Journal of Computational and Applied Mathematics 365 (2020) 112395

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Bitcoin price prediction using machine learning: An approach to sample dimension engineering

Zheshi Chen *, Chunhong Li, Wenjun Sun

School of Management, Harbin Institute of Technology, Harbin, China

ARTICLE INFO

Article history:
Received 10 March 2019
Received in revised form 7 July 2019

Keywords:
Sample dimension engineering
Occam's Razor principle
Bitcoin price prediction
Machine learning algorithms

ABSTRACT

After the boom and bust of cryptocurrencies' prices in recent years, Bitcoin has been increasingly regarded as an investment asset. Because of its highly volatile nature, there is a need for good predictions on which to base investment decisions. Although existing studies have leveraged machine learning for more accurate Bitcoin price prediction, few have focused on the feasibility of applying different modeling techniques to samples with different data structures and dimensional features. To predict Bitcoin price at different frequencies using machine learning techniques, we first classify Bitcoin price by daily price and high-frequency price. A set of high-dimension features including property and network, trading and market, attention and gold spot price are used for Bitcoin daily price prediction, while the basic trading features acquired from a cryptocurrency exchange are used for 5-minute interval price prediction. Statistical methods including Logistic Regression and Linear Discriminant Analysis for Bitcoin daily price prediction with high-dimensional features achieve an accuracy of 66%, outperforming more complicated machine learning algorithms. Compared with benchmark results for daily price prediction, we achieve a better performance, with the highest accuracies of the statistical methods and machine learning algorithms of 66% and 65.3%, respectively. Machine learning models including Random Forest, XGBoost, Quadratic Discriminant Analysis, Support Vector Machine and Long Short-term Memory for Bitcoin 5-minute interval price prediction are superior to statistical methods, with accuracy reaching 67.2%. Our investigation of Bitcoin price prediction can be considered a pilot study of the importance of the sample dimension in machine learning techniques.

1. Introduction

Bitcoin, invented in 2008 to solve the inherent weakness of the trust-based model of transactions and initially defined as a purely peer-to-peer electronic cash system [1], has become an asset or commodity-like product traded in more than 16,000 markets around the world.[1] Although proponents hold that one of Bitcoin's important application is to take the place of fiat currency, the true nature of Bitcoin remains a vexing problem. Investors do not treat Bitcoin as a currency according to the criteria used by economists; instead, they regard Bitcoin as a speculative investment similar to the Internet stocks of the last century [2]. Before Bitcoin disrupted existing payment and monetary systems, its several-year trading and increasing popularity attracted attention from across society, including from policymakers, and the peak of Bitcoin's market capitalization in 2017 reached 300 billion US dollars, almost equal to that of Amazon in 2016.

* Corresponding author.
E-mail addresses: chenjessie08@163.com (Z. Chen), lichunhong2010@163.com (C. Li), wjsun@hit.edu.cn (W. Sun).

[1] CoinMarketCap https://coinmarketcap.com/.

https://doi.org/10.1016/j.cam.2019.112395
0377-0427/

Heated discussions have arisen in response to two key questions: Why does Bitcoin have value? What determines the value of Bitcoin? In the area of financial innovation, Bitcoin's value reflects the confidence of investors in cryptocurrency [3]. Therefore, most previous studies concentrate on the determinants or formation of Bitcoin's price. Price fluctuations due to the inherent volatility of Bitcoin have plagued investors since it began to be traded. It is also important to be able to predict Bitcoin price changes. Stock market prediction has grown over decades using daily data and accessible high-frequency data [4]. However, research on how to predict Bitcoin price is still lacking. Previous studies have predicted Bitcoin price in two ways: empirical analysis and analysis of robust machine learning algorithms. Machine learning algorithms have been widely applied to make accurate predictions in many areas, including product manufacturing [5–8] and finance [9,10]. By learning the details of past instances, machine learning programs and models can be produced that make predictions based on training data. Such algorithms can be replicated for the Bitcoin market, even in the world of cryptocurrency, due to the higher liquidity and volatility caused by the T+0 trading rules.[2]

