Research Article

Anticipating Cryptocurrency Prices Using Machine Learning

Laura Alessandretti,1 Abeer ElBahrawy,2 Luca Maria Aiello,3 and Andrea Baronchelli2,4

1Technical University of Denmark, 2800 Kgs. Lyngby, Denmark
2City, University of London, Department of Mathematics, London ECIV 0HB, UK
3Nokia Bell Labs, Cambridge CB3 0FA, UK
4UCL Centre for Blockchain Technologies, University College London, UK

Correspondence should be addressed to Andrea Baronchelli; andrea.baronchelli.1@city.ac.uk

Received 29 May 2018; Revised 28 September 2018; Accepted 17 October 2018; Published 4 November 2018

1. Introduction

The popularity of cryptocurrencies has skyrocketed in 2017 due to several consecutive months of superexponential growth of their market capitalization [1], which peaked at more than $800 billions in Jan. 2018. Today, there are more than 1,500 actively traded cryptocurrencies. Between 2.9 and 5.8 millions of private as well as institutional investors are in the different transaction networks, according to a recent survey [2], and access to the market has become easier over time. Major cryptocurrencies can be bought using fiat currency in a number of online exchanges (e.g., Binance [3], Upbit [4], Kraken [5], etc.) and then be used in their turn to buy less popular cryptocurrencies. The volume of daily exchanges is currently superior to $15 billions. Since 2017, over 170 hedge funds specialised in cryptocurrencies have emerged and Bitcoin futures have been launched to address institutional demand for trading and hedging Bitcoin [6].

The market is diverse and provides investors with many different products. Just to mention a few, Bitcoin was expressly designed as a medium of exchange [7, 8]; Dash offers improved services on top of Bitcoin’s feature set, including instantaneous and private transactions [9]; Ethereum is a public, blockchain-based distributed computing platform featuring smart contract (scripting) functionality, and Ether is a cryptocurrency whose blockchain is generated by the Ethereum platform [10]; Ripple is a real-time gross settlement system (RTGS), currency exchange, and remittance network Ripple [11], and IOTA is focused on providing secure communications and payments between agents on the Internet of Things [12].

The emergence of a self-organised market of virtual currencies and/or assets whose value is generated primarily by social consensus [13] has naturally attracted interest from the scientific community [8, 14–30]. Recent results have shown that the long-term properties of the cryptocurrency market have remained stable between 2013 and 2017 and are compatible with a scenario in which investors simply sample the market and allocate their money according to the cryptocurrency’s market shares [1]. While this is true on average, various studies have focused on the analysis and forecasting of price fluctuations, using mostly traditional approaches for financial markets analysis and prediction [31–35].

The success of machine learning techniques for stock markets prediction [36–42] suggests that these methods...
could be effective also in predicting cryptocurrencies prices. However, the application of machine learning algorithms to the cryptocurrency market has been limited so far to the analysis of Bitcoin prices, using random forests [43], Bayesian neural network [44], long short-term memory neural network [45], and other algorithms [32, 46]. These studies were able to anticipate, to different degrees, the price fluctuations of Bitcoin, and revealed that best results were achieved by neural network based algorithms. Deep reinforcement learning was showed to beat the uniform buy and hold strategy [47] in predicting the prices of 12 cryptocurrencies over one-year period [48].

Other attempts to use machine learning to predict the prices of cryptocurrencies other than Bitcoin come from nonacademic sources [49–54]. Most of these analyses focused on a limited number of currencies and did not provide benchmark comparisons for their results.

Here, we test the performance of three models in predicting daily cryptocurrency price for 1,681 currencies. Two of the models are based on gradient boosting decision trees [55] and one is based on long short-term memory (LSTM) recurrent neural networks [56]. In all cases, we build investment portfolios based on the predictions and we compare their performance in terms of return on investment. We find that all of the three models perform better than a baseline ‘simple moving average’ model [57–60] where a currency’s price is predicted as the average price across the preceding days and that the method based on long short-term memory recurrent neural networks systematically yields the best return on investment.

