Prediction of cryptocurrency returns using machine learning — **Source link**

Erdinc Akyildirim, Erdinc Akyildirim, Erdinc Akyildirim, Ahmet Goncu ...+2 more authors

Institutions: ETH Zurich, Mehmet Akif Ersoy University, University of Zurich, University of Liverpool ...+2 more institutions

Related papers:

- The inefficiency of Bitcoin revisited: a high-frequency analysis with alternative currencies

- Speculative bubbles in Bitcoin markets? An empirical investigation into the fundamental value of Bitcoin

- Volatility estimation for Bitcoin: A comparison of GARCH models

- Random Forests

- Bitcoin technical trading with artificial neural network

Prediction of cryptocurrency returns using machine learning

Erdinc Akyildirim[1,2,6] · **Ahmet Goncu**[3,4] · **Ahmet Sensoy**[5]

Abstract

In this study, the predictability of the most liquid twelve cryptocurrencies are analyzed at the daily and minute level frequencies using the machine learning classification algorithms including the support vector machines, logistic regression, artificial neural networks, and random forests with the past price information and technical indicators as model features. The average classification accuracy of four algorithms are consistently all above the 50% threshold for all cryptocurrencies and for all the timescales showing that there exists predictability of trends in prices to a certain degree in the cryptocurrency markets. Machine learning classification algorithms reach about 55–65% predictive accuracy on average at the daily or minute level frequencies, while the support vector machines demonstrate the best and consistent results in terms of predictive accuracy compared to the logistic regression, artificial neural networks and random forest classification algorithms.

Keywords Cryptocurrency · Machine learning · Artificial neural networks · Support vector machine · Random forest · Logistic regression

1 Introduction

The dramatic growth of Bitcoin prices and other cryptocurrencies has attracted great attention in recent years. Increasing more than 120% in the year 2016, and reaching to a 'hard to believe' level of $20,000 from $900 in the year 2017, Bitcoin prices has experienced an exponential growth that led to opportunities of enormous gains that no other financial asset class can

---

✉ Ahmet Sensoy
 ahmet.sensoy@bilkent.edu.tr

 Ahmet Goncu
 Ahmet.Goncu@xjtlu.edu.cn

1 Department of Mathematics, ETH, Zurich, Switzerland

2 Department of Banking and Finance, University of Zurich, Zurich, Switzerland

3 Department of Mathematical Sciences, Xian Jiaotong Liverpool University, Suzhou 215123, China

4 Hedge Fund Research Center, Shanghai Advanced Institute of Finance, Shanghai Jiaotong University, Shanghai, China

5 Faculty of Business Administration, Bilkent University, 06800 Cankaya, Ankara, Turkey

6 Department of Banking and Finance, Burdur Mehmet Akif Ersoy University, Burdur, Turkey
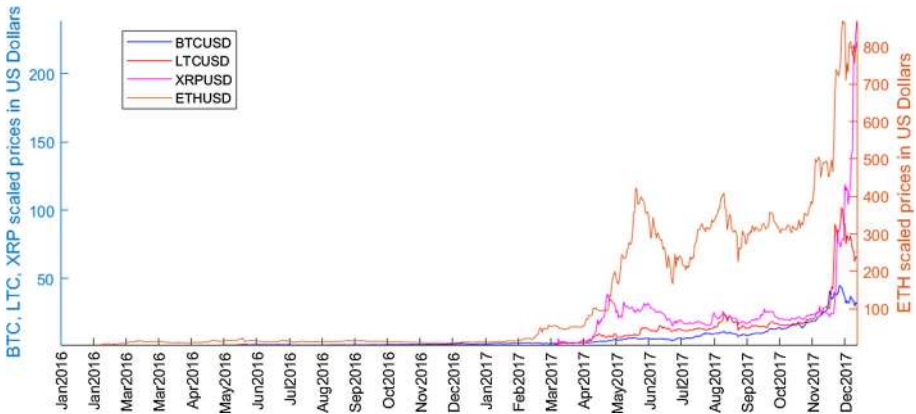
Fig. 1 Daily scaled prices in US dollars for Bitcoin (BTC), Ethereum (ETH), Litecoin (LTC) and Ripple (XRP)

bring in such a short time. Other cryptocurrencies like Ethereum, Ripple and Litecoin were no exception and their prices have increased several thousand percent in 2017 alone (see Fig. 1).

In addition to that, Bitcoin's dominance in market capitalization over the cryptocurrency market has gradually faded from 85% in 2010 to 50% today, showing that an overall attraction to the cryptocurrencies has taken place in the last couple of years.

Lately, as Bitcoin spirals to new lows everyday in the year 2018, while dragging the entire cryptocurrency market down with it, market participants are becoming increasingly interested in the factors that lead to such downturns to understand the price dynamics of these digital cryptocurrencies.

However, from the perspective of a cryptocurrency trader, whether the prices going up or down is no problem as long as the direction is predictable. In the case of an expected boom period, investors can take a long position in cryptocurrencies beforehand to realize their returns once the prices reach up to a certain level. Whereas in the case of a bust period foreseen in the future, investors can short sell these cryptocurrencies through margin trading (allowed by many cryptocurrency exchanges) to gain excess returns. Moreover, taking long or short positions has become much easier after the action taken by the CBOE in December 2017 when they introduced Bitcoin futures. Such a financial asset provides investors to speculate on Bitcoin prices in both directions through leverage without even holding Bitcoins. Similar strategies can be implemented lately for other cryptocurrencies through binary options traded in the offshore exchanges.

All these anecdotes lead us to the question of whether cryptocurrency prices are predictable? In other words, does Efficient Market Hypothesis (EMH) hold for cryptocurrencies? In an efficient market (Fama 1970), any past information should already be reflected into the current prices so that prices might be effected by only future events. However, since the future is unknown, prices should follow a random walk (or a martingale process, to be precise). In the special case of the weak-form efficiency, future returns can not be predicted on the basis of past price changes, however, since the earlier works by Mandelbort (1971) and others (Fama and French 1988; Lo and Mackinlay 1988; Poterba and Summers 1988; Brock et al. 1992; Cochran et al. 1993), weak-form of EMH has been found to be violated in various types of

asset returns,[1] which in turn leads to important problems such as: (i) preferred investment horizon being a risk factor (Mandelbort 1997); and (ii) the fail of common asset pricing models such as CAPM or APT, or derivative pricing models like the Black-Scholes-Merton model (Kyaw et al. 2006; Jamdee and Los 2007).

As evident by the discussions above, EMH has been an intriguing subject for both academicians and market professionals for a very long time, and naturally, the efficiency of cryptocurrencies (especially of Bitcoin) has gained immediate interest due to this fact. For example, the pricing efficiency of Bitcoin has been studied extensively in the last couple of years in various academic papers: Urquhart (2016) provides the earliest evidence on the status of market efficiency for Bitcoin and concludes that Bitcoin is not weakly efficient, however it has a tendency of becoming weakly efficient over time. Building upon that, Nadarajah and Chu (2017) run various weak-form efficiency tests on Bitcoin prices via power transformations and state that Bitcoin is mostly weak-form efficient throughout their sample period. Excessive amount of studies followed on the same topic with different approaches in methodologies, sample frequency, benchmark currency etc. (Bariviera 2017; Vidal-Tomas and Ibanez 2018; Jiang et al. 2018; Tiwari et al. 2018; Khuntia and Pattanayak 2018; Sensoy 2019). Against all different perspectives, the main conclusion is that Bitcoin is inefficient, but to gain weak-form efficiency over time.[2]

Even though the related literature on Bitcoin is satisfactory, other cryptocurrencies has attracted relatively less attention. Brauneis and Mestel (2018) investigate the weak-form efficiency of cryptocurrencies in the cross-section, and show that liquidity and market capitalization has a significant effect on the pricing efficiency. In another study, Wei (2018) analyzes the return predictability of 456 cryptocurrencies and concludes that there is a strong negative relationship between return predictability with cryptocurrency liquidity. Bouri et al. (2019a) analyse various cryptocurrencies and find that price explosivity in one cryptocurrency leads to explosivity in another. Bouri et al. (2019b) examine the role of trading volume in predicting the returns and volatility of several cryptocurrencies and show that trading volume carries useful information to predict extreme negative and positive returns of all sample cryptocurrencies, however it has limited ability to forecast future volatility for a small group of cryptocurrencies. Mensi et al. (2019) compare the efficiency of Bitcoin and Ethereum at the intraday level and show that Bitcoin is more inefficient for overall, upward, and downward trends. In a recent study, Ji et al. (2019) show that cryptocurrencies are integrated within broadly-defined commodity markets.

According to our view, the problem with the abovementioned literature is threefold: (i) There are vast amount of studies on the pricing efficiency of Bitcoin, however other cryptocurrencies are mostly ignored in this strand of literature; (ii) almost all of the previous literature rely on common statistical tests where the outcome of these tests simply reject the null hypothesis of weak-form efficiency or not. However, in the case of inefficiency, these tests provide no explicit way of exploiting the opportunities nor state the potential excess gains that could be obtained through consistent active trading; and (iii) most of the literature deals with daily returns, however high-frequency analysis is ignored.

In this study, we aim to tackle all the abovementioned problems. Using returns obtained at various intraday frequencies for the most liquid twelve cryptocurrencies, we test their return predictability via several methods including support vector machines, logistic regression,

---

[1] See Noakes and Rajaratnam (2016) and Avdoulas et al. (2018) for more recent evidence.

[2] Some studies focus on dependency structure between Bitcoin prices and other variables. For example, see El Alaoui et al. (2019) and Bouri et al. (2018a, b, c). For other various aspects of the cryptocurrency markets, see Cretarola and Figà-Talamanca (2019), Giudici and Polinesi (2019) and Koutmos (2019).

artificial neural networks and random forest classification algorithms. Naturally, our contribution to the literature is manyfold: First, unlike the previous studies that mostly focus on only Bitcoin, we cover a sample of twelve cryptocurrencies. This helps us to understand the overall price dynamics of the cryptocurrency market rather than a single digital currency. Second, previous studies usually use daily data, whereas we cover a sample ranging from a few minutes to daily frequency. This is especially important since in the current status of the financial markets, algorithmic (especially high-frequency) trading is implemented actively, with average asset holding periods barely extend over a few minutes (Glantz and Kissell 2013). Several cryptocurrency exchanges provide algorithmic trading connections to their customers which makes it essential to analyse the cryptocurrency market's pricing efficiency at the intraday level (Sensoy 2019). Third, rather than using the common statistical methodologies to test the pricing efficiency, we refer to the state of the art methodologies used in the decision sciences that provide us the potential patterns to be exploited and the resulting gains if the selected strategy is implemented. Finally, the use of many cryptocurrencies and different timescales the set of features utilized for prediction can be easily verified in their ability to generalize in different timescales for different cryptocurrencies. This is particularly important since most studies on using machine learning algorithms for forecasting consider a single asset at a single timescale without showing the potential of generalization ability of algorithms in different markets and timescales.

Accordingly, we find that the direction of returns in cryptocurrency markets can be predicted for the daily or minute level timescales in a consistent manner with classification accuracies reaching as high as 69% success ratio and with average accuracy for all sample cryptocurrencies around 55–60%. Furthermore, we identify that the support vector machines and even logistic regression algorithms outperform the artificial neural networks and random forest algorithms. Support vector machines are well known for their robustness with respect to noisy data and also have great ability to generalize to different timescales and market conditions. Overall, support vector machines and also logistic regression perform sufficiently well across different timescales and different cryptocurrencies. For market practitioners these results indicate the possibility to design trading rules based on these classification algorithms.

Our findings complement many of the earlier studies by showing that weak form efficiency property in cryptocurrency markets is violated both at daily and various minutely levels supporting the works by Sensoy (2019) on Bitcoin and Mensi et al. (2019) on Ethereum, and extending them for several cryptocurrencies. However, in addition, we show that these inefficiencies can be exploited explicitly with specific algorithms and the resulting potential gains are reported. Our findings also show that trading volume, as an input in the algorithms, can be used in forecasting cryptocurrency returns which supports the results of El Alaoui et al. (2019) on Bitcoin and Bouri et al. (2019b) on a small group of cryptocurrencies and extends them for a much larger group of cryptocurrencies.

The rest of the paper is organized as follows: Sect. 2 presents the data we use in this study. Section 3 explains the methodologies that we employ to uncover the predictability patterns. Section 4 provides the empirical findings and also checks the robustness of the results and finally, Sect. 5 concludes.