A natural question arises when predicting the Bitcoin price using machine learning algorithms: What features should be taken into account? Though more methods about feature selection [11,12] and measurements [13,14] are leveraged, previous related works have depended on the researchers' domain knowledge [4,15–17] and lack a comprehensive consideration of feature dimensions. We address this problem by integrating the conclusions of recent empirical works by researchers with a deep understanding of factors influencing the Bitcoin price. Specifically, we leverage Google Trends search volume index and Baidu media search volume, important measures of investor attention and media hype that reflect the sentiment in the highly speculative cryptocurrency market. As discussed in relation with the differences between the properties of Bitcoin and gold, both of them have proved useful and can be substituted for each other [18]. Hence, we additionally include the gold spot when considering features of the Bitcoin market. Integrating gold spot price with regular features such as property, network, trading and market in the machine learning algorithm, we develop higher-dimensional features and avoid the problem of simplifying Bitcoin price prediction.

There is a need to find a method that can accurately use machine learning algorithms to predict Bitcoin price. As Bitcoin lacks seasonality, machine learning models are applicable and useful. Various popular machine learning algorithms, including recurrent neural network, long short-term memory, support vector machines, and random forest models have therefore been implemented in previous studies. However, previous works simply put data into models without distinguishing data frequency or sample size. Different-frequency data have different structures, and the simple copying machine learning algorithms will lead to errors such as overfitting due to complicated methods.

Our paper addresses leveraging appropriate machine learning techniques to engineer sample dimensions for Bitcoin price prediction. Inspired by the principle of Occam's razor and the characteristics of our datasets, we tackle the problem as follows. First, the prediction sample is divided into daily intervals with small sample size and 5-minute intervals with a big sample size. Second, we conduct the features engineering: select high-dimension features for daily price and few features for 5-minute interval trading data respectively. Third, we conduct simple statistical models including Logistic Regression and Linear Discriminant Analysis and the more complicated machine learning models including Random Forest, XGBoost, Quadratic Discriminant Analysis, Support Vector Machine and Long Short-term Memory. Fourth, we adopt the simple statistical methods to predicting Bitcoin daily price with high-dimensional features to avoid overfitting. Meanwhile, the machine learning models are leveraged in high-frequency price few features. Fig. 1 shows the overview of our research framework.

Our paper makes observations in two ways. One is to extend the feature dimensions, and the other is to evaluate different machine learning techniques for solving problems of multiple frequency Bitcoin prices. The study makes the following contributions. (1) To the best of our knowledge, we are at the forefront of establishing higher dimensional features for problems of Bitcoin daily price prediction by integrating investor attention, media hype and XAU gold spot features with common and traditional features such as network and market. (2) We address the importance of the sample dimension by classifying Bitcoin price data by interval. The real-time 5-minute interval trading data acquired from the top cryptocurrency exchange is high-frequency and large scale, and the aggregated Bitcoin daily price obtained from CoinMarketCap is low-frequency and small scale. Hence, the problem of Bitcoin price prediction is addressed from a broad perspective. (3) To find appropriately complex models and meet the requirement of accuracy, we evaluate different machine learning techniques using problems of multiple frequency Bitcoin price. Specifically, we lower the complexity of algorithms for low-frequency daily price prediction with higher-dimension features and apply more complicated models for high-frequency price prediction with a few features. The results show that simple statistical methods outperform machine learning models for daily Bitcoin price prediction while more complicated models should be adopted for high-frequency Bitcoin price prediction. We envision this study as a pilot for dealing with datasets with different scales and intervals, which can shed light on other industrial prediction problems in the context of machine learning.

The remainder of the paper is organized as follows. Section 2 presents a literature review. The problem statement and methodology, including features engineering, are described in Section 3. Section 4 describes the experiments, algorithms, parameter settings and model configuration. Evaluations of two Bitcoin price datasets are presented in Section 5. A discussion and conclusions are presented in Section 6.

[2] $T+0$ refers to cryptocurrency transactions. The T stands for transaction date, which is the day the transaction takes place. The number 0 denotes how many days after the transaction date the settlement or the transfer of money and cryptocurrency ownership takes place.