The article is structured as follows: In Materials and Methods we describe the data (see Data Description and Preprocessing), the metrics characterizing cryptocurrencies that are used along the paper (see Metrics), the forecasting algorithms (see Forecasting Algorithms), and the evaluation metrics (see Evaluation). In Results, we present and compare the results obtained with the three forecasting algorithms and the baseline method. In Conclusion, we conclude and discuss results.

2. Materials and Methods

2.1. Data Description and Preprocessing. Cryptocurrency data was extracted from the website Coin Market Cap [61], collecting daily data from 300 exchange markets platforms starting in the period between November 11, 2015, and April 24, 2018. The dataset contains the daily price in US dollars, the market capitalization, and the trading volume of 1,681 cryptocurrencies, where the market capitalization is the product between price and circulating supply, and the volume is the number of coins exchanged in a day. The daily price is computed as the volume weighted average of all prices reported at each market. Figure 1 shows the number of currencies with trading volume larger than \( V_{\text{min}} \) over time, for different values of \( V_{\text{min}} \). In the following sections, we consider that only currencies with daily trading volume larger than \( 10^5 \) USD (United States dollar) can be traded at any given day.

![Figure 1: Number of cryptocurrencies. The cryptocurrencies with volume higher than \( V_{\text{min}} \) as a function of time, for different values of \( V_{\text{min}} \). For visualization purposes, curves are averaged over a rolling window of 10 days.](image)

The profitability of a currency \( c \) over time can be quantified through the return on investment (ROI), measuring the return of an investment made at day \( t \) relative to the cost [62]. The index \( i \) rolls across days and it is included between 0 and 895, with \( t_0 = \text{November 11, 2015} \), and \( t_{895} = \text{April 24, 2018} \). Since we are interested in the short-term performance, we consider the return on investment after 1 day defined as

\[
\text{ROI}(c, t_i) = \frac{\text{price}(c, t_i) - \text{price}(c, t_{i-1})}{\text{price}(c, t_{i-1})}.
\] (1)
In Figure 2, we show the evolution of the ROI over time for Bitcoin (orange line) and on average for currencies whose volume is larger than $V_{\text{min}} = 10^5$ USD (blue line). In both cases, the average return on investment over the period considered is larger than 0, reflecting the overall growth of the market.

2.3. Forecasting Algorithms. We test and compare three supervised methods for short-term price forecasting. The first two methods rely on XGBoost [63], an open-source scalable machine learning system for tree boosting used in a number of winning Kaggle solutions (17/29 in 2015) [64]. The third method is based on the long short-term memory (LSTM) algorithm for recurrent neural networks [56] that have demonstrated to achieve state-of-the-art results in time-series forecasting [65].

Method 1. The first method considers one single regression model to describe the change in price of all currencies (see Figure 3). The model is an ensemble of regression trees built by the XGBoost algorithm. The features of the model are characteristics of a currency between time $t_j-w$ and $t_j-1$ and the target is the ROI of the currency at time $t_j$, where $w$ is a parameter to be determined. The characteristics considered for each currency are price, market capitalization, market share, rank, volume, and ROI (see (1)). The features for the regression are built across the window between $t_j-w$ and $t_j-1$ included (see Figure 3). Specifically, we consider the average, the standard deviation, the median, the last value, and the trend (e.g., the difference between last and first value) of the properties listed above. In the training phase, we include all currencies with volume larger than $10^5$ USD and $t_j$, between $t_j-W_{\text{training}}$ and $t_j-1$. In general, larger training windows do not necessarily lead to better results (see results section), because the market evolves across time. In the prediction phase, we test on the set of existing currencies at day $t_j$. This procedure is repeated for values of $t_j$ included between January 1, 2016, and April 24, 2018.