2 Data

In this study, we use the dollar-denominated cryptocurrency data from the Bitfinex exchange. We obtain the trade data from the Kaiko digital asset store. Our initial dataset covers the
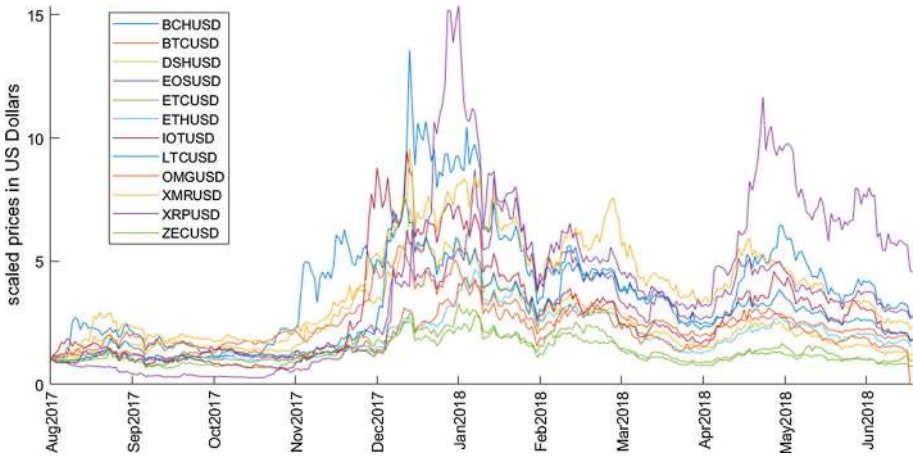
Fig. 2 Daily scaled prices for the selected coins

period from 1 April 2013 to 23 June 2018, where the starting date covers only the trade data of Bitcoin. Within this time period, there are seventy-seven cryptocurrencies which trade against US dollar, however, there are only a few cryptocurrencies with data covering longer than 1 year.[3] The earliest starting date for the other cryptocurrencies' data is 10 August 2017. Hence, in order to have enough number of cryptocurrencies with reasonable number of observations to draw meaningful and robust conclusions, we use two filtering mechanisms. First, we choose cryptocurrencies which have data starting on 10 August 2017 and is still available on the last day of our sample, 23 June 2018. Second, for any selected frequency,[4] we choose the cryptocurrencies having less than 1% non-trading time interval within all time intervals. These two criteria leave us with twelve cryptocurrencies to be analysed: Bitcoin Cash (BCH), Bitcoin (BTC), Dash (DSH), EOS (EOS), Ethereum Classic (ETC), Ethereum (ETH), Iota (IOT), Litecoin (LTC), OmiseGO (OMG), Monero (XMR), Ripple (XRP), and Zcash (ZEC). Figure 2 shows the daily closing prices for each of the selected coins. The prices are scaled by the price as of date 10 August 2017.

As of 17 June 2018, there are 1298 cryptocurrencies traded in the global markets.[5] As it is shown in Table 1, twelve cryptocurrencies that we use in this study already represent 79.8% of the total cryptocurrency market capitalization as of this date. This shows that sample is a good representative of the overall cryptocurrency markets and enables the reader to make more general inferences about the cryptocurrency market as a whole using the results of this paper.

Table 2 presents the number of trade intervals (intervals in which there is at least one trade) for each time frequency together with the ratios of no trade time intervals (intervals in which there is no trade) to the total number intervals in that time frequency. For daily data we have 318 days with at least one trade in each day. As opposed to standard procedure, we do not carry over the last available price for the missing values (no trade intervals). This also strengthens the robustness of our results.

---

[3] These are Bitcoin (BTC): 01/04/2013 to 23/06/2018, Litecoin (LTC): 19/05/2013 to 23/06/2018, Ethereum (ETH): 09/03/2016 to 23/06/2018, Ethereum Classic (ETC): 26/07/2016 to 23/06/2018

[4] We use four different sampling frequencies: 15-min, 30-min, 60-min, and daily.

[5] https://coinmarketcap.com/historical/20180617/.

Table 1 Market value of the selected cryptocurrencies

	Mcap	% of total Mcap
BCH	$14,762,203,191	5.2
BTC	$112,259,483,017	39.9
DSH	$2,168,194,562	0.8
EOS	$9,570,108,557	3.4
ETC	$1,485,200,676	0.5
ETH	$50,463,958,154	17.9
IOT	$3,328,954,782	1.2
LTC	$5,574,619,332	2.0
OMG	$938,946,264	0.3
XMR	$2,031,741,763	0.7
XRP	$21,032,740,889	7.5
ZEC	$804,911,554	0.3
All	$224,421,062,741	79.8

Table 2 Summary statistics of trade intervals for each time frequency

	15-min		30-min		60-min	
	# of trade intervals	% ratio of no trade intervals	# of trade intervals	% ratio of no trade intervals	# of trade intervals	% ratio of no trade intervals
BCH	30,293	0.56	7582	0.52	7582	0.45
BTC	30,373	0.51	7603	0.45	7603	0.38
DSH	30,236	0.86	7593	0.52	7593	0.42
EOS	30,299	0.53	7586	0.46	7586	0.39
ETC	30,354	0.57	7599	0.50	7599	0.43
ETH	30,367	0.53	7603	0.47	7603	0.38
IOT	30,289	0.56	7582	0.52	7582	0.43
LTC	30,359	0.55	7598	0.51	7598	0.45
OMG	29,959	1.00	7515	0.79	7515	0.69
XMR	30,241	0.94	7602	0.52	7602	0.39
XRP	30,303	0.54	7586	0.47	7586	0.41
ZEC	30,240	0.77	7590	0.48	7590	0.38

Tables 3, 4, 5 and 6 provide descriptive statistics for the log-returns in each time frequency. EOS, BCH, and XRP have the highest average returns, and ETC whereas ZEC have the lowest average returns for each of the different time frequencies. Similarly, IOT, EOS, and OMG have the highest volatilities measured by the unconditional standard deviation and DSH, ETH, and BTC have the lowest volatilities, respectively. It is also important to note from the tables that the number of up-moves (increases) and down-moves (decreases) are evenly distributed inside the data for each time frequency. This also significantly contributes to the robustness of our empirical results.

We utilize forty features in prediction of the daily and minute level returns in the next time period. The fundamental features used are the open, close, high, low prices, high-low

Table 3 Summary statistics for 15-min log-returns

Cryptocurrency	Mean	Median	Min	Max	Std	No observations	Up moves	%	Down moves	%
BCH	0.000036	0.000000	−0.178978	0.191568	0.010650	30,273	14,926	49.30	15,116	49.93
BTC	0.000022	0.000046	−0.082220	0.054986	0.006540	30,353	15,330	50.51	14,890	49.06
DSH	0.000007	0.000000	−0.085569	0.187685	0.007805	30,128	14,777	49.05	14,915	49.51
EOS	0.000048	0.000000	−0.193872	0.173927	0.011300	30,272	14,851	49.06	15,059	49.75
ETC	0.000003	0.000000	−0.154112	0.126657	0.009216	30,326	14,934	49.24	15,087	49.75
ETH	0.000015	0.000033	−0.082721	0.105889	0.007087	30,346	15,298	50.41	14,802	48.78
IOT	0.000017	0.000061	−0.259154	0.213093	0.012477	30,269	15,249	50.38	14,772	48.80
LTC	0.000015	−0.000036	−0.125150	0.119767	0.008499	30,340	14,902	49.12	15,224	50.18
OMG	0.000007	0.000000	−0.143482	0.191159	0.011109	29,878	14,696	49.19	14,822	49.61
XMR	0.000029	0.000000	−0.113363	0.138623	0.009099	30,104	14,791	49.13	14,979	49.76
XRP	0.000030	−0.000016	−0.314957	0.314541	0.010783	30,278	14,835	49.00	15,156	50.06
ZEC	−0.000013	0.000000	−0.130850	0.145834	0.008932	30,145	14,651	48.60	15,020	49.83

Table 4 Summary statistics for 30-min log-returns

Cryptocurrency	Mean	Median	Std	Min	Max	No observations	Up moves	%	Down moves	%
IOT	0.000041	0.000014	0.017663	−0.419231	0.239673	15,134	7568	50.01	7500	49.56
EOS	0.000094	−0.000012	0.015909	−0.321127	0.200766	15,145	7457	49.24	7575	50.02
OMG	0.000014	−0.000012	0.015774	−0.198070	0.213111	14,993	7390	49.29	7503	50.04
BCH	0.000069	−0.000130	0.015298	−0.224456	0.183102	15,136	7411	48.96	7652	50.55
XRP	0.000060	−0.000066	0.015269	−0.472213	0.319516	15,147	7444	49.15	7620	50.31
ETC	0.000002	−0.000092	0.013166	−0.222273	0.150772	15,171	7439	49.03	7645	50.39
ZEC	−0.000025	−0.000098	0.012993	−0.138498	0.256382	15,147	7366	48.63	7662	50.58
XMR	0.000054	0.000000	0.012971	−0.217858	0.153393	15,163	7524	49.62	7554	49.82
LTC	0.000028	−0.000076	0.012087	−0.159426	0.168502	15,170	7474	49.27	7641	50.37
DSH	0.000013	0.000000	0.011353	−0.133980	0.264371	15,153	7484	49.39	7553	49.84
ETH	0.000028	0.000104	0.010128	−0.113845	0.149449	15,175	7715	50.84	7399	48.76
BTC	0.000042	0.000119	0.009196	−0.108209	0.096121	15,182	7726	50.89	7413	48.83

Table 5 Summary statistics for 60-min log-returns

Cryptocurrency	Mean	Median	Std	Min	Max	No observations	Up moves	%	Down moves	%
IOT	0.000081	0.000159	0.023858	−0.328414	0.162893	7570	3814	50.38	3726	49.22
EOS	0.000181	−0.000157	0.022045	−0.152634	0.168841	7574	3725	49.18	3819	50.42
OMG	0.000023	−0.000291	0.021795	−0.153523	0.249346	7500	3640	48.53	3829	51.05
BCH	0.000142	−0.000296	0.020927	−0.205439	0.224958	7569	3705	48.95	3849	50.85
XRP	0.000123	−0.000094	0.020360	−0.152697	0.290517	7574	3734	49.30	3818	50.41
ZEC	−0.000052	−0.000216	0.018572	−0.172306	0.374035	7579	3688	48.66	3861	50.94
ETC	0.000010	−0.000125	0.018570	−0.178427	0.207669	7586	3730	49.17	3827	50.45
XMR	0.000109	0.000044	0.018411	−0.166931	0.213591	7589	3800	50.07	3758	49.52
LTC	0.000065	−0.000117	0.016871	−0.185433	0.200330	7585	3726	49.12	3836	50.57
DSH	0.000024	−0.000072	0.016281	−0.158828	0.259575	7581	3735	49.27	3815	50.32
ETH	0.000063	0.000270	0.014203	−0.138714	0.140573	7593	3909	51.48	3659	48.19
BTC	0.000088	0.000183	0.012751	−0.127953	0.112746	7593	3867	50.93	3713	48.90

Table 6 Descriptive statistics for daily returns

Cryptocurrency	Mean	Median	Std	Min	Max	No observations	Up moves	%	Down moves	%
EOS	0.004780	− 0.002710	0.098627	− 0.351719	0.355888	317	151	47.63	166	52.37
BCH	0.003235	− 0.007815	0.096203	− 0.330927	0.435461	317	145	45.74	172	54.26
IOT	0.001931	0.001209	0.095810	− 0.344305	0.365627	317	160	50.47	157	49.53
XRP	0.003159	− 0.002955	0.091568	− 0.375561	0.629046	317	151	47.63	166	52.37
OMG	0.001030	− 0.000012	0.085653	− 0.307426	0.274779	315	157	49.84	158	50.16
ETC	− 0.000078	0.000102	0.082333	− 0.365808	0.293236	317	160	50.47	157	49.53
XMR	0.002668	− 0.002544	0.078107	− 0.287523	0.351604	317	155	48.90	162	51.10
LTC	0.001815	− 0.000549	0.074585	− 0.314821	0.373660	317	157	49.53	160	50.47
ZEC	− 0.000973	− 0.005420	0.074452	− 0.306337	0.234530	317	150	47.32	167	52.68
DSH	0.000608	− 0.001549	0.069781	− 0.232448	0.348499	317	152	47.95	165	52.05
ETH	0.001482	0.001723	0.061021	− 0.225025	0.207365	317	162	51.10	155	48.90
BTC	0.001866	0.003553	0.055051	− 0.206464	0.206737	317	168	53.00	148	46.69

range, number of trades in the given close to open time intervals, US dollar denominated volume, number of cryptocurrencies traded (cryptocurrency volume) for all trades and similarly number of trades, US dollar trade volume, cryptocurrency volume for buyer initiated trades which altogether sum up to eleven features. The second group of features include the last five lagged log-returns. The exponential weighted moving average of the close prices, and the cumulative sum of the last 3 and 5 days of log-returns, and their differences are utilized as ten additional features. Two different relative strength index, two rate of change index and the weighted moving average of close prices are also utilized as features. Next we briefly explain some of the commonly used technical indicators. The complete list of all the features utilized are given in Table 7. Among others, Kara et al. (2011), Huang et al. (2005), Guresen et al. (2011) and Kim (2003) utilize the past price information and similar set of technical indicators as features to predict the asset returns in various other markets except the cryptocurrency markets.