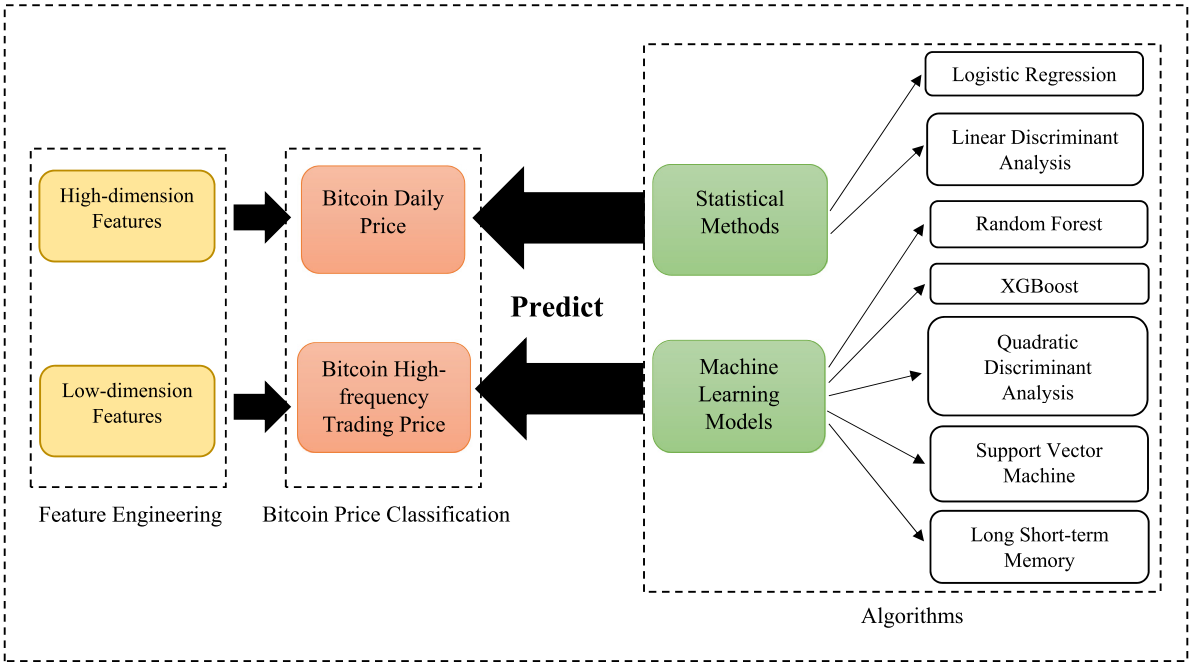

Fig. 1. Overview of the research framework.

2. Related work

When considering the problem of Bitcoin price formation, the literature mostly consists of empirical works to analyze the determinants. Ladislav Kristoufek [19] first regarded the Bitcoin market as comprising purely speculative traders with no fundamentalists, and evaluated the connection between Bitcoin and search volume of Wikipedia and Google Trends. His results showed that the relationship between Bitcoin price and search queries had a pronounced asymmetry, suggesting that speculation and trend-chasing drive the price dynamics of Bitcoin in the cryptocurrency market. Later, Kristoufek [20] used wavelet coherence analysis to search for the potential drivers of Bitcoin price, including fundamental source, speculative and technical drivers and revealed the unique properties of Bitcoin as both a standard financial asset and a speculative one. Adam Hayes [21] examined 66 "coins" using cross-sectional empirical data to identify what factors drive the value of cryptocurrencies in the technical area and noted three main drivers: the aggregate computational power used in mining for units of cryptocurrency, the rate of unit production and the cryptologic algorithm used for the protocol. Implementing time-series analytical mechanisms with daily Bitcoin data, Pavel et al. [22] conducted an empirical analysis on the price determinants of Bitcoin within the framework of a Barro [23] model. Their results revealed that among the traditional determinants and factors specific to digital currencies, market forces and attractiveness to investors drive Bitcoin price most strongly, disproving that macro-level financial developments are the drivers of Bitcoin over a long period. Using a non-parametric causality-in-quantiles test, Balcilar et al. revealed that volume can be a predictor of Bitcoin returns, suggesting the importance of modeling nonlinearity and accounting for tail behavior [24].