Method 2. Also the second method relies on XGBoost, but now the algorithm is used to build a different regression model for each currency $c_i$ (see Figure 4). The features of the model for currency $c_i$ are the characteristics of all the currencies in the dataset between $t_j-w$ and $t_j-1$ included and the target is the ROI of $c_i$ at day $t_j$ (i.e., now the algorithm learns to predict the price of the currency $i$ based on the features of all the currencies in the system between $t_j-w$ and $t_j-1$). The features of the model are the same used in Method 1 (e.g., the average, standard, deviation, median, last value, and difference between last and first value of the following

![Figure 2: Return on investment over time.](image-url) The daily return on investment for Bitcoin (orange line) and the average for currencies with volume larger than $V_{\text{min}} = 10^5$ USD (blue line). Their average value across time (dashed lines) is larger than 0. For visualization purposes, curves are averaged over a rolling window of 10 days.

![Figure 3: Schematic description of Method 1.](image-url) The training set is composed of features and target (T) pairs, where features are various characteristics of a currency $c_i$, computed across the $w$ days preceding time $t_j$, and the target $T$ is the price of $c_i$ at $t_j$. The features-target pairs are computed for all currencies $c_i$ and all values of $t_j$ included between $t_j-W_{\text{training}}$ and $t_j-1$. The test set includes features-target pairs for all currencies with trading volume larger than $10^5$ USD at $t_j$, where the target is the price at time $t_j$ and features are computed in the $w$ days preceding $t_j$. 

![Figure 4: Complexity](image-url)
The geometric mean return is defined as
\[ G(t_i) = \sqrt[\tau]{\prod_{j=1}^{t_i} 1 + R(t_j)}, \] (4)
where \( t_i \) corresponds to the total number of days considered. The cumulative return obtained at \( t_i \) after investing and selling on the following day for the whole period is defined as \( G(t_i)^\tau \).

The number of currencies \( n \) to include in a portfolio is chosen at \( t_i \) by optimising either the geometric mean \( G(t_i - 1) \) (geometric mean optimisation) or the Sharpe ratio \( S(t_i - 1) \) (Sharpe ratio optimisation) over the possible choices of \( n \). The same approach is used to choose the parameters of Method 1 (\( w \) and \( W_{\text{training}} \), Method 2 (\( w \) and \( W_{\text{training}} \)), and the baseline method (\( w \)).

3. Results

We predict the price of the currencies at day \( t_i \), for all \( t_i \) included between Jan 1, 2016, and Apr 24, 2018. The analysis considers all currencies whose age is larger than 50 days since their first appearance and whose volume is larger than $100000. To discount for the effect of the overall market movement (i.e., market growth, for most of the considered period), we consider cryptocurrencies prices expressed in BTC (Bitcoin). This implies that Bitcoin is excluded from our analysis.

3.1. Parameter Setting. First, we choose the parameters for each method. Parameters include the number of currencies \( n \) to include in the portfolio as well as the parameters specific to each method. In most cases, at each day \( t_i \) we choose the parameters that maximise either the geometric mean \( G(t_i - 1) \) (geometric mean optimisation) or the Sharpe ratio \( S(t_i - 1) \) (Sharpe ratio optimisation) computed between times 0 and \( t_i \).

Baseline Strategy. We test the performance of the baseline strategy for choices of window \( w \geq 2 \) (the minimal
Method 1. We explore values of the window $w$ in $\{3, 5, 7, 10\}$ days and the training period $W_{\text{training}}$ in $\{5, 10, 20\}$ days (see Appendix Section A). We find that the median value of the selected window $w$ across time is 7 for both the Sharpe ratio and the geometric mean optimisation. The median value of $W_{\text{training}}$ is 5 under geometric mean optimisation and 10 under Sharpe ratio optimisation. The number of currencies included in the portfolio oscillates between 1 and 11 with median at 3, both for the Sharpe ratio (see Appendix Section A) and the geometric mean return (see Appendix Section A) optimisation.

Method 2. We explore values of the window $w$ in $\{3, 5, 7, 10\}$ days and the training period $W_{\text{training}}$ in $\{5, 10, 20\}$ days (see Appendix, Figure 10). The median value of the selected window $w$ across time is 3 for both the Sharpe ratio and the geometric mean optimisation. The median value of $W_{\text{training}}$ is 10 under geometric mean and Sharpe ratio optimisation. The number of currencies included has median at 15 for the Sharpe ratio and 9 for the geometric mean return (see Appendix Section A) optimisation.

