencies

Figures 5-8 show the portions of returns due to trade related to each cryptocurrency (excluding those for which no trades were opened). Notably, the distribution is quite heterogenous across different years. For example, in year 2018 most of the profit were due to a

Table 6: Overall returns yielded by the best performing models in years 2011-2014.

Year	Model	Setting	Overall return
2014	EXPSMOOTH	(0.7,0.3)	208%
2013	MNB	(0.2)	693%
2012	SVC	(rbf)	149%
2011	EXPSMOOTH	(0.9,0.1)	1660%

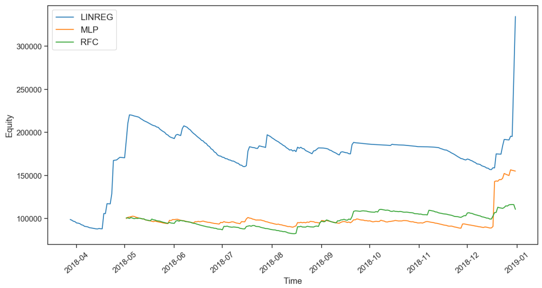

Figure 1: Year 2018: examples of equity lines.

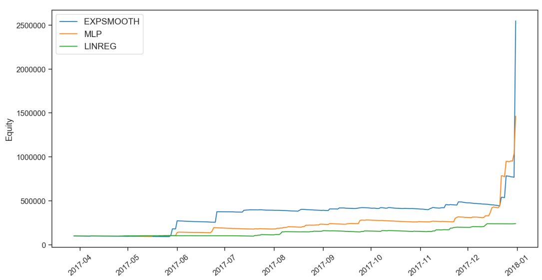

Figure 2: Year 2017: examples of equity lines.

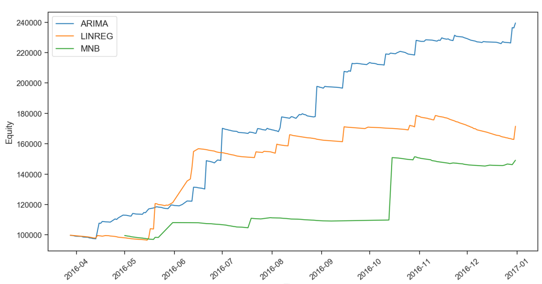

Figure 3: Year 2016: examples of equity lines

single cryptocurrency (BNB), while in the previous years the return distribution is more balanced. To better understand the reasons why profits are unevenly distributed across different assets, we consider again the minimal and maximal daily variations reported in Table 1. For example, the BNB cryptocurrency was quite volatile over year 2018 (maximum 62%, minimum -31%). However, volatility is a necessary but not sufficient condition to yield high returns. For example, although the QTUM cryptocurrency yielded an impressive maximum daily variation (1364%), its trend in the rest of the year

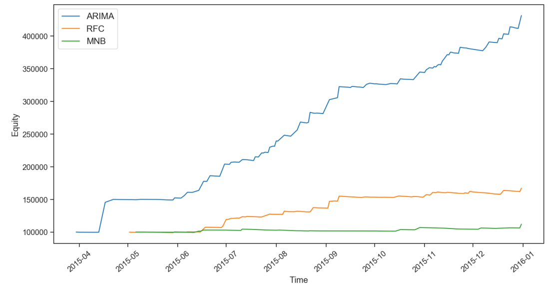

Figure 4: Year 2015: examples of equity lines

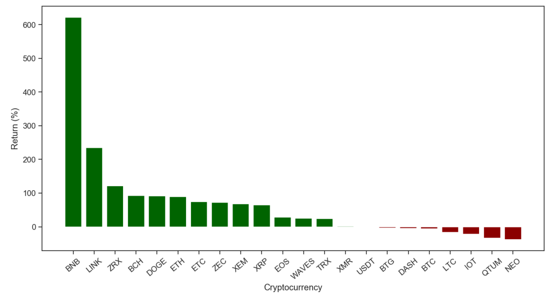

Figure 5: Return distribution over different cryptocurrencies. LINREG. Year 2018.