However, irrational factors such as sentiment have more favored in empirical research on the Bitcoin market [18]. Jaroslav Bukovina [25] evaluated sentiment as an influencing factor of Bitcoin volatility, finding that the explanatory power of sentiment has a positive relationship with volatility, especially when Bitcoin trades in a bubble stage. Kim et al. [26] analyzed user comments in online cryptocurrency communities for Bitcoin, Ethereum and Ripple to predict price fluctuations and transaction numbers. Implementing a Granger causality model and crawling data achieved an accuracy of 79.57% and price fluctuations and transaction number were significantly associated with positive topics, comments, and replies. Mai at al. [3] examined the dynamic interactions between Bitcoin value and social media based on information system and finance literature. They reported that social media can be an important predictor of Bitcoin price, noting that 5% of active users contribute most of the messages, which is consistent with the silent majority hypothesis.

Thus far, empirical studies do not demonstrate a clear advantage for the emerging techniques of using machine learning algorithms to predict the Bitcoin price, and research in this area is insufficient. Shah et al. [27] utilized a "latent source model" Bayesian regression created by Chen et al. [28], which is designed to leverage binary classification to predict Bitcoin price variations. Using a support vector machine algorithm, Georgoula et al. [29] examined the relationship between Bitcoin price and determinants including economic variables, technological factors, and sentimenFt. Greaves et al. [30] implemented machine learning algorithms to analyze the influence of network features on Bitcoin price and obtained an

accuracy of around 55%. Madan et al. [4] applied machine learning algorithms to predict Bitcoin price with an accuracy of 98.7% for daily price and 50%–55% for high-frequency price. McNally et al. [15] compared the accuracy of recurrent neural network (RNN), long short-term memory (LSTM) and autoregressive integrated moving average (ARIMA) models for predicting Bitcoin price and showed that LSTM achieves the highest accuracy (52%).

The Occam's razor principle suggests that assuming all other criteria are equal, the simplest model can be used when leveraging machine learning [31]. Evidence and arguments have been put forward against this proposition [32]. Domingos [33] stated as interpreted in Occam's razor that given the same training set error, simpler models should be favored because the lower generation error is false. Proponents of Domingos' interpretation proved with an empirical analysis that the risk of overfitting is due to the number of models rather than the models' complexity [32]. Although the utility of Occam's razor remains controversial in modern science and in machine learning, in general, it is supported. Questions arise about whether we should apply simple models under any conditions for predictions and whether relatively simple statistical methods are capable of predicting Bitcoin price more accurately. In practice, the main solution to most prediction problems is to implement high VC-dimension models [34]. There are two usual paradigms: statistical methods with massive features and complicated models with few features. Occam's razor has been extended to machine learning with datasets of high dimension and a large number of features. Langford and Blum [35] demonstrated new adaptive bounds that can be used to make machine learning algorithms self-bounding in the style of Freund [36]. Ebrahmpour et al. [37] took Occam's razor into account in feature subset selection to release high-dimensional datasets from computational search methods arranged by importance and fundamental concept. Based on Occam's razor theory, Zhenin et al. examined whether the complexity inherent in scoring functions for ligand affinity prediction from docking simulation is justified [38]. Wang [39] explained the inadequacy of adopting traditional machine learning problem formulations according to Occam's razor and suggested that learning from data can be an alternative [39].

3. Problem and methodology

3.1. Problem statement

The prediction of Bitcoin price using machine learning techniques is an important problem. Many existing works simply focus on higher accuracy without considering the sample dimension. We are the first to conduct dimension engineering on Bitcoin price granularity and then leverage machine learning. We develop a binary classification algorithm to predict the sign change of Bitcoin price, which is easier for traders to make decisions and follow.