The relative strength index is calculated using the following formula:

$$RSI = 100 - \frac{100}{1 + SMMA(U, n)/SMMA(D, n)}, \tag{1}$$

where the smoothed moving average SMMA, i.e. an exponential moving average of the upward and downward price changes in the last n trading days. During the upward price change, U is calculated as

$$U = \text{close}_{now} - \text{close}_{previous} \quad \text{and} \quad D = 0, \tag{2}$$

whereas during downward price changes, the price closes below the previous close price and we have

$$D = \text{close}_{previous} - \text{close}_{now} \quad \text{and} \quad U = 0. \tag{3}$$

Once the upward and downward price changes are calculated, the exponential moving average is calculated over those values for the predetermined last n trading days. In our set of features, we utilize two separate n values of 9 and 14 days, respectively. Once the relative strength index is calculated, four other indicator features are also created to signal the potentially overbought or oversold assets with respect to the 9- and 14-days RSI values. As given in Table 7, whenever the RSI value is below 20%, a buy signal is given with value equal to 1, whereas another variable is constructed for the sell signal yielding a value of -1 whenever the RSI is above 80%. These additional features are created for both 9- and 14-days RSI values.

Furthermore, the rate of change (ROC) indicator is utilized with 9 and 14 day periods, which is calculated based on the following formula

$$ROC(n) = -100 \times (\text{Last close} - \text{Price } n \text{ days ago})/\text{Price } n \text{ days ago}. \tag{4}$$

Finally, William's percentage R is used as another technical indicator based on the past high and low prices over n-days window as

$$\%R = \frac{High(n) - \text{Last close}}{High(n) - Low(n)}, \tag{5}$$

where n is set equal to 14 trading days window.

Finally, detailed explanation and list of formulas for a wide range of technical indicators can be found in Achelis (1995).

Table 7 Set of features utilized in the classification algorithms

Feature name	Number of lags/window size	Number of features
Open, high, low, close	Last one period	4
High–low	Last one period	1
# of trades, US dollar volume and cryptocurrency volume (for all trades)	Last one period	3
# of trades, US dollar volume and cryptocurrency volume (for buyer initiated trades)	Last one period	3
Returns ($r_{t-1}, \ldots, r_{t-5}$)	Last five periods	5
Moving average (MA)	Last 5 days	1
Correlation MA and close	Last one value	1
$\Sigma_k = \sum_{t=k}^{N} r_{t-1} + \cdots + r_{t-k}$, $k = 3, 5$	Last 3/5 days	2
$\Sigma_5 - \Sigma_3$	Last 3/5 days	1
Relative strength index (RSI)	Window size = 6, 14	2
$\mathbf{1}(RSI_6) < 20\%$, $\mathbf{1}(RSI_{14}) < 20\%$	Buy signals w.r.t. the RSIs	2
$-\mathbf{1}(RSI_6) > 80\%$, $-\mathbf{1}(RSI_{14}) > 80\%$	Sell signals w.r.t. the RSIs	2
Moving average convergence divergence (MACD)	Fast period = 5, slow = 10, signal period = 5	3
Rate of change, rate of change return	Window = 9, 14	2
Exponential weighted moving average (EWMA)	$\lambda = 0.9$	1
Momentum indicator	Window = 5	1
Average true range	Window = 5, 10	2
Williams' %R	Window = 14	1
Aroon stochastic oscillator	Window = 14	1
Commodity channel index	Window = 14	1
Double exponential moving average (DEMA)	Window = 10	1

3 Classification algorithms

There are four time intervals used to calculate the target returns and depending on the return calculated at different time horizons, the binary classification problem is considered. Therefore, the binary target variable is denoted as 1 if the next time step return is up and denoted as − 1 if the next step return is down. Thus, the target variable in the classification algorithm is defined as

$$y_t = \begin{cases} 1 & \text{if} \quad Close > Open \\ -1 & \text{if} \quad Close \leq Open. \end{cases} \tag{6}$$

Four different frequency of returns are utilized including the daily, 15-, 30-, and 60-min returns. By considering the cross-section of twelve cryptocurrencies and a wide range of time

frequencies, we characterize the predictability and forecasting power of supervised machine learning algorithms.

There are four different classification algorithms tested for classifying the target variables at different time frequencies. The implementation of classification algorithms including logistic regression, support vector machines, artificial neural networks and random forest algorithm, are done with the Python's well-known scikit-learn package.[6] In this section, we briefly discuss the application of these classification algorithms without much details but references are provided for detailed discussions of these algorithms in the existing literature.

3.1 Logistic regression

The logistic regression is a widely used classification algorithm which can also be considered as a single layer neural network with binary response variable.

Given the binary classification problem of identifying the next return as up or down, the logistic regression assigns probabilities to each row of the features matrix X. Let's denote the sample size of the dataset with N and thus we have N rows of the input vector. Given the set of d features, i.e. $x = (x_1, ..., x_d)$, and parameter vector w, the logistic regression with the penalty term minimizes the following optimization problem:

$$\min_{w,c} \frac{w^T w}{2} + C \sum_{i=1}^{N} \log(\exp(-y_i(x_i^T w + c)) + 1) \tag{7}$$

where the optimal value of C is selected via the built in cross validation in the logistic regression function of scikit-learn package in Python. Naturally, the value of C determined using the in-sample portion of the dataset and this value is utilized in the out-of-sample predictions.

In most of the scientific computing software there are well-developed packages for logistic regression and other machine learning classification algorithms as well. The main advantage of the logistic regression model is due to its parsimony and speed of implementation. Due to its less number of parameters to be estimated, it is also less prone to the over-fitting problem compared to the artificial neural networks.

3.2 Support vector machines

Support vector machine forecasting algorithms have been successfully used in the literature and such examples can be found in Kim (2003), Huang et al. (2005), Kumar and Thenmozhi (2006), Patel et al. (2015), Lee (2009), and Ince and Trafalis (2008). In particular, support vector machines are suggested to work well with small or noisy data and thus have been widely used in the asset return prediction problems. As discussed in the literature, support vector machine classification has the advantage of yielding globally optimal values. However, still the results of the support vector machines are dependent on the choice of the kernel functions. In this study, the Gaussian (rbf) kernel is utilized however the average performance under the linear kernel is also comparable with the Gaussian kernel in the support vector machines classification.

---

[6] The logistic regression and other classification algorithms are implemented in Python 3.7 with the *scikit-learn package* available on the following website: https://scikit-learn.org/.

Given the training vectors x_i for $i = 1, 2, ..., N$ with a sample size of N observations, the support vector machine classification algorithm solves the following problem given by

$$\min_{w,h,\xi} \frac{w^T w}{2} + C \sum_{i=1}^{N} \xi_i \qquad (8)$$

subject to $y_i(w^T \phi(x_i)) \geq 1 - \xi_i$ and $\xi_i \geq 0, i = 1, 2, ..., N$. The dual of the above problem is given by

$$\min_{\alpha} \frac{\alpha^T Q \alpha}{2} - e^T \alpha \qquad (9)$$

subject to $y^T \alpha = 0$ and $0 \leq \alpha_i \leq C$ for $i = 1, 2, ..., N$, where e is the vector of all ones, $C > 0$ is the upper bound. Q is an n by n positive semi-definite matrix. $Q_{ij} = y_i y_j K(x_i, x_j)$, where $K(x_i, x_j) = \phi(x_i)^T \phi(x)$ is the kernel. Here training vectors are implicitly mapped into higher dimensional space by the function ϕ. The decision function in the support vector machines classification is given by

$$sign\left(\sum_{i=1}^{N} y_i \alpha_i K(x_i, x) + \rho\right). \qquad (10)$$

The optimization problem in Eq. 8 can be solved globally using the Karush–Kuhn–Tucker (KKT) conditions and the details of the derivation can be found in Huang et al. (2005).

3.3 Random forest

Random forest is a learning method that operates by constructing multiple decision trees. The final decision is made based on the majority of the trees and is chosen by the random forest. The main advantage in the use of random forest algorithm is that it reduces the risk of overfitting and the required training time. Random foreast algorithm offers a high level of accuracy and runs efficiently in large datasets while it can be used both in classification and regression problems. Decision trees, which is the building block of random forest algorithm, can be used for various machine learning applications. But trees that are grown really deep to learn highly irregular patterns tend to overfit the training sets. A slight noise in the data may cause the tree to grow in a completely different manner. This is because of the fact that decision trees have very low bias and high variance. Random Forest overcomes this problem by training multiple decision trees on different subspace of the feature space at the cost of slightly increased bias. This means none of the trees in the forest sees the entire training data. The data is recursively split into partitions. At a particular node, the split is done by asking a question on an attribute. The choice for the splitting criterion is based on some impurity measures such as Shannon Entropy or Gini impurity.

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set. Although the use of random forest directly in the return classification is less common compared to the support vector machines or artificial neural networks, promising results have been reported for a few stocks from the US equity market in the recent study by Khaidem et al. (2016).

In random forest method as proposed by Breiman (2001), a random vector θ_k is generated, independent of the past random vectors $\theta_1, ..., \theta_{k-1}$ but with the same distribution; and a tree is grown using the training set and θ_k resulting in a classifier $h(x, \theta_k)$ where x is an input vector. In random selection, θ consists of a number of independent random integers between 1 and K. The nature and dimensionality of θ depends on its use in tree construction. After a large number of trees are generated, they vote for the most popular class. This procedure is called random forests. A random forest is a classifier consisting of a collection of tree structured classifiers $h(x, \theta), k = 1, ...$ where the θ_k's are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x.

3.4 Artificial neural networks

The multilayer perceptron is one of the most commonly used and flexible architecture of neural networks. Multilayer perceptron is capable of approximating a wide range of functions (see Principe et al. 1999). Multilayer perceptron's ability to capture nonlinearity is achieved by the use of smooth activation functions connecting different layers, where common choices of activation functions are the logistic or hyperbolic tangent functions. Furthermore, any element of a given layer feeds all the elements of the next layer. MLPs are normally trained with the backpropagation algorithm. The back propagation rule propagates the errors through the network and allows adaptation of the hidden PEs. The multilayer perceptron is trained with error correction learning, which means that the desired response for the system must be known.

In this study, we utilize the widely used multilayer perceptron (MLP) model of artificial neural networks. In the artificial neural network model, we utilize two hidden layers with forty and two nodes in these layers, i.e. (40, 2), respectively. The number of hidden layers is sufficient to capture potential non-linear relations between the input features, whereas the number of nodes is consistent with the number of features tested. In terms of mapping abilities, the MLP is believed to be capable of approximating arbitrary functions (Principe et al. 1999). This has been important in the study of nonlinear dynamics, and other function mapping problems. Two important characteristics of the multilayer perceptron are: (i) its nonlinear processing elements (PEs) which have a non-linearity that must be smooth (the logistic function and the hyperbolic tangent are the most widely used); and (ii) their massive inter-connectivity, i.e. any element of a given layer feeds all the elements of the next layer (Principe et al. 1999). MLPs are normally trained with the backpropagation algorithm. The backpropagation rule propagates the errors through the network and allows adaptation of the hidden PEs. The multilayer perceptron is trained with error correction learning, which means that the desired response for the system must be known. An example of artificial neural networks successfully utilized in the prediction of the stock returns is given by Kara et al. (2011).

4 Empirical results

As stated earlier, four different classification algorithms are tested at four different time scales including the daily, and the 15-, 30-, 60-min time intervals. The target variable in all these forecasting problems is the open to close return in the next time period. For example, for the daily time horizon we predict the next day's open to close log-returns. The format of our dataset is the same including the open, high, low, and close prices at different frequencies.