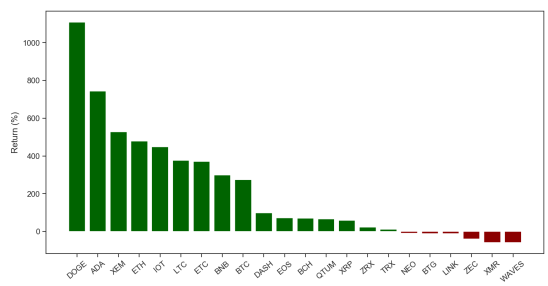

Figure 6: Return distribution over different cryptocurrencies. EXPSMOOTH. Year 2017.

was quite stable. Hence, machine learning models were not able to predict any significant price variations.

3.7 Execution time

Running a complete simulation entails accomplishing data preparation, model training and application. For all the performed tests, the time required to prepare data and generate trading signals are negligible compared to the model training phase.

Training time is in the order of seconds or minutes for all tested combinations of datasets and algorithms. For example, on the dataset corresponding to year 2017 time series forecasting models took between 5s (LINREG) and 1,000s (EXPSMOOTH). Instead, the training time of classification algorithms ranged between 40s (MNB) and 400s (MLP).

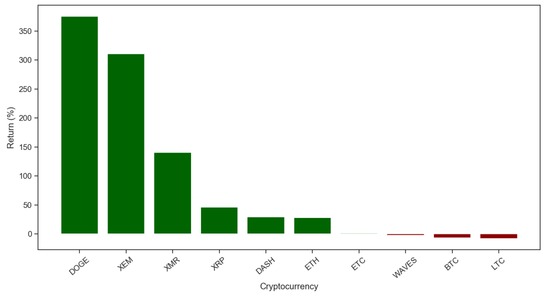

Figure 7: Return distribution over different cryptocurrencies. Year ARIMA. 2016.

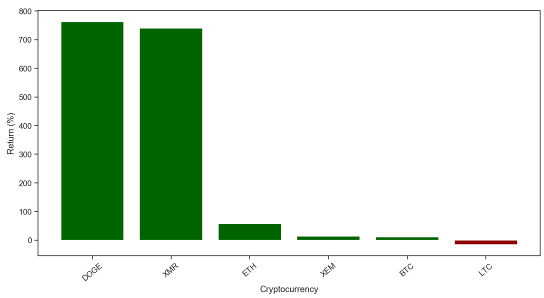

Figure 8: Return distribution over different cryptocurrencies. ARIMA. Year 2015.

4 CONCLUSIONS AND FUTURE WORKS

This paper explored the use of machine learning algorithm to predict the next-day price of several cryptocurrencies. The proposed methods, which integrates a variety of algorithms, was tested on a 8-year dataset. The peculiar characteristics of cryptocurrency series make models relying on very short-term information more effective than traditional approaches based on technical indicators. Furthermore, due to the high volatility of the series profits are typically unevenly distributed across different assets. Hence, trading multiple cryptocurrencies at the same time with the aid of machine learning algorithms is particularly appealing. Since the choice of the best model to use is particularly challenging, as future work we plan to design an ensemble method combining the effectiveness of the time series forecasting methods in capturing short-term terms and the reliability of classification models in making predictions supported by medium- and long-term recurrences in the analyzed data.

REFERENCES

[1] E. Michael Azoff. 1994. *Neural Network Time Series Forecasting of Financial Markets* (1st ed.). John Wiley & Sons, Inc., New York, NY, USA.

[2] Pavel Ciaian, Miroslava Rajcaniova, and d'Artis Kancs. 2016. The economics of BitCoin price formation. *Applied Economics* 48, 19 (2016), 1799–1815.

[3] H. Jang and J. Lee. 2018. An Empirical Study on Modeling and Prediction of Bitcoin Prices With Bayesian Neural Networks Based on Blockchain Information. *IEEE Access* 6 (2018), 5427–5437.

[4] Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. 2014. *Mining of Massive Datasets* (2nd ed.). Cambridge University Press, New York, NY, USA.

[5] Isaac Madan, Shaurya Saluja, and Aojia Zhao. 2015. Automated bitcoin trading via machine learning algorithms. (2015).