Method 3. The LSTM has three parameters: The number of epochs, or complete passes through the dataset during the training phase; the number of neurons in the neural network, and the length of the window $w$. These parameters are chosen by optimising the price prediction of three currencies (Bitcoin, Ripple, and Ethereum) that have on average the largest market share across time (excluding Bitcoin Cash that is a fork of Bitcoin). Results (see Appendix Section A) reveal that, in the range of parameters explored, the best results are achieved for $w = 50$. Results are not particularly affected by the choice of the number of neurons nor the number of epochs. We choose 1 neuron and 1000 epochs since the larger these two parameters, the larger the computational time. The number of currencies to include in the portfolio is optimised over time by maximising the geometric mean return (see Appendix Section A) and the Sharpe ratio (see Appendix Section A). In both cases the median number of currencies included is 1.

3.2. Cumulative Return. In Figure 5, we show the cumulative return obtained using the 4 methods. The cumulative returns achieved on April 24 under the Sharpe ratio optimisation are $\sim 65$ BTC (Baseline), $\sim 1.1 \cdot 10^3$ BTC (Method 1), $\sim 95$ BTC (Method 2), $\sim 1.2 \cdot 10^3$ BTC (Method 3). Under geometric mean optimisation we obtain $\sim 25$ BTC (Baseline), $\sim 19 \cdot 10^3$ BTC (Method 1), $\sim 1.25$ BTC (Method 2), $\sim 3.6 \cdot 10^3$ BTC (Method 3). The cumulative returns obtained in USD are higher (see Appendix Section D). This is expected, since the Bitcoin price has increased during the period considered. While some of these figures appear exaggerated, it is worth noticing that (i) we run a theoretical exercise assuming that the availability of Bitcoin is not limited and (ii) under this assumption the upper bound to our strategy, corresponding to investing every day in the most performing currency results in a total cumulative return of $6 \cdot 10^{123}$ BTC (see Appendix Section B). We consider also the more realistic scenario of investors paying a transaction fee when selling and buying currencies (see Appendix Section C). In most exchange markets, the fee is typically included between 0.1% and 0.5% of the traded amount [66]. For fees up to 0.2%, all the investment methods presented above lead, on average, to positive returns over the entire period (see Appendix Section C). The best performing method, Method 3, achieves positive
The geometric mean return obtained within different periods of time. The geometric mean return computed between time “start” and “end” using the Sharpe ratio optimisation for the baseline (a), Method 1 (b), Method 2 (c), and Method 3 (d). Note that, for visualization purposes, the figure shows the translated geometric mean return $G^{-1}$. Shades of red refer to negative returns and shades of blue to positive ones (see colour bar).

Figure 6: Geometric mean return obtained within different periods of time. The geometric mean return computed between time “start” and “end” using the Sharpe ratio optimisation for the baseline (a), Method 1 (b), Method 2 (c), and Method 3 (d). Note that, for visualization purposes, the figure shows the translated geometric mean return $G^{-1}$. Shades of red refer to negative returns and shades of blue to positive ones (see colour bar).

The cumulative return in Figure 5 is obtained by investing between January 1st, 2016 and April 24th, 2018. We investigate the overall performance of the various methods by looking at the geometric mean return obtained in different periods (see Figure 6). Results presented in Figure 6 are obtained under Sharpe ratio optimisation for the baseline (Figure 6(a)), Method 1 (Figure 6(b)), Method 2 (Figure 6(c)), and Method 3 (Figure 6(d)). Note that, while in this case the investment can start after January 1, 2016, we optimised the parameters by using data from that date on in all cases. Results are considerably better than those achieved using geometric mean return optimisation (see Appendix Section E). Finally, we observe that better performance is achieved when the algorithms consider prices in Bitcoin rather than USD (see Appendix Section D).
3.3. Feature Importance. In Figure 7, we illustrate the relative importance of the various features in Method 1 and Method 2. For Method 1, we show the average feature importance. For Method 2, we show the average feature importance for two sample currencies: Ethereum and Ripple.