For example, for the 5 min level, we predict the open to close return of the next 5 min. The training sample size for all the daily prediction time horizons is set as 80% of the total sample size rounded to the closest integer value, whereas the remaining 20% is utilized as the out-of-sample dataset. Since there is a big difference between the number of observations in the daily versus minute level data, the minute level dataset is split into three different sub-periods for robustness check as well. For example, the results in Table 8 are organized in the order of daily, 60-, 30-, and 15-min. The in sample and out of sample results are given in the first two rows for the 80% versus 20% split of the in-sample and out-of-sample sizes, respectively. The columns in Table 8 show the accuracy of the logistic regression results for different coins considered. Although the accuracy across all the coins are not the same, support vector machines and the logistic regression seem to work in yielding predictive success rate that is always higher than 50% except very few cases. The accuracy rates for the daily timescale can reach over 60% although the dataset has relatively small size and no fine tuning or customization is done for each coin separately.

Alternatively, sub-periods with increasing number of out-of-sample sizes are considered with 200, 400, and 600 observations to verify the performance in different sub-periods. Due to the large number of observations in the minute level data, we also consider a 90% versus 10% split for the in-sample and out-of-sample separation of the dataset separately as presented in the results. The sub-periods that are formed by leaving the 200, 400 or 600 observations out as the out-of-sample part of the data are utilized as well. For each sub-period, we consider a shifted starting point for the training and out-of-sample backtesting. Numerical results are produced with an iMac 3.7 GHz Intel Core i5 computer utilizing the python 3.7 with the sklearn machine learning package. Running the algorithms at the daily timescale can be completed in the order of seconds, whereas at higher frequencies, such as the 15 min sampling frequency, the computational time requirements for training and prediction increases to the order of minutes for the multi-layer artificial neural networks models. In particular, logistic regression and random forest algorithms are much faster in terms of training the models and producing predictions.

As given in Table 8, the logistic regression classifier provides out-of-sample accuracy that is often around 55% with little deviation across different time scales. Depending on the specific coin, the accuracy can also be higher than 60%. Furthermore, it can be noted that with specific model selection methods applied for each cryptocurrency, the performance of the logistic regression can be boosted as well. Overall, the performance of the logistic regression is consistent across most of the cryptocurrencies and different timescales.

In Table 9, the accuracy results obtained from the support vector machine classification algorithm are presented. Compared to the logistic regression classification, support vector machine algorithm provides slightly better performance both on average and also in terms of the best performing case in different cryptocurrencies. The best performance is obtained for the ETH and XMR both at the daily time scale with 69% for both. The last four columns dedicated to the mean, median, minimum and maximum values of each row across different cryptocurrencies. Therefore, when the average performance of the support vector machines classification is compared with the other algorithms, the average accuracy of the support vector machine algorithm is very stable and consistently outperforming the alternatives considered.

The results for the artificial neural networks are given in the Table 10. The accuracy obtained from the artificial neural networks is also at a slightly lower level compared with the support vector machines and the logistic regression. Although the artificial neural networks utilize a more complex model structure and potential to capture the non-linear relationships, there is not any significant or systematic gain from the use of artificial neural networks in

Table 8 Logistic regression classification: in-sample and out-of-sample accuracy results

Logistic	BCH	BTC	DSH	EOS	ETC	ETH	IOT	LTC	OMG	XMR	XRP	ZEC	Across cryptocurrency			
													Mean	Median	Min	Max
Daily 0.8-0.2 in-s	0.62	0.59	**0.67**	0.64	0.63	0.60	0.66	0.59	0.66	0.66	0.62	0.64	0.63	0.63	0.59	0.67
Daily 0.8-0.2 out-s	0.48	0.55	**0.62**	0.53	0.55	0.53	0.59	0.57	0.55	0.57	0.59	0.55	0.56	0.55	0.48	0.62
60 min 0.9-0.1 in-s	0.55	**0.56**	0.53	0.55	0.55	0.54	0.55	0.55	0.55	0.54	0.55	0.54	0.55	0.55	0.53	0.56
60 min 0.9-0.1 out-s	0.55	0.55	0.48	0.54	0.54	0.54	0.55	0.57	0.56	0.53	**0.57**	**0.57**	0.55	0.55	0.48	0.57
60 min out-s sub1	**0.61**	0.52	0.46	0.57	0.54	0.55	0.56	0.56	0.57	0.57	0.59	0.55	0.55	0.56	0.46	0.61
60 min out-s sub2	0.56	0.52	0.48	0.56	0.53	0.53	0.56	0.56	0.56	0.55	**0.57**	0.55	0.54	0.55	0.48	0.57
60 min out-s sub3	0.56	0.53	0.48	0.53	0.52	0.54	**0.57**	0.56	0.57	0.54	0.56	0.54	0.54	0.54	0.48	0.57
30 min 0.9-0.1 in-s	0.54	**0.55**	0.53	0.53	0.53	0.53	0.54	0.53	0.53	0.52	0.54	0.53	0.53	0.53	0.52	0.55
30 min 0.9-0.1 out-s	0.55	**0.56**	0.50	0.54	0.52	0.54	0.54	0.55	0.55	0.52	0.55	0.52	0.54	0.54	0.50	0.56
30 min out-s sub1	0.51	0.53	0.53	**0.57**	**0.57**	0.51	0.56	**0.57**	**0.57**	0.51	0.56	0.51	0.54	0.54	0.51	0.57
30 min out-s sub2	0.53	0.56	0.51	**0.57**	0.54	0.53	**0.57**	0.56	0.55	0.52	**0.57**	0.53	0.54	0.54	0.51	0.57
30 min out-s sub3	0.56	0.55	0.52	0.55	0.54	0.53	0.55	**0.57**	0.55	0.52	0.56	0.52	0.54	0.55	0.52	0.57
15 min 0.9-0.1 in-s	0.52	**0.54**	0.53	0.53	0.53	0.52	0.54	0.53	0.53	0.53	0.53	0.53	0.53	0.53	0.52	0.54
15 min 0.9-0.1 out-s	0.54	0.54	**0.55**	0.54	0.50	0.53	0.53	0.54	0.54	0.53	0.53	**0.55**	0.54	0.54	0.50	0.55
15 min out-s sub1	0.59	0.59	0.55	0.59	0.54	0.55	0.50	0.61	**0.62**	0.57	0.56	0.60	0.57	0.57	0.50	0.62
15 min out-s sub2	0.56	0.57	0.54	0.57	0.52	0.56	0.54	0.59	**0.62**	0.54	0.57	0.60	0.56	0.56	0.52	0.62
15 min out-s sub3	0.53	0.56	0.56	0.54	0.52	0.54	0.53	0.57	**0.60**	0.55	0.55	0.59	0.55	0.55	0.52	0.60

The numbers in bold indicate the maximum value of the numbers in that row

Table 9 Support vector machine (SVM) classification: in-sample and out-of-sample accuracy results

SVM	BCH	BTC	DSH	EOS	ETC	ETH	IOT	LTC	OMG	XMR	XRP	ZEC	Across cryptocurrency			
													Mean	Median	Min	Max
Daily 0.8–0.2 in-s	0.70	0.69	0.72	0.72	0.70	0.72	0.69	0.67	**0.78**	0.71	0.69	0.70	0.71	0.70	0.67	0.78
Daily 0.8–0.2 out-s	0.57	0.52	0.62	0.60	0.53	**0.69**	0.47	0.55	0.66	**0.69**	0.53	0.60	0.59	0.59	0.47	0.69
60 min 0.9–0.1 in-s	0.61	**0.63**	0.60	0.60	0.61	0.61	0.60	0.61	0.61	0.61	0.61	0.59	0.61	0.61	0.59	0.63
60 min 0.9–0.1 out-s	0.56	0.54	0.51	0.51	0.54	0.52	0.55	0.55	0.50	0.53	**0.57**	0.55	0.54	0.54	0.50	0.57
60 min out-s sub1	0.55	0.53	0.54	0.54	0.54	0.55	0.55	0.52	0.59	0.55	0.60	**0.60**	0.55	0.55	0.52	0.60
60 min out-s sub2	0.54	0.51	0.52	0.54	0.53	0.53	**0.55**	0.53	0.54	0.49	**0.55**	0.53	0.53	0.53	0.49	0.55
60 min out-s sub3	**0.57**	0.53	0.50	0.52	0.53	0.53	0.55	0.55	0.54	0.52	**0.57**	0.55	0.54	0.54	0.50	0.57
30 min 0.9–0.1 in-s	0.58	**0.61**	0.58	0.59	0.59	0.59	0.58	0.58	0.59	0.58	0.58	0.58	0.59	0.58	0.58	0.61
30 min 0.9–0.1 out-s	**0.55**	**0.55**	0.53	0.54	0.51	0.54	**0.55**	0.55	0.54	0.51	**0.55**	0.53	0.54	0.54	0.51	0.55
30 min out-s sub1	0.54	0.53	0.56	**0.58**	0.51	0.53	**0.58**	0.58	**0.58**	0.49	**0.58**	0.54	0.55	0.55	0.49	0.58
30 min out-s sub2	**0.58**	0.55	0.53	0.56	0.50	0.56	0.56	**0.58**	0.56	0.51	**0.58**	0.55	0.55	0.56	0.50	0.58
30 min out-s sub3	**0.58**	0.52	0.53	0.54	0.49	0.53	0.55	0.57	0.56	0.50	0.56	0.52	0.54	0.54	0.49	0.58
15 min 0.9–0.1 in-s	0.57	**0.59**	0.57	0.57	0.57	0.58	0.57	0.58	0.58	0.58	0.57	0.57	0.57	0.57	0.57	0.59
15 min 0.9–0.1 out-s	0.55	0.54	**0.56**	0.53	0.51	0.50	0.53	0.55	0.55	0.53	0.53	0.55	0.54	0.54	0.50	0.56
15 min out-s sub1	0.58	0.58	0.51	0.57	0.56	0.57	0.54	0.57	**0.63**	0.57	0.57	0.56	0.57	0.57	0.51	0.63
15 min out-s sub2	0.56	0.56	0.54	0.56	0.53	0.54	0.57	0.58	**0.62**	0.54	0.56	0.58	0.56	0.56	0.53	0.62
15 min out-s sub3	0.54	0.54	0.55	0.54	0.52	0.53	0.56	0.55	**0.60**	0.54	0.56	0.58	0.55	0.55	0.52	0.60

The numbers in bold indicate the maximum value of the numbers in that row

Table 10 Artificial Neural Networks classification: in-sample and out-of-sample accuracy results

ANN	BCH	BTC	DSH	EOS	ETC	ETH	IOT	LTC	OMG	XMR	XRP	ZEC	Across cryptocurrency			
													Mean	Median	Min	Max
Daily 0.8–0.2 in-s	0.97	0.97	0.89	0.94	0.91	0.96	0.94	0.95	0.96	**0.98**	0.80	0.71	0.91	0.94	0.71	0.98
Daily 0.8–0.2 out-s	0.31	0.53	0.53	0.62	0.48	0.59	0.59	0.55	**0.66**	0.52	0.60	0.55	0.54	0.55	0.31	0.66
60 min 0.9–0.1 in-s	0.72	**0.74**	0.71	0.72	0.71	0.73	**0.74**	0.70	0.73	0.71	0.66	0.72	0.71	0.72	0.66	0.74
60 min 0.9–0.1 out-s	0.52	**0.53**	**0.53**	0.52	**0.53**	**0.53**	0.50	0.51	0.52	0.51	0.52	0.49	0.52	0.52	0.49	0.53
60 min out-s sub1	0.46	0.57	0.55	0.54	0.51	0.51	0.53	0.55	0.54	0.49	0.54	0.51	0.52	0.53	0.46	0.57
60 min out-s sub2	0.46	0.54	0.49	0.53	0.54	0.51	0.51	**0.57**	0.54	0.49	0.52	0.52	0.52	0.52	0.46	0.57
60 min out-s sub3	0.52	0.52	**0.53**	0.51	0.52	0.52	0.48	0.51	0.53	**0.53**	0.51	0.49	0.51	0.52	0.48	0.53
30 min 0.9–0.1 in-s	0.64	**0.66**	0.65	**0.66**	0.65	0.65	0.65	**0.66**	**0.66**	**0.66**	0.65	0.65	0.65	0.65	0.64	0.66
30 min 0.9–0.1 out-s	0.53	0.51	0.52	0.52	0.51	0.51	0.53	**0.53**	0.52	0.52	0.51	0.51	0.52	0.52	0.51	0.53
30 min out-s sub1	0.51	0.55	0.55	0.55	0.55	0.52	0.54	0.53	0.54	0.50	0.48	**0.56**	0.53	0.54	0.48	0.56
30 min out-s sub2	0.53	0.56	0.54	0.51	0.52	0.52	0.53	0.54	0.51	0.51	0.54	0.51	0.52	0.52	0.51	0.56
30 min out-s sub3	0.52	0.51	0.53	0.51	0.52	0.51	0.54	**0.55**	0.53	0.51	0.51	0.52	0.52	0.52	0.51	0.55
15 min 0.9–0.1 in-s	0.61	**0.62**	0.61	0.61	0.61	0.60	0.60	0.61	0.60	0.61	0.61	0.61	0.61	0.61	0.60	0.62
15 min 0.9–0.1 out-s	**0.53**	0.52	0.52	0.52	0.50	0.50	0.51	**0.53**	0.51	**0.53**	0.52	0.52	0.52	0.52	0.50	0.53
15 min out-s sub1	0.50	0.52	0.52	0.55	0.58	0.58	0.57	0.55	**0.63**	0.56	0.57	0.61	0.56	0.56	0.50	0.63
15 min out-s sub2	0.54	0.55	0.51	0.52	0.52	0.55	0.54	0.54	**0.64**	0.52	0.51	0.56	0.54	0.54	0.51	0.64
15 min out-s sub3	0.53	0.55	0.50	0.49	0.55	0.53	0.54	0.54	**0.61**	0.52	0.53	0.53	0.53	0.53	0.49	0.61