[6] Martina Matta, Ilaria Lunesu, and Michele Marchesi. 2015. Is Bitcoin's Market Predictable? Analysis of Web Search and Social Media. In *IC3K 2015, Lisbon, Portugal*. 155–172.

[7] Sean McNally, Jason Roche, and Simon Caton. 2018. Predicting the Price of Bitcoin Using Machine Learning. In *26th Euromicro International Conference on Parallel, Distributed and Network-based Processing, Cambridge, UK*. 339–343.

[8] J.J. Murphy. 1998. *Study Guide for Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance.

[9] Akhter Mohiuddin Rather, V. N. Sastry, and Arun Agarwal. 2017. Stock market prediction and Portfolio selection models: a survey. *OPSEARCH* 54, 3 (2017), 558–579.

[10] Muhammad Saad and Aziz Mohaisen. 2018. Towards characterizing blockchain-based cryptocurrencies for highly-accurate predictions. In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 704–709.

[11] Devavrat Shah and Kang Zhang. 2014. Bayesian regression and Bitcoin. In *52nd Annual Allerton Conference on Communication, Control, and Computing, Allerton 2014, Allerton Park & Retreat Center, Monticello, IL, USA*. 409–414.

[12] E. Sin and L. Wang. 2017. Bitcoin price prediction using ensembles of neural networks. In *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*. 666–671.

[13] Taiguara Melo Tupinambás, Rafael Aeraf Leão Cadence, and André Paim Lemos. 2018. Cryptocurrencies Transactions Advisor Using a Genetic Mamdani-type Fuzzy Rules Based System. In *FUZZ-IEEE 2018*. IEEE, 1–7.

[14] C. Wu, C. Lu, Y. Ma, and R. Lu. 2018. A New Forecasting Framework for Bitcoin Price with LSTM. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*. 168–175.

APPENDIX

A TECHNICAL INDICATORS

This appendix contains the full list of features derived from technical indicators and oscillators (see Table 7).

Table 7: List of dataset features derived from technical analysis.

Feature	Description	Discretization range
RSMA(5, 20)	Relative difference between SMA(5) and SMA(20)	(-∞, 0), [0, +∞)
RSMA(8, 15)	Relative difference between SMA(8) and SMA(15)	(-∞, 0), [0, +∞)
RSMA(20, 50)	Relative difference between SMA(20) and SMA(50)	(-∞, 0), [0, +∞)
REMA(5, 20)	Relative difference between EMA(5) and EMA(20)	(-∞, 0), [0, +∞)
REMA(8, 15)	Relative difference between EMA(8) and EMA(15)	(-∞, 0), [0, +∞)
REMA(20, 50)	Relative difference between EMA(20) and EMA(50)	(-∞, 0), [0, +∞)
MACD	Moving Average Convergence/Divergence	(-∞, 0), [0, +∞)
AO(14)	Aroon Oscillator	(-∞, 0), [0, +∞)
ADX(14)	Average Directional Index	(-∞, 20), [20, +∞)
WD(14)	Difference between Positive Directional Index (DI+) and Negative Directional Index (DI-)	(-∞, 0), [0, +∞)
PPO(12, 26)	Percentage Price Oscillator	(-∞, 0), [0, +∞)
RSI(14)	Relative Strength Index	(-∞, 30), [30,70), [70,+∞)
MFI(14)	Money Flow Index	(-∞, 30), [30,70), [70,+∞)
TSI	True Strength Index	(-∞, -25), [-25,25), [25,+∞)
SO(14)	Stochastic Oscillator	(-∞, 20), [20,80), [80,+∞)
CMO(14)	Chande Momentum Oscillator	(-∞, -50), [-50,50), [50,+∞)
ATRP(14)	Average True Range Percentage: ratio, in percentage, between Average True Range and the closing price	(-∞, 30), [30, +∞)
PVO(14)	Percentage Volume Oscillator	(-∞, 0), [0, +∞)
ADL	Accumulation Distribution Line	(-∞, 0), [0, +∞)
OBV	On Balance Volume	(-∞, 0), [0, +∞)