3.4. Portfolio Composition. The 10 most selected currencies under Sharpe ratio optimisation are the following:

**Baseline.** Factom (91 days), E-Dinar Coin (89 days), Ripple (76 days), Ethereum (71 days), Steem (70 days), Lisk (70 days), MaidSafeCoin (69 days), Monero (58 days), BitShares (55 days), EDRCoin (52 days).

**Method 1.** Ethereum (154 days), Dash (128 days), Monero (111 days), Factom (104 days), Ripple (94 days), Litecoin (93 days), Dogecoin (92 days), Maid Safe Coin (86 days), BitShares (73 days), Tether (59 days)

---

**Figure 7: Feature importance for Methods 1 and 2.** (a) The average importance of each feature for the XGBoost regression model of Method 1. Results are shown for $w = 7$ and $W_{training} = 10$. (b, c) Examples of average feature importance for the XGBoost regression model developed in Method 2. Results are shown for $w = 3$, $W_{training} = 10$, for Ethereum (b) and Ripple (c). For visualization purposes, we show only the top features.
Method 2. Ethereum (63 days), Monero (61 days), Factom (51 days), Ripple (42 days), Dash (40 days), Maid Safe Coin (40 days), Siacoin (30 days), NEM (26 days), NXT (26 days), Steem (23 days).

Method 3. Factom (48 days), Monero (46 days), Ethereum (39 days), Lisk (36 days), Maid Safe Coin (32 days), E-Dinar Coin (32 days), BitShares (26 days), B3 Coin (26 days), Dash (25 days), Cryptonite (22 days).

4. Conclusion

We tested the performance of three forecasting models on daily cryptocurrency prices for 1,681 currencies. Two of them (Method 1 and Method 2) were based on gradient boosting decision trees and one is based on long short-term memory recurrent neural networks (Method 3). In Method 1, the same model was used to predict the return on investment of all currencies; in Method 2, we built a different model for each currency that uses information on the behaviour of the whole market to make a prediction on that single currency; in Method 3, we used a different model for each currency, where the prediction is based on previous prices of the currency.

We built investment portfolios based on the predictions of the different method and compared their performance with that of a baseline represented by the well-known simple moving average strategy. The parameters of each model were optimised for all but Method 3 on a daily basis, based on the outcome of each parameters choice in previous times. We used two evaluation metrics used for parameter optimisation: The geometric mean return and the Sharpe ratio. To discount the effect of the overall market growth, cryptocurrencies prices were expressed in Bitcoin. All strategies produced profit (expressed in Bitcoin) over the entire considered period and for a large set of shorter trading periods (different combinations of start and end dates for the trading activity), also when transaction fees up to 0.2% are considered.

The three methods performed better than the baseline strategy when the investment strategy was ran over the whole period considered. The optimisation of parameters based on the Sharpe ratio achieved larger returns. Methods based on gradient boosting decision trees (Methods 1 and 2) worked best when predictions were based on short-term windows of 5/10 days, suggesting they exploit well mostly short-term dependencies. Instead, LSTM recurrent neural networks worked best when predictions were based on ~50 days of data, since they are able to capture also long-term dependencies and are very stable against price volatility. They allowed making profit also if transaction fees up to 1% are considered. Methods based on gradient boosting decision trees allow better interpreting results. We found that the prices and the returns of a currency in the last few days preceding the prediction were leading factors to anticipate its behaviour. Among the two methods based on random forests, the one considering a different model for each currency performed best (Method 2). Finally, it is worth noting that the three methods proposed perform better when predictions are based on prices in Bitcoin rather than prices in USD. This suggests that forecasting simultaneously the overall cryptocurrency market trend and the developments of individual currencies is more challenging than forecasting the latter alone.

It is important to stress that our study has limitations. First, we did not attempt to exploit the existence of different prices on different exchanges, the consideration of which could open the way to significantly higher returns on investment. Second, we ignored intraday price fluctuations and considered an average daily price. Finally, and crucially, we run a theoretical test in which the available supply of Bitcoin is unlimited and none of our trades influence the market. Notwithstanding these simplifying assumptions, the methods we presented were systematically and consistently able to identify outperforming currencies. Extending the current analysis by considering these and other elements of the market is a direction for future work.