The numbers in bold indicate the maximum value of the numbers in that row

the prediction of cryptocurrency returns. This might indicate that the more complex model structure might be easily yielding local optima and has lower ability to generalize in the out-of-sample periods. Furthermore, considering the complexity of the artificial neural networks applied in our relatively small sample sizes increases the possibility of achieving sub-optimal results and lower generalization ability. In a larger dataset, it is possible to get better results from the artificial neural network models. In this regard, due to the global optimality of the support vector machines classification, the results confirm the use of support vector machines instead of artificial neural networks given the accuracy results and higher robustness in terms of low variation across different cryptocurrencies and out-of-sample periods.

Finally, the results for the random forest classification algorithm are presented in Table 11. As can be noted in Table 11, the in-sample fit of the random forest algorithm is the highest, however, the out-of-sample performance is drastically lower than the in-sample fits. This indicates the high variance in the random forest classification with high in sample fit to the noisy data but lower out-of-sample performance.

In Table 12, performances of four different models are averaged. First, it is noted that when we consider an ensemble of all the four classification algorithms with the naive equally weights, then all the prediction accuracy results are above 50%. This implies the features utilized, which are various transformations derived from past prices, contain predictive information regarding the direction of the next return. Furthermore, the average accuracy of different classification algorithms across different time scales is consistently above 50% as well, indicating that the past prices contain significant information that yields predictive power of the next time steps' trends.

In our analysis, machine learning algorithms generate consistent results across different cryptocurrencies and over different out-of-sample backtesting samples. However, it is interesting to verify the performance of the machine learning algorithms with some benchmarks like random walk and the traditional time series methods such as autoregressive integrated moving average (ARIMA) models. Since there is no need for in-sample training and out-of-sample testing for the random walk, we consider the whole data set for each time frequency. For each coin and each time scale, we simulate the random walk 1000 times and then compute the average success ratio over these simulations. The results are presented in Table 13. As expected, the success ratios for minute data at higher frequency are very close to 0.5 but for daily time scale the ratios slightly deviate from 0.5 because of a small number of observations. However, the average value across the coins is again 0.5.

For the ARIMA method, we choose the best model by minimizing the Akaike information criterion for the in-sample period. In Table 13, the prediction accuracies in all the out-of-sample backtests are presented for all the cryptocurrencies in the dataset. Comparing the machine learning results versus the ARIMA prediction, it is clear that the ARIMA based models are giving accuracy values scattered around 0.5 indicating that the predictions are no better than the coin toss to estimate the direction of the market. More detailed statistical tests are presented in the rest of the paper to verify whether the differences between the considered methods are statistically different.

In this study, trading strategies are not considered in detail, however, the average accuracies obtained from the four different classification algorithms are all above 50% accuracy regardless of the timescale and cryptocurrency. Also noting that it is possible to improve the results for each cryptocurrency by focusing on the different feature selection methods for each coin separately much higher average accuracies can be obtained. In summary, the average results indicate the inherent predictability of the next period's return directions via transformations of the past price and volume information. Therefore, trading strategies can be designed to exploit the inherent predictability of future price directions.

Table 11 Random forest classification: in-sample and out-of-sample accuracy results

Random forest	BCH	BTC	DSH	EOS	ETC	ETH	IOT	LTC	OMG	XMR	XRP	ZEC	Across Cryptocurrency			
													Mean	Median	Min	Max
Daily 0.8–0.2 in-s	0.99	0.98	0.98	0.98	**0.99**	0.97	0.97	0.98	0.98	**0.99**	0.98	0.97	0.98	0.98	0.97	0.99
Daily 0.8–0.2 out-s	0.57	**0.62**	0.57	0.45	0.53	0.48	0.59	0.48	0.57	0.57	0.57	**0.62**	0.55	0.57	0.45	0.62
60 min 0.9–0.1 in-s	0.98	**0.99**	**0.99**	0.98	**0.99**	0.98	**0.99**	0.98	**0.99**	0.98	0.98	0.98	0.98	0.98	0.98	0.99
60 min 0.9–0.1 out-s	0.53	0.52	0.53	**0.55**	0.51	0.53	0.51	0.53	0.49	0.47	0.52	0.52	0.52	0.52	0.47	0.55
60 min out-s sub1	0.49	0.52	0.52	0.49	0.45	0.50	0.54	0.53	0.52	0.51	0.54	**0.55**	0.51	0.52	0.45	0.55
60 min out-s sub2	0.50	0.50	0.53	0.51	0.45	**0.57**	0.53	0.52	0.52	0.51	0.51	0.49	0.51	0.51	0.45	0.57
60 min out-s sub3	0.53	0.51	**0.55**	0.48	0.52	0.49	0.53	0.53	0.51	0.48	0.51	0.51	0.51	0.51	0.48	0.55
30 min 0.9–0.1 in-s	0.98	**0.99**	**0.99**	0.98	0.98	**0.99**	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.99
30 min 0.9–0.1 out-s	0.51	**0.55**	0.51	0.50	0.51	0.50	0.52	0.51	0.51	0.50	0.52	0.50	0.51	0.51	0.50	0.55
30 min out-s sub1	0.45	0.49	0.49	0.52	0.55	0.54	0.52	0.49	**0.56**	0.53	0.54	0.52	0.51	0.52	0.45	0.56
30 min out-s sub2	0.52	0.51	0.46	**0.57**	0.50	0.53	0.51	0.51	0.50	0.51	0.51	0.54	0.51	0.51	0.46	0.57
30 min out-s sub3	0.50	0.53	0.53	**0.54**	0.48	**0.54**	0.47	0.53	0.53	0.50	0.53	0.51	0.52	0.53	0.47	0.54
15 min 0.9–0.1 in-s	0.99	**0.99**	0.98	0.98	0.98	**0.99**	**0.99**	0.98	**0.99**	0.98	**0.99**	0.98	0.98	0.98	0.98	0.99
15 min 0.9–0.1 out-s	0.50	**0.53**	0.52	0.52	0.50	0.51	0.51	0.51	0.52	0.52	0.52	0.52	0.51	0.52	0.50	0.53
15 min out-s sub1	0.50	0.55	0.51	0.54	0.53	0.55	**0.60**	0.54	0.58	0.57	0.51	0.53	0.54	0.54	0.50	0.60
15 min out-s sub2	0.54	0.52	0.53	0.53	0.48	0.54	0.44	0.50	0.53	0.49	0.54	**0.55**	0.51	0.53	0.44	0.55
15 min out-s sub3	0.51	**0.55**	0.52	0.49	0.49	0.50	0.53	0.51	0.53	0.53	0.54	0.53	0.52	0.52	0.49	0.55

The numbers in bold indicate the maximum value of the numbers in that row

Table 12 Average performance of four classification algorithms, including the logistic regression, support vector machines, artificial neural networks, and random forest classifier, over different cryptocurrencies and different time scales

Average	BCH	BTC	DSH	EOS	ETC	ETH	IOT	LTC	OMG	XMR	XRP	ZEC	Across cryptocurrency			
													Mean	Median	Min	Max
Daily 0.8-0.2 in-s	0.82	0.81	0.81	0.82	0.81	0.81	0.81	0.80	0.85	0.83	0.77	0.75	0.81	0.81	0.75	0.85
Daily 0.8-0.2 out-s	0.48	0.56	0.59	0.55	0.53	0.57	0.56	0.54	0.61	0.59	0.57	0.58	0.56	0.56	0.48	0.61
60 min 0.9-0.1 in-s	0.71	0.73	0.71	0.71	0.71	0.72	0.72	0.71	0.72	0.71	0.70	0.71	0.71	0.71	0.70	0.73
60 min 0.9-0.1 out-s	0.54	0.54	0.51	0.53	0.53	0.53	0.53	0.54	0.52	0.51	0.54	0.53	0.53	0.53	0.51	0.54
60 min out-s sub1	0.53	0.53	0.51	0.53	0.51	0.52	0.54	0.54	0.55	0.53	0.56	0.55	0.53	0.53	0.51	0.56
60 min out-s sub2	0.51	0.52	0.50	0.54	0.51	0.53	0.54	0.54	0.54	0.51	0.54	0.52	0.53	0.53	0.50	0.54
60 min out-s sub3	0.54	0.52	0.51	0.51	0.52	0.52	0.53	0.54	0.53	0.52	0.54	0.52	0.53	0.52	0.51	0.54
30 min 0.9-0.1 in-s	0.69	0.70	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.68	0.69	0.69	0.68	0.70
30 min 0.9-0.1 out-s	0.54	0.54	0.52	0.52	0.52	0.52	0.54	0.54	0.53	0.51	0.53	0.51	0.53	0.53	0.51	0.54
30 min out-s sub1	0.50	0.52	0.53	0.55	0.54	0.52	0.55	0.54	0.56	0.50	0.54	0.53	0.53	0.53	0.50	0.56
30 min out-s sub2	0.54	0.55	0.51	0.55	0.51	0.53	0.54	0.55	0.53	0.51	0.55	0.53	0.53	0.53	0.51	0.55
30 min out-s sub3	0.54	0.53	0.53	0.54	0.51	0.53	0.53	0.55	0.54	0.51	0.54	0.52	0.53	0.53	0.51	0.55
15 min 0.9-0.1 in-s	0.67	0.68	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.68
15 min 0.9-0.1 out-s	0.53	0.53	0.54	0.53	0.50	0.51	0.52	0.53	0.53	0.52	0.53	0.54	0.53	0.53	0.50	0.54
15 min out-s sub1	0.54	0.56	0.52	0.56	0.55	0.56	0.55	0.57	0.61	0.57	0.55	0.57	0.56	0.56	0.52	0.61
15 min out-s sub2	0.55	0.55	0.53	0.54	0.51	0.55	0.52	0.55	0.60	0.52	0.55	0.57	0.54	0.55	0.51	0.60
15 min out-s sub3	0.53	0.55	0.53	0.51	0.52	0.52	0.54	0.54	0.59	0.53	0.55	0.56	0.54	0.54	0.51	0.59

Table 13 Performance of Random Walk and ARIMA time series forecasting over different cryptocurrencies and different time scales