Below, we present our Bitcoin price prediction problem, starting with the corresponding notations. Let t denote the index of time and v denote the index of features. Let g denote the *granularity* of Bitcoin price, where the term granularity here means the time interval of the time series, satisfying $g \in \mathbf{N}^+$. In this study, we investigate two cases, under granularity of five minutes and one day, i.e., $g \in \{5\,\text{min}, 60 \times 24\,\text{min}\}$.[3] Let the current time be c. Let the time to be predicted be $c + h$, where h denotes the prediction length. We consider that

$$\begin{aligned} \mathbf{X}_g &= \left[x_t^v\right] \\ \mathbf{Y}_g &= \{y_t\}_g \end{aligned} \tag{1}$$

where $\mathbf{X}_g$ denotes the matrix of training samples under granularity g, $x_t^v \in \mathbf{R}$ denotes the value of the vth feature at time t, $\mathbf{Y}_g$ denotes the vector of labels, $y \in \{-1, 1\}$ denotes the value of the label at time t. Here $y_t = -1$ indicates that the price drops, while $y_t = 1$ indicates that the price increases. L_g denotes the prediction error, e.g.,

$$L_g = \sum_{h \in \mathrm{H}} \left\| y_{c+h,g} - \hat{y}_{c+h,g} \right\|_2^2 . \tag{2}$$

A Granularity-aware Bitcoin price prediction problem (GBP)

Given the following training samples

$$\begin{aligned} \mathbf{X}_g &= \left[x_t^v\right]_g \\ \mathbf{Y}_g &= \{y_t\}_g \end{aligned} \tag{3}$$

where $g \in \{5\,\text{min}, 60 \times 24\,\text{min}\}$ with time indexes $t \in [0, c]$, and $x_t^v \in \mathbf{R}$, $y \in \{-1, 1\}$, our objective is to minimize the prediction error L_g, i.e.,

$$\min L_g = \sum \left\| y_{c+h,g} - \hat{y}_{c+h,g} \right\|_2^2 . \tag{4}$$

We present two versions of **GBP**. First, we study a scenario in which we only focus on daily price. We call this problem **A Granularity-aware Bitcoin price prediction under daily data (GBP-D)**. Second, we study a scenario in which we only consider the 5-minute interval price. This problem is called **A Granularity-aware Bitcoin price prediction under minute**

[3] In this paper, we investigate the most common cases, daily and 5 min. Other granularities are out of the scope of this paper but could be investigated in future works.

data (GBP-M). In this paper, we assume that daily features in GBP have little impact on the 5-minute interval Bitcoin price and vice versa. Therefore, the samples, features and models in the GBP-D and GBP-M are studied in a separately.

Problem GBP-D

Given the training samples

$$\begin{aligned} \mathbf{X}_g &= \left[x_t^v\right]_g \\ \mathbf{Y}_g &= \{y_t\}_g \end{aligned} \tag{5}$$

where $g = 60 \times 24$ min, with time indexes $t \in [0, c]$, $x_t^v \in \mathbf{R}$, and $y \in \{-1, 1\}$, our objective is to minimize the prediction error L_g, i.e.,

$$\min L_g = \sum \left\| y_{c+h,g} - \hat{y}_{c+h,g} \right\|_2^2. \tag{6}$$

Problem GBP-M

Given the training samples

$$\begin{aligned} \mathbf{X}_g &= \left[x_t^v\right]_g \\ \mathbf{Y}_g &= \{y_t\}_g \end{aligned} \tag{7}$$

where the granularity $g = 5$ min, with time indexes $t \in [0, c]$, $x_t^v \in \mathbf{R}$, $y \in \{-1, 1\}$. Our objective is to minimize the prediction error L_g, i.e.,

$$\min L_g = \sum \left\| y_{c+h,g} - \hat{y}_{c+h,g} \right\|_2^2. \tag{8}$$

3.2. Feature engineering and feature evaluation

As described in the introduction, the key insight of our research is to adopt high-dimensional features and machine learning algorithms to predict the Bitcoin price. We separately select the proper feature sets for Bitcoin daily and high-frequency prices prediction via feature engineering.