A different yet promising approach to the study cryptocurrencies consists in quantifying the impact of public opinion, as measured through social media traces, on the market behaviour, in the same spirit in which this was done for the stock market [67]. While it was shown that social media traces can be also effective predictors of Bitcoin [68–74] and other currencies [75] price fluctuations, our knowledge of their effects on the whole cryptocurrency market remain limited and is an interesting direction for future work.

Appendix

A. Parameter Optimisation

In Figure 8, we show the optimisation of the parameters $w$ (a, c) and $n$ (b, d) for the baseline strategy. In Figure 9, we show the optimisation of the parameters $w$ (a, d), $W_{\text{training}}$ (b, e), and $n$ (c, f) for Method 1. In Figure 10, we show the optimisation of the parameters $w$ (a, d), $W_{\text{training}}$ (b, e), and $n$ (c, f) for Method 2. In Figure 11, we show the median squared error obtained under different training window choices (a), number of epochs (b) and number of neurons (c), for Ethereum, Bitcoin and Ripple. In Figure 12, we show the optimisation of the parameter $n$ (c, f) for Method 3.

B. Return under Full Knowledge of the Market Evolution

In Figure 13, we show the cumulative return obtained by investing every day in the top currency, supposing one knows the prices of currencies on the following day.

C. Return Obtained Paying Transaction Fees

In this section, we present the results obtained including transaction fees between 0.1% and 1% [66]. In general, one can not trade a given currency with any given other. Hence, we consider that each day we trade twice: We sell altcoins to buy Bitcoin, and we buy new altcoins using Bitcoin. The mean return obtained between Jan. 2016 and Apr. 2018 is larger than.
**Figure 8:** Baseline strategy: parameters optimisation. The sliding window $w$ (a, c) and the number of currencies $n$ (b, d) chosen over time under the geometric mean (a, b) and the Sharpe ratio optimisation (c, d). Analyses are performed considering prices in BTC.

**Figure 9:** Method 1: parameters optimisation. The sliding window $w$ (a, d), the training window $W_{training}$ (b, e), and the number of currencies $n$ (c, f) chosen over time under the geometric mean (a, b, c) and the Sharpe ratio optimisation (d, e, f). Analyses are performed considering prices in BTC.
Figure 10: Method 2: parameters optimisation. The sliding window \( w \) (a, d), the training window \( W_{\text{training}} \) (b, e), and the number of currencies \( n \) (c, f) chosen over time under the geometric mean (a, b, c) and the Sharpe ratio optimisation (d, e, f). Analyses are performed considering prices in BTC.

Figure 11: Method 3: parameters optimisation. The median squared error of the ROI as a function of the window size (a), the number of epochs (b), and the number of neurons (c). Results are shown considering prices in Bitcoin.
Table 1: **Daily geometric mean return for different transaction fees.** Results are obtained considering the period between Jan. 2016 and Apr. 2018.

<table>
<thead>
<tr>
<th></th>
<th>no fee</th>
<th>0.1%</th>
<th>0.2%</th>
<th>0.3%</th>
<th>0.5%</th>
<th>1%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>1.005</td>
<td>1.003</td>
<td>1.001</td>
<td>0.999</td>
<td>0.995</td>
<td>0.985</td>
</tr>
<tr>
<td>Method 1</td>
<td>1.008</td>
<td>1.006</td>
<td>1.004</td>
<td>1.002</td>
<td>0.998</td>
<td>0.988</td>
</tr>
<tr>
<td>Method 2</td>
<td>1.005</td>
<td>1.003</td>
<td>1.001</td>
<td>0.999</td>
<td>0.995</td>
<td>0.985</td>
</tr>
<tr>
<td>Method 3</td>
<td>1.025</td>
<td>1.023</td>
<td>1.021</td>
<td>1.019</td>
<td>1.015</td>
<td>1.005</td>
</tr>
</tbody>
</table>

**Figure 12:** **Method 3: parameters optimisation.** The number of currencies $n$ chosen over time under the geometric mean (a) and the Sharpe ratio optimisation (b). Analyses are performed considering prices in BTC.