	BCH	BTC	DSH	EOS	ETC	ETH	IOT	LTC	OMG	XMR	XRP	ZEC	Across cryptocurrency			
													Mean	Median	Min	Max
Random walk																
Daily	0.48	0.44	0.45	0.54	0.53	0.49	0.52	0.52	0.48	0.50	0.48	0.51	0.50	0.50	0.44	0.54
60 min	0.50	0.50	0.50	0.50	0.50	0.51	0.50	0.50	0.50	0.49	0.49	0.51	0.50	0.50	0.49	0.51
30 min	0.50	0.51	0.49	0.50	0.50	0.50	0.50	0.50	0.50	0.51	0.50	0.50	0.50	0.50	0.49	0.51
15 min	0.50	0.50	0.50	0.50	0.50	0.50	0.49	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.49	0.50
ARIMA																
Daily 0.8-0.2 in-s	0.47	0.50	0.51	0.48	**0.52**	0.48	0.45	0.48	0.50	0.43	0.47	0.49	0.48	0.48	0.43	0.52
Daily 0.8-0.2 out-s	0.45	0.39	0.42	0.44	0.42	**0.48**	**0.48**	0.47	0.41	0.45	**0.48**	0.45	0.45	0.45	0.39	0.48
60 min 0.9-0.1 in-s	0.46	0.45	**0.48**	0.47	0.46	0.47	0.46	0.47	0.47	0.46	0.45	0.47	0.46	0.46	0.45	0.48
60 min 0.9-0.1 out-s	0.47	0.44	**0.53**	0.46	0.46	0.47	0.47	0.44	0.47	0.50	0.44	0.49	0.47	0.47	0.44	0.53
60 min out-s sub1	0.47	0.45	0.50	0.50	0.50	0.50	0.52	0.52	**0.54**	0.51	0.50	0.48	0.50	0.50	0.45	0.54
60 min out-s sub2	0.45	0.43	0.51	0.44	0.45	0.48	**0.51**	0.46	0.48	0.47	0.48	0.47	0.47	0.47	0.43	0.51
60 min out-s sub3	0.45	0.42	**0.49**	0.46	0.44	0.48	**0.49**	0.47	0.46	0.46	0.45	0.47	0.46	0.46	0.42	0.49
30 min 0.9-0.1 in-s	0.48	0.46	**0.50**	0.48	0.48	0.49	0.47	0.48	0.48	0.48	0.47	0.48	0.48	0.48	0.46	0.50
30 min 0.9-0.1 out-s	0.46	0.47	0.49	0.47	0.47	0.47	0.47	0.47	0.49	0.48	0.45	**0.51**	0.47	0.47	0.45	0.51
30 min out-s sub1	0.53	0.48	0.52	0.52	**0.54**	0.53	0.46	0.47	0.52	0.46	0.47	0.44	0.49	0.49	0.44	0.54
30 min out-s sub2	0.48	0.42	0.46	0.48	0.47	0.50	0.49	0.47	**0.51**	0.47	0.46	0.45	0.47	0.47	0.42	0.51
30 min out-s sub3	0.47	0.44	0.48	0.47	0.49	0.49	0.48	0.47	**0.50**	0.49	0.45	0.48	0.47	0.48	0.44	0.50
15 min 0.9-0.1 in-s	0.49	0.47	0.49	0.48	0.49	**0.50**	0.48	0.49	0.49	0.49	0.47	0.49	0.49	0.49	0.47	0.50
15 min 0.9-0.1 out-s	0.48	0.45	0.47	0.48	0.49	0.49	0.47	0.48	0.49	0.48	0.47	**0.50**	0.48	0.48	0.45	0.50
15 min out-s sub1	0.50	0.48	0.48	0.46	0.50	**0.52**	0.50	0.44	0.51	0.49	0.45	0.50	0.48	0.49	0.44	0.52
15 min out-s sub2	0.52	0.47	0.50	0.47	0.52	0.50	**0.53**	0.45	0.46	0.46	0.45	0.49	0.48	0.48	0.45	0.53
15 min out-s sub3	0.50	0.47	0.50	0.48	0.50	**0.51**	0.50	0.46	0.48	0.47	0.45	**0.51**	0.49	0.49	0.45	0.51

The numbers in bold indicate the maximum value of the numbers in that row

In the context of machine learning estimations, it is important to verify the consistency of predictive power over different products and sub-periods. Furthermore, generalization of the algorithms in new dataset is desirable for robustness. In order to verify the robustness of our empirical results, we not only adopt various statistical tests but also construct two different cryptocurrency indices, namely equally weighted and market capitalization weighted, to verify the predictive power of alternative machine learning models. By constructing these indices, we can test the generalization property of our machine learning model with respect to new data.

In Table 16, t test is applied to check the statistical significance of the estimation results between alternative algorithms. We compare the methods by using the results for each in-sample and out-of-sample sub-period for the same time scale across different coins. For instance, in order to compare Logistic and ARIMA models for daily 0.8–0.2 in-sample period, we use the values from the same daily 0.8–0.2 in-sample period across different cryptocurrencies. We test the null hypothesis that the difference of the values computed from Logistic and ARIMA models comes from a normal distribution with mean equal to zero and unknown variance. The results show that the difference between these algorithms are statistically significant. Similarly, we also use Wilcoxon signed rank test as an alternative testing method. In Table 17 the results are presented for the signed rank test, which tests the null hypothesis that the difference between different models for each in-sample and out-of-sample period comes from a distribution with zero median. In this way we can check whether the accuracies obtained in different experiments are significantly different from each other for each subperiod. The results obtained from the signed rank test are similar with the results from the t test indicating that statistically significant differences are observed between the algorithms including the ARIMA models.

As a further analysis of the results, we implement the Model Confidence Set (MCS) procedure developed by Hansen et al. (2011). This procedure consists of a sequence of tests which permits to construct a set of superior models, where the null hypothesis of Equal Predictive Ability (EPA) is not rejected at a certain confidence level. The EPA statistic tests is calculated based on a loss function and MCS identifies the subset of models with superior performance accordingly. The advantage of this test is that it takes into account the time series of individual loss functions and not just the overall average over all methods.[7] In our case of sign prediction, we construct a loss function such that it takes the value of zero when the predicted sign is correct and one if the predicted sign is wrong. For each time interval under consideration, we stack the loss functions of each coin vertically in order to form the complete loss function of a particular prediction method. For instance, for the 60 min out-s sub1, we stack the loss functions of size 200 coming from ANN method for each coin to construct the complete loss function of size 2400 for ANN method for that particular time scale. Furthermore, similar to Hansen et al. (2011), we produce the results for 75% confidence level. Table 14 presents the best performing models for the coins from MCS procedure with T_R statistic and 10,000 bootstraps. It is observed that there are only two best performing methods i.e. logistic and random forest at 75% confidence level. Although these methods clearly dominate the others in terms of minimizing the loss function for each time scale, there is no clear distinction between them. Indeed, in almost half of the time scales, six out of thirteen, random forest is the best performing method but in the others, logistic is the best performing one. These results are also consistent with those coming from t test and Wilcoxon test as they show that the methods produce significantly different outcomes from each other.

---

[7] A recent application of MCS test to cryptocurrency markets to determine the drivers of bitcoin volatility can be found in Walther et al. (2019).

Table 14 Model confidence set (MCS) test results at each time scale

	1st best model	2nd best model
Daily 0.8-0.2 out-s	Random forest (1)	
60 min 0.9-0.1 out-s	Logistic (1)	
60 min out-s sub1	Random forest (1)	Logistic (0.77)
60 min out-s sub2	Logistic (1)	
60 min out-s sub3	Logistic (1)	
30 min 0.9-0.1 out-s	Random forest (1)	
30 min out-s sub1	Random forest (1)	
30 min out-s sub2	Random forest (1)	
30 min out-s sub3	Logistic (1)	
15 min 0.9-0.1 out-s	Random forest (1)	Logistic (0.90)
15 min out-s sub1	Logistic (1)	
15 min out-s sub2	Logistic (1)	
15 min out-s sub3	Logistic (1)	Random forest (0.83)

Values in the parenthesis correspond to the p values from MCS with T_R statistic and 10,000 bootstraps at 75% confidence level for the specified loss function

To verify whether the proposed methodology and model can predict the performance of a new dataset, we construct two types of cryptocurrency indices. First, we consider the equal weighted cryptocurrency index where the market index is calculated by the average of scaled prices of each cryptocurrency over the same sample period. Once the average scaled price is obtained for the equal weighted market index, then the exact same features are calculated for this index as given in Table 7. Second, a market capitalization weighted index is utilized where the weights are calculated with respect to the historical average market capitalization of each cryptocurrency. The prediction results for the same algorithms considered are presented for these two types of indices in Table 15. The results for both the equally weighted (EW) and market capitalization weighted (MCW) indices at the daily timescale provide high accuracy for almost all the algorithms in the in-sample and out-of-sample backtesting. Different from the prediction of individual cryptocurrencies, the formation of the index offers smoothed time series. Furthermore, the series is smoother at the daily timescale and less volatile than the higher time frequency of index returns. Therefore, a higher predictive power can be achieved using the same models for the prediction of cryptocurrency indices. Finally, the statistical significance between the alternative algorithms are tested using t test and signed rank test and the results are presented in Table 18. Similar to the previous results, most of the relative differences between the accuracy of alternative models are statistically different from each other.

We also apply MCS procedure to find the best performing methods for both the equally weighted and market capitalization weighted indices in terms of the loss minimization. The results from MCS methodology are presented as superscripts in Table 15. Different from the MCS results for coins, there are now up to five best performing methods at 75% confidence interval for equally weighted index. Among them, SVM is the best performing method for more than half of the time scales and it is followed by ANN for only three of the time scales. For the MCW index, SVM is still the most frequent best performing method. It is also observed that there are not as many as best performing models other than the first best for the MVW index compared to EW index. Again these results also show that there are significant differences between methods in terms of the loss minimization as well.

Table 15 Performance of five classification algorithms, including ARIMA, logistic regression, support vector machines, artificial neural networks, and random forest classifier over equally weighted (EW) and market capitalization weighted (MCW) indices and different time scales

	ARIMA		Logistic		SVM		Random F.		ANN	
	EW	MCW	EW	MCW	EW	MCW	EW	MCW	EW	MCW
Daily 0.8-0.2 in-s	0.48	0.52	0.71	0.64	0.71	0.65	0.96	0.99	1.00	1.00
Daily 0.8-0.2 out-s	0.45	0.42	0.69	0.65	0.74	0.65	0.80	0.86	$0.83^{1(1)}$	$0.85^{1(1)}$
60 min 0.9-0.1 in-s	0.47	0.46	0.54	0.56	0.53	0.56	0.99	0.98	0.72	0.71
60 min 0.9-0.1 out-s	0.48	0.44	0.53	0.55	$0.51^{1(1)}$	$0.54^{1(1)}$	0.85	0.78	0.69	0.66
60 min out-s sub1	0.47	0.43	0.60	$0.60^{1(1)}$	$0.60^{1(1)}$	0.59	0.52	0.47	0.23	0.52
60 min out-s sub2	0.46	0.46	$0.58^{3(0.85)}$	0.53	$0.51^{2(0.96)}$	$0.52^{1(1)}$	0.54	$0.50^{2(0.83)}$	$0.55^{1(1)}$	0.54
60 min out-s sub3	0.44	0.44	$0.52^{2(0.94)}$	$0.55^{2(0.99)}$	$0.54^{1(1)}$	0.55	$0.56^{3(0.80)}$	$0.53^{1(1)}$	0.55	$0.52^{3(0.79)}$
30 min 0.9-0.1 in-s	0.48	0.46	0.53	0.54	0.52	0.54	0.99	0.99	0.65	0.66
30 min 0.9-0.1 out-s	0.49	0.46	0.52	0.53	$0.53^{1(1)}$	$0.53^{1(1)}$	0.85	0.80	0.63	0.62
30 min out-s sub1	$0.54^{1(1)}$	0.53	$0.55^{2(0.99)}$	0.56	$0.56^{3(0.98)}$	0.57	0.51	0.53	$0.50^{4(0.86)}$	$0.51^{1(1)}$
30 min out-s sub2	0.49	0.47	0.53	$0.53^{1(1)}$	$0.50^{1(1)}$	0.52	0.52	0.50	0.51	0.55
30 min out-s sub3	0.47	0.47	$0.52^{1(1)}$	0.50	0.60	$0.51^{1(1)}$	0.57	0.53	0.54	0.55
15 min 0.9-0.1 in-s	0.50	0.47	0.52	0.54	0.51	0.54	0.99	0.99	0.60	0.61
15 min 0.9-0.1 out-s	0.47	0.47	0.52	0.52	$0.52^{1(1)}$	$0.53^{1(1)}$	0.85	0.79	0.58	0.58
15 min out-s sub1	0.49	0.48	$0.54^{1(1)}$	0.55	$0.55^{2(0.76)}$	0.56	0.60	$0.57^{1(1)}$	0.51	0.49
15 min out-s sub2	$0.51^{4(0.89)}$	0.49	$0.53^{3(0.97)}$	0.54	$0.52^{1(1)}$	0.53	0.61	$0.58^{1(1)}$	$0.52^{2(0.99)}$	0.53
15 min out-s sub3	$0.50^{4(0.99)}$	0.48	$0.51^{2(0.99)}$	0.51	$0.48^{3(0.99)}$	0.52	$0.58^{5(0.81)}$	0.53	$0.53^{1(1)}$	$0.54^{1(1)}$

Model confidence set (MCS) test results for equally weighted (EW) and market capitalization weighted (MCW) indices are indicated as the superscripts. The first number in the superscript stands for the order of the model like first best model or second best model etc. and values in the parenthesis correspond to the p values from MCS with T_R statistic and 10,000 bootstraps at 75% confidence level for the specified loss function

5 Conclusion

Cryptocurrency markets attract attention due to the recent advances in their underlying technology and investors consider this as a part of the alternative investments space. With this increasing attention from the investment community, cryptocurrency markets constitute an important asset class both for researchers and traders. In this study, we consider twelve major cryptocurrencies and study their predictability using machine learning classification algorithms over four different time scales, including the daily, 15-,30-, and 60- min returns. Numerical experiments conducted for four different classification algorithms, namely, logistic regression, support vector machines, artificial neural networks, and random forest algorithms demonstrate the predictability of the upward or downward price moves. The best performing and robust models in the prediction of the next day's return is the support vector machines with consistently above 50% fit, low variation across all the products and different timescales, and good generalization ability to different sub-periods consistently. Although the performance of classification algorithms are not uniform over different coins, in many occasions predictive accuracy over 69% is achieved without fine tuning or specifically searching different variations of the features for each coin. This implies that with extra fine tuning of model selection step, the machine learning algorithms and space of features can easily achieve over 70% predictive accuracy. The results indicate that the use of machine learning is promising for short term forecasting of trends in the cryptocurrency markets. Finally, the predictive accuracy is not too volatile across different forecasting horizons. However, due to the higher sample size of observations at the minute level frequencies, more complex models or higher number of features can be adopted at the high frequency of returns to generate models with even better predictive accuracy at shorter time intervals.

Findings have important implications with regards to market efficiency. First, if the cryptocurrency markets were efficient in the weak form, their past price movements would not help us to forecast the future returns. However, our findings imply that applying machine learning tools on historical prices enables one to predict cryptocurrency returns with successful accuracy at the intraday level. Considering the fact that financial markets have significantly evolved over the past decade, driven by new technologies in automated trading and financial infrastructure, our finding is particularly important for algorithmic trading purposes since deciding when to buy and sell cryptocurrencies within the day can successfully be automated with machine learning algorithms. The success of such newly developed algorithms might also cause more high-tech dominant financial markets. Second, even though cryptocurrency markets are far from being able to be regulated, policy makers still need to pay attention to these markets. The reason is that CBOE and CME introduced Bitcoin futures in December 2017, thus, it would be realistic to assume that in the near future, Bitcoin options might also be introduced. Earlier studies show that in the presence of weak form inefficiency, the methods used to price financial derivatives, such as the Black-Scholes model, tend to be heavily biased and may not be useful anymore. Therefore, mispriced derivatives might let investors to take excessive risk (more than they think) in the market and also make it hard to track the systemic risk carried in the overall market.

Appendix

See Tables 16, 17 and 18.

Table 16 *t* test results for comparison of five classification algorithms, including ARIMA, logistic regression, support vector machines, artificial neural networks, and random forest classifier over different time scales

t test	Logistic versus ARIMA	SVM versus ARIMA	ANN versus ARIMA	Random forest versus ARIMA	Logistic versus SVM	Logistic versus ANN	Logistic versus Random forest	SVM versus ANN	SVM versus Random forest	ANN versus Random forest
Daily 0.8–0.2 in-s	0.1511****	0.2267****	0.4338****	0.4987****	-0.0756***	-0.2828****	-0.3476****	-0.2071****	-0.272****	-0.0649**
	(13.68)	(23.84)	(16.44)	(66.35)	(-9.28)	(-10.97)	(-40.9)	(-8.83)	(-32.85)	(-2.86)
Daily 0.8–0.2 out-s	0.1109***	0.1396***	0.0979***	0.1051***	-0.0287	0.0129	0.0057	0.0417	0.0345	-0.0072
	(8.44)	(6.16)	(3.68)	(5.2)	(-1.22)	(0.59)	(0.34)	(1.39)	(1.2)	(-0.22)
60 min 0.9–0.1 in-s	0.0825***	0.1417***	0.2502***	0.5198***	-0.0592***	-0.1677***	-0.4373***	-0.1085***	-0.3781***	-0.2696***
	(19.71)	(31.76)	(39.23)	(219.2)	(-21.84)	(-24.3)	(-196.7)	(-17.11)	(-147.72)	(-43.82)
60 min 0.9–0.1 out-s	0.0759***	0.0661***	0.0489***	0.0466***	0.0099	0.0271**	0.0293**	0.0172*	0.0194**	0.0022
	(5.45)	(5.64)	(5.67)	(4.53)	(1.55)	(2.83)	(3.07)	(2.08)	(2.45)	(0.38)
60 min out-s sub1	0.055***	0.0571***	0.0254**	0.0129	-0.0021	0.0296	0.0421***	0.0317**	0.0442***	0.0125
	(4.61)	(6.02)	(2.39)	(1.28)	(-0.18)	(1.74)	(3.12)	(2.65)	(5.61)	(1.36)
60 min out-s sub2	0.075***	0.0613***	0.0508***	0.0435***	0.0138*	0.0242**	0.0315**	0.0104	0.0177*	0.0073
	(6.78)	(7.96)	(4.22)	(6.14)	(2.18)	(2.37)	(2.75)	(1.14)	(1.96)	(0.59)
60 min out-s sub3	0.08***	0.0778***	0.0528***	0.0503***	0.0022	0.0272**	0.0297**	0.025**	0.0275***	0.0025
	(8)	(8.78)	(6.23)	(6.54)	(0.51)	(2.72)	(2.74)	(2.9)	(3.44)	(0.32)
30 min 0.9–0.1 in-s	0.0525***	0.1054***	0.1713***	0.5036***	-0.0529***	-0.1188***	-0.4511***	-0.0659***	-0.3982***	-0.3323***
	(11.2)	(24.66)	(43.58)	(200.16)	(-26.38)	(-40.15)	(-186.95)	(-35.07)	(-190.08)	(-168.07)
30 min 0.9–0.1 out-s	0.0617***	0.0632***	0.0434***	0.0357***	-0.0015	0.0184***	0.026***	0.0199***	0.0275***	0.0077
	(7.08)	(8.25)	(7.07)	(4.91)	(-0.47)	(3.42)	(6.21)	(4.84)	(6.2)	(1.58)

Table 16 continued

t test	Logistic versus ARIMA	SVM versus ARIMA	ANN versus ARIMA	Random forest versus ARIMA	Logistic versus SVM	Logistic versus ANN	Logistic versus Random forest	SVM versus ANN	SVM versus Random forest	ANN versus Random forest
30 min out-s sub1	0.0454***	0.0538***	0.0354**	0.0200	-0.0083	0.0100	0.0254**	0.0183	0.0338**	0.0154
	(3.84)	(3.78)	(3.04)	(1.53)	(−1.05)	(1)	(2.61)	(1.61)	(2.58)	(1.3)
30 min out-s sub2	0.075***	0.081***	0.0554***	0.0429***	-0.0060	0.0196**	0.0321***	0.0256***	0.0381***	0.0125
	(8.05)	(8.47)	(5.26)	(4.63)	(−0.96)	(2.81)	(4.57)	(3.14)	(4.55)	(1.22)
30 min out-s sub3	0.0692***	0.0629***	0.0451***	0.0415***	0.0063	0.024***	0.0276***	0.0178**	0.0214**	0.0036
	(8.28)	(6.07)	(8.84)	(4.56)	(1.03)	(4.47)	(3.45)	(2.25)	(2.4)	(0.43)
15 min 0.9-0.1 in-s	0.0434***	0.0887***	0.1213***	0.4989***	-0.0453***	-0.0779***	-0.4555***	-0.0327***	-0.4102***	-0.3775***
	(11.11)	(26.44)	(37.39)	(194.37)	(−31.67)	(−51.77)	(−285.29)	(−18.09)	(−245.6)	(−258.75)
15 min 0.9-0.1 out-s	0.0574***	0.0577***	0.0388***	0.0357***	-0.0003	0.0186***	0.0217***	0.0188***	0.022***	0.0031
	(10.46)	(8.29)	(7.26)	(7.6)	(−0.08)	(5.34)	(6.98)	(5.66)	(4.86)	(0.91)
15 min out-s sub1	0.0871***	0.0829***	0.0775***	0.0575***	0.0042	0.0096	0.0296*	0.0054	0.0254**	0.02*
	(6.45)	(9.12)	(7.07)	(5.93)	(0.53)	(0.68)	(2.04)	(0.52)	(2.5)	(1.88)
15 min out-s sub2	0.081***	0.076***	0.0579***	0.0308*	0.0050	0.0231***	0.0502***	0.0181**	0.0452***	0.0271*
	(5.56)	(6.2)	(4.32)	(2.16)	(1.18)	(3.22)	(5.41)	(2.96)	(3.96)	(2.05)
15 min out-s sub3	0.064***	0.0632***	0.0478***	0.0317***	0.0008	0.0163*	0.0324***	0.0154*	0.0315***	0.0161*
	(6.28)	(6.65)	(4.45)	(3.37)	(0.2)	(2.11)	(4.66)	(2.1)	(4.91)	(1.86)

The values in the parentheses are t-statistics. ***, ** and * denote significance at the 1%, 5% and 10% level respectively

Table 17 Wilcoxon signed rank test results for comparison of five classification algorithms, including ARIMA, logistic regression, support vector machines, artificial neural networks, and random forest classifier over different time scales

Signedrank	Logistic versus ARIMA	SVM versus ARIMA	ANN versus ARIMA	Random forest versus ARIMA	Logistic versus SVM	Logistic versus ANN	Logistic versus Random forest	SVM versus ANN	SVM versus Random forest	ANN versus Random forest
Daily 0.8–0.2 in-s	0.1526***	0.2243***	0.4669***	0.4972***	− 0.0692***	− 0.2969***	− 0.3478***	− 0.2301***	− 0.2803***	− 0.0311***
	(78)	(78)	(78)	(78)	(0)	(0)	(0)	(0)	(0)	(0)
Daily 0.8–0.2 out-s	0.1018***	0.1385***	0.1018**	0.1142***	− 0.0259	0.0086	0.0086	0.0259	0.0259	− 0.0172
	(78)	(78)	(69)	(77)	(20)	(32)	(31)	(39)	(38.5)	(33.5)
60 min 0.9–0.1 in-s	0.0818***	0.1433***	0.2506***	0.5195***	− 0.0588***	− 0.1759***	− 0.4353***	− 0.1096***	− 0.3778***	− 0.2663***
	(78)	(78)	(78)	(78)	(0)	(0)	(0)	(0)	(0)	(0)
60 min 0.9–0.1 out-s	0.082***	0.0691***	0.057***	0.0478***	0.0119	0.0238**	0.0371**	0.0112	0.0258**	0.0007
	(76)	(77)	(78)	(75)	(59)	(68)	(68)	(58)	(65)	(41)
60 min out-s sub1	0.0525***	0.0475***	0.0175**	0.0125	0.0000	0.0350	0.05**	0.0375**	0.0525***	0.0150
	(75.5)	(66)	(66.5)	(46)	(39)	(58)	(58.5)	(66.5)	(77)	(55.5)
60 min out-s sub2	0.08***	0.065***	0.0438***	0.04***	0.0163*	0.025**	0.0375***	0.0100	0.02*	0.0138
	(77)	(78)	(75)	(78)	(46)	(65)	(65.5)	(46)	(61.5)	(45.5)
60 min out-s sub3	0.0792***	0.0817***	0.06***	0.055***	0.0017	0.0283**	0.0417**	0.015**	0.0342**	− 0.0042
	(77)	(78)	(77)	(78)	(46)	(68)	(66)	(69.5)	(67)	(42)
30 min 0.9–0.1 in-s	0.0487***	0.1037***	0.1729***	0.5032***	− 0.0537***	− 0.1183***	− 0.4547***	− 0.0649***	− 0.3998***	− 0.3336***
	(78)	(78)	(78)	(78)	(0)	(0)	(0)	(0)	(0)	(0)
30 min 0.9–0.1 out-s	0.068***	0.0699***	0.0455***	0.0398***	0.0007	0.0182***	0.0276***	0.0212***	0.0327***	0.0122*
	(78)	(78)	(77)	(76)	(36)	(71)	(77)	(75)	(77)	(62)
30 min out-s sub1	0.0475***	0.0525***	0.0325**	0.0150	− 0.0175	0.0125	0.0325**	0.0225	0.045**	0.0250
	(73)	(74)	(70.5)	(50)	(16)	(50.5)	(67.5)	(57.5)	(67)	(48.5)