**Figure 13:** **Upper bound for the cumulative return.** The cumulative return obtained by investing every day in the currency with highest return on the following day (black line). The cumulative return obtained with the baseline (blue line), Method 1 (orange line), Method 2 (green line), and Method 3 (red line). Results are shown in Bitcoin.

1 for all methods, for fees up to 0.2% (see Table 1). In this period, Method 3 achieves positive returns for fees up to 1%. The returns obtained with a 0.1% (see Figure 14) and 0.2% (see Figure 15) fee during arbitrary periods confirm that, in general, one obtains positive gains with our methods if fees are small enough.

### D. Results in USD

In this section, we show results obtained considering prices in USD. The price of Bitcoin in USD has considerably increased in the period considered. Hence, gains in USD (Figure 16) are higher than those in Bitcoin (Figure 5). Note that, in Figure 16, we have made predictions and computed portfolios considering prices in Bitcoin. Then, gains have been converted to USD (without transaction fees). In Table 2, we show instead the gains obtained running predictions considering directly all prices in USD. We find that, in most cases, better results are obtained from prices in BTC.

### E. Geometric Mean Optimisation

In Figure 17, we show the geometric mean return obtained by between two arbitrary points in time under geometric mean return optimisation for the baseline (Figure 17(a)), Method 1
Table 2: Geometric mean returns in USD. Results are obtained for the various methods by running the algorithms considering prices in BTC (left column) and USD (right column).

<table>
<thead>
<tr>
<th>Method</th>
<th>Geometric mean in USD (from BTC prices)</th>
<th>Geometric mean in USD (from USD prices)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>1.0086</td>
<td>1.0141</td>
</tr>
<tr>
<td>Method 1</td>
<td>1.0121</td>
<td>1.0085</td>
</tr>
<tr>
<td>Method 2</td>
<td>1.0091</td>
<td>1.0086</td>
</tr>
<tr>
<td>Method 3</td>
<td>1.0289</td>
<td>1.0134</td>
</tr>
</tbody>
</table>

Figure 14: Daily geometric mean return obtained under transaction fees of 0.1%. The geometric mean return computed between time "start" and "end" using the Sharpe ratio optimisation for the baseline (a), Method 1 (b), Method 2 (c), and Method 3 (d). Note that, for visualization purposes, the figure shows the translated geometric mean return G-1. Shades of red refer to negative returns and shades of blue to positive ones (see colour bar).
Figure 15: **Daily geometric mean return obtained under transaction fees of 0.2%**. The geometric mean return computed between time “start” and “end” using the Sharpe ratio optimisation for the baseline (a), Method 1 (b), Method 2 (c), and Method 3 (d). Note that, for visualization purposes, the figure shows the translated geometric mean return $G^{-1}$. Shades of red refer to negative returns and shades of blue to positive ones (see colour bar).

Figure 16: **Cumulative returns in USD**. The cumulative returns obtained under the Sharpe ratio optimisation (a) and the geometric mean optimisation (b) for the baseline (blue line), Method 1 (orange line), Method 2 (green line), and Method 3 (red line). Analyses are performed considering prices in BTC.
Figure 17: Geometric mean return obtained within different periods of time. The geometric mean return computed between time “start” and “end” using the Sharpe ratio optimisation for the baseline (a), Method 1 (b), Method 2 (c), and Method 3 (d). Note that, for visualization purposes, the figure shows the translated geometric mean return G-1. Shades of red refer to negative returns and shades of blue to positive ones (see colour bar).

(Figure 17(b)), Method 2 (Figure 17(c)), and Method 3 (Figure 17(d)).

Data Availability
The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest
The authors declare that they have no conflicts of interest.

References


P. Ceruleo, Bitcoin: a rival to fiat money or a speculative financial asset?, 2014.


16 Complexity