Table 17 continued

Signedrank	Logistic versus ARIMA	SVM versus ARIMA	ANN versus ARIMA	Random forest versus ARIMA	Logistic versus SVM	Logistic versus ANN	Logistic versus Random forest	SVM versus ANN	SVM versus Random forest	ANN versus Random forest
30 min out-s sub2	0.0738*** (78)	0.075*** (78)	0.0538*** (78)	0.0388*** (65)	−0.0075 (29)	0.0213** (68.5)	0.0413*** (75.5)	0.04** (68)	0.0475*** (64.5)	0.0188 (55)
30 min out-s sub3	0.0633*** (78)	0.0667*** (77)	0.0417*** (78)	0.0408*** (75)	0.0000 (34)	0.0183*** (76)	0.0225*** (72.5)	0.0167** (65)	0.01** (65)	0.0042 (42)
15 min 0.9-0.1 in-s	0.0417*** (78)	0.0895*** (78)	0.1195*** (78)	0.4984*** (78)	−0.0453*** (0)	−0.0787*** (0)	−0.4563*** (0)	−0.0338*** (0)	−0.4104*** (0)	−0.377*** (0)
15 min 0.9-0.1 out-s	0.0582*** (78)	0.0594*** (78)	0.0431*** (78)	0.0353*** (78)	−0.0035 (26)	0.0221*** (78)	0.0216*** (78)	0.0183*** (78)	0.0193*** (77)	0.0018 (45)
15 min out-s sub1	0.0925*** (66)	0.0825*** (78)	0.08*** (66)	0.06*** (77)	0.0000 (44)	−0.0025 (45.5)	0.04* (55)	−0.0050 (41)	0.03** (56.5)	0.0075* (62)
15 min out-s sub2	0.0875*** (77)	0.085*** (78)	0.0588*** (78)	0.045* (63)	0.0087 (57.5)	0.0163*** (71)	0.0425*** (78)	0.0188** (68.5)	0.035*** (66)	0.0238* (53.5)
15 min out-s sub3	0.065*** (78)	0.0625*** (78)	0.0408*** (78)	0.0225*** (74)	0.0017 (40.5)	0.0142* (61.5)	0.0292*** (78)	0.015* (61.5)	0.0283*** (77)	0.0075 (57)

The values in the parentheses are signedrank statistics from Wilcoxon signed rank test. ***, ** and * denote significance at the 1%, 5% and 10% level respectively

Table 18 *t* test and Wilcoxon signed rank test results for comparison of five classification algorithms, including ARIMA, logistic regression, support vector machines, artificial neural networks, and random forest classifier over equally weighted (EW) and market capitalization weighted (MCW) indices and different time scales

	Logistic versus ARIMA	SVM versus ARIMA	ANN versus ARIMA	Random forest versus ARIMA	Logistic versus SVM	Logistic versus ANN	Logistic versus Random forest	SVM versus ANN	SVM versus Random forest	ANN versus Random forest
t test										
EW	0.0736***	0.0723***	0.1161**	0.2419***	0.0006	−0.0412	−0.1676***	−0.0418	−0.1682***	−0.1265***
	(4.47)	(3.76)	(2.88)	(5.18)	(0.07)	(−1.21)	(−3.63)	(−1.22)	(−3.56)	(−3.67)
MCW	0.0852***	0.0844***	0.1459***	0.2333***	−0.0006	−0.0612**	−0.1482***	−0.0606**	−0.1476***	−0.0871**
	(6.69)	(6.82)	(4.43)	(4.73)	(−0.29)	(−2.32)	(−3.17)	(−2.33)	(−3.17)	(−2.6)
Signedrank										
EW	0.0491***	0.0491***	0.1069***	0.1237***	0.002	−0.036*	−0.081***	−0.046*	−0.089***	−0.091
	(153)	(149)	(134)	(152)	(68)	(39)	(14)	(37)	(16)	(17)
MCW	0.0705***	0.0682***	0.09***	0.0913***	0	−0.051**	−0.037**	−0.046**	−0.056**	−0.026***
	(153)	(153)	(151)	(153)	(59.5)	(32)	(25.5)	(33)	(28)	(35.5)

The values in the parentheses in the *t* test are statistics from *t* test. The values in the parentheses in Wilcoxon signed rank test are signedrank statistics. ***, ** and * denote significance at the 1%, 5% and 10% level, respectively

References

Achelis, S. B. (1995). *Technical analysis from A to Z* (2nd ed.). New York: McGraw-Hill.

Avdoulas, C., Bekiros, S., & Boubaker, S. (2018). Evolutionary-based return forecasting with nonlinear STAR models: Evidence from the Eurozone peripheral stock markets. *Annals of Operations Research*, *262*, 307–333.

Bariviera, A. F. (2017). The inefficiency of Bitcoin revisited: A dynamic approach. *Economics Letters*, *161*, 1–4.

Bouri, E., Das, M., Gupta, R., & Roubaud, D. (2018c). Spillovers between Bitcoin and other assets during bear and bull markets. *Applied Economics*, *50*, 5935–5949.

Bouri, E., Gupta, R., Lahiani, A., & Shahbaz, M. (2018a). Testing for asymmetric nonlinear short- and long-run relationships between Bitcoin, aggregate commodity and gold prices. *Resources Policy*, *57*, 224–235.

Bouri, E., Gupta, R., Lau, C. K. M., Roubaud, D., & Wang, S. (2018b). Bitcoin and global financial stress: A copula-based approach to dependence and causality in the quantiles. *Quarterly Review of Economics and Finance*, *69*, 297–307.

Bouri, E., Lau, C. K. M., Lucey, B. M., & Roubaud, D. (2019b). Trading volume and the predictability of return and volatility in the cryptocurrency market. *Finance Research Letters*, *29*, 340–346.

Bouri, E., Shahzad, S. J. H., & Roubaud, D. (2019a). Co-explosivity in the cryptocurrency market. *Finance Research Letters*, *29*, 178–183.

Brauneis, A., & Mestel, R. (2018). Price discovery of cryptocurrencies: Bitcoin and beyond. *Economics Letters*, *165*, 58–61.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*, 5–32.

Brock, W., Lakonishok, J., & LeBaron, B. (1992). Simple technical trading rules and the stochastic properties of stock returns. *Journal of Finance*, *47*, 1731–1764.

Cochran, S. J., De Fina, R. H., & Mills, L. O. (1993). International evidence on predictability of stock returns. *Financial Review*, *28*, 159–180.

Cretarola, A., & Figà-Talamanca, G. (2019). Detecting bubbles in Bitcoin price dynamics via market exuberance. *Annals of Operations Research*. https://doi.org/10.1007/s10479-019-03321-z.

El Alaoui, M., Bouri, E., & Roubaud, D. (2019). Bitcoin price–volume: A multifractal cross-correlation approach. *Finance Research Letters*. https://doi.org/10.1016/j.frl.2018.12.011.

Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *Journal of Finance*, *25*, 383–417.

Fama, E. F., & French, K. (1988). Permanent and temporary components of stock prices. *Journal of Political Economy*, *96*, 246–273.

Giudici, P., & Polinesi, G. (2019). Crypto price discovery through correlation networks. *Annals of Operations Research*. https://doi.org/10.1007/s10479-019-03282-3.

Glantz, M., & Kissell, R. (2013). *Multi-asset risk modeling: Techniques for a global economy in an electronic and algorithmic trading era*. Cambridge: Academic Press.

Guresen, E., Kayakutlu, G., & Daim, T. (2011). Using artificial neural network models in stock market index prediction. *Expert Systems with Applications*, *38*, 10389–10397.

Hansen, P. R., Lunde, A., & Nason, J. M. (2011). The model confidence set. *Econometrica*, *79*, 453–497.

Huang, W., Nakamori, Y., & Wang, S.-Y. (2005). Forecasting stock market movement direction with support vector machine. *Computers & Operations Research*, *32*, 2513–2522.

Ince, H., & Trafalis, T. B. (2008). Short term forecasting with support vector machines and application to stock price prediction. *International Journal of General Systems*, *37*, 677–687.

Jamdee, S., & Los, C. A. (2007). Long memory options: LM evidence and simulations. *Research in International Business and Finance*, *21*, 260–280.

Ji, Q., Bouri, E., Roubaud, D., & Kristoufek, L. (2019). Information interdependence among energy, cryptocurrency and major commodity markets. *Energy Economics*, *81*, 1042–1055.

Jiang, Y., Nie, H., & Ruan, W. (2018). Time-varying long-term memory in Bitcoin market. *Finance Research Letters*, *25*, 280–284.

Kara, Y., Acar, M., Boyacioglu, O., & Baykan, K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. *Expert Systems with Applications*, *38*, 5311–5319.

Khaidem, L., Saha, S., & Dey, S. R. (2016). Predicting the direction of stock market prices using random forest. arXiv:1605.00003.

Khuntia, S., & Pattanayak, J. K. (2018). Adaptive market hypothesis and evolving predictability of Bitcoin. *Economics Letters*, *167*, 26–28.

Kim, K.-J. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, *55*, 307–319.

Koutmos, D. (2019). Market risk and Bitcoin returns. *Annals of Operations Research*. https://doi.org/10.1007/s10479-019-03255-6.

Kumar, M., Thenmozhi, M. (2006). *Forecasting stock index movement: A comparison of support vector machines and random forest*. SSRN Working Paper.

Kyaw, N. A., Los, C. A., & Zong, S. (2006). Persistence characteristics of Latin American financial markets. *Journal of Multinational Financial Management*, *16*, 269–290.

Lee, M.-C. (2009). Using support vector machine with a hybrid feature selection method to the stock trend prediction. *Expert Systems with Applications*, *36*, 10896–10904.

Lo, A. W., & Mackinlay, A. C. (1988). Stock market prices do not follow random walks: Evidence from a simple specification test. *Review of Financial Studies*, *1*, 41–66.

Mandelbort, B. (1971). When can price be arbitraged efficiently? A limit to the validity of the random walk and martingale properties. *Review of Economic Statistics*, *53*, 225–236.

Mandelbort, B. (1997). *Fractals and scaling in finance: Discontinuity, concentration, risk*. New York: Springer.

Mensi, W., Lee, Y. J., Al-Yahyaee, K. H., Sensoy, A., & Yoon, S. M. (2019). Intraday downward/upward multifractality and long memory in Bitcoin and Ethereum markets: An asymmetric multifractal detrended fluctuation analysis. *Finance Research Letters*, *31*, 19–25.

Nadarajah, S., & Chu, J. (2017). On the inefficiency of Bitcoin. *Economics Letters*, *150*, 6–9.

Noakes, M. A., & Rajaratnam, K. (2016). Testing market efficiency on the Johannesburg Stock Exchange using the overlapping serial test. *Annals of Operations Research*, *243*, 273–300.

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications*, *42*, 259–268.

Poterba, J., & Summers, L. H. (1988). Mean reversion in stock returns: Evidence and implications. *Journal of Financial Economics*, *22*, 27–60.

Principe, J. C., Euliano, N. R., & Lefebvre, W. C. (1999). *Neural and adaptive systems: Fundamentals through simulations*. New York: Wiley.

Sensoy, A. (2019). The inefficiency of Bitcoin revisited: A high-frequency analysis with alternative cryptocurrencies. *Finance Research Letters*, *28*, 68–73.

Tiwari, A. K., Jana, R., Das, D., & Roubaud, D. (2018). Informational efficiency of Bitcoin—An extension. *Economics Letters*, *163*, 106–109.

Urquhart, A. (2016). The inefficiency of Bitcoin. *Economics Letters*, *148*, 80–82.

Vidal-Tomas, D., & Ibanez, A. (2018). Semi-strong efficiency of Bitcoin. *Finance Research Letters*, *27*, 259–265.

Walther, T., Klein, T., & Bouri, E. (2019). Exogenous drivers of Bitcoin and cryptocurrency volatility—A mixed data sampling approach to forecasting. *Journal of International Financial Markets, Institutions and Money*, *63*, 101133.

Wei, W. C. (2018). Liquidity and market efficiency in cryptocurrencies. *Economics Letters*, *168*, 21–24.