There are four types of features to select for daily price prediction. The first is based on their significance to the problem as observed in previous works, knowledge domain and understanding. Bitcoin's original property and network data, such as hash rate and block size, can be useful in predicting daily price [4]. The second feature type is related to the Bitcoin market and trading, and includes independent features such as market capitalization and transaction value. The first two types of daily data are acquired from Bitcoinity.org[4] and Blockchain Explorer.[5] We also consider a third feature type associated with Bitcoin price prediction: attention from the media and investors. Market confidence and the perceived usefulness of Bitcoin are reflected in its value [40], and the price of digital currency has been observed to be strongly correlated with trends in Google queries [19]. In a speculative market, sentiment plays a more important role in the price formation, so we take media hype into account by obtaining Baidu media search volume from the Baidu index. Furthermore, Bitcoin has been widely compared with gold by economists due to their similarities. Most of the value of Bitcoin and gold comes from the cost of extraction and the fact that they are scarce, and, like gold, Bitcoin must have some intrinsic value if its users are rational [41]. Both of them are mined by individuals or independent organizations rather than governments. As Bitcoin is regarded to overcome the weakness of fiat and gold-based money [2] at the beginning, we include the XAU gold spot price as one feature for our prediction. The 15 daily features included in the model are described in Table 1.

We evaluate which features to include in our models using Boruta [15], in which the classification is performed by voting for multiple classifiers in an ensemble method, based on a working principle very similar to the random forest classifier. We use additional forward and backward stepwise selection to further corroborate which features are the most useful and meaningful for the prediction models.

We leverage the best result for the final features selection and select the following 12 features: Block Size, Hash Rate, Mining Difficulty, Number of Transactions, Confirmed Transactions per Day, Mempool Transaction Count, Mempool Size, Market Capitalization, Estimated Transaction Value, Total Transaction Fees, Google Trend Search Volume Index, and Gold Spot Price. The summary statistics of the features are provided in Table 2. With these daily features, binary classification is developed for daily Bitcoin price prediction (see Fig. 2).

Because of the unique T+0 trading rules and the relatively high barrier for Bitcoin trading among different exchanges, more price manipulation has been done in recent years. The rollercoaster often moves very quickly, leading to micro-variations in the Bitcoin price. To gain deeper insight into these variations, we select the features for Bitcoin high-frequency price prediction. As feature data for tick trading data are poor or unavailable for very small intervals, we consider the original features acquired from the Bitcoin exchange Binance.

[4] A data-driven platform that provides some features of Bitcoin.

[5] A web tool that provides detailed information about Bitcoin blocks, addresses, and transactions.

Table 1
Daily features.

Feature	Definition	Feature type	Number
Block size	The average block size in MB.	Property & Network	1
Hash rate	The estimated number of tera hashes per second (trillions of hashes per second) the Bitcoin network is performing.	Property & Network	2
Mining difficulty	A relative measure of how difficult it is to find a new block. The difficulty is adjusted periodically as a function of how much hashing power has been deployed by the network of miners.	Property & Network	3
Time between blocks	The average time it takes to mine a block in minutes.	Property & Network	4
Trades per minute	The number of Bitcoin traded in minutes from the top and other exchanges.	Trading & Market	5
Number of transactions	The number of transactions per day.	Trading & Market	6
Confirmed transactions per Day	The number of daily confirmed Bitcoin transactions.	Trading & Market	7
Mempool transaction count	The number of transactions waiting to be confirmed.	Trading & Market	8
Market capitalization	The total US dollar market value of Bitcoin.	Trading & Market	9
Estimated transaction value	The total estimated value of transactions on the Bitcoin blockchain in US dollars (does not include coins returned to the sender as change).	Trading & Market	10
Total transaction fees	The total value of all transaction fees paid to miners in US dollars (not including the coinbase value of block rewards).	Trading & Market	11
Mempool size	The aggregate size of transactions waiting to be confirmed.	Trading & Market	12
Google trend search volume index	The normalized search volume for the inquiry "Bitcoin" per day.	Attention	13
Baidu media search volume	The weighted volume for media coverage of the keyword "Bitcoin."	Attention	14
Gold spot price	XAU gold spot price in US dollars.	Gold Spot Price	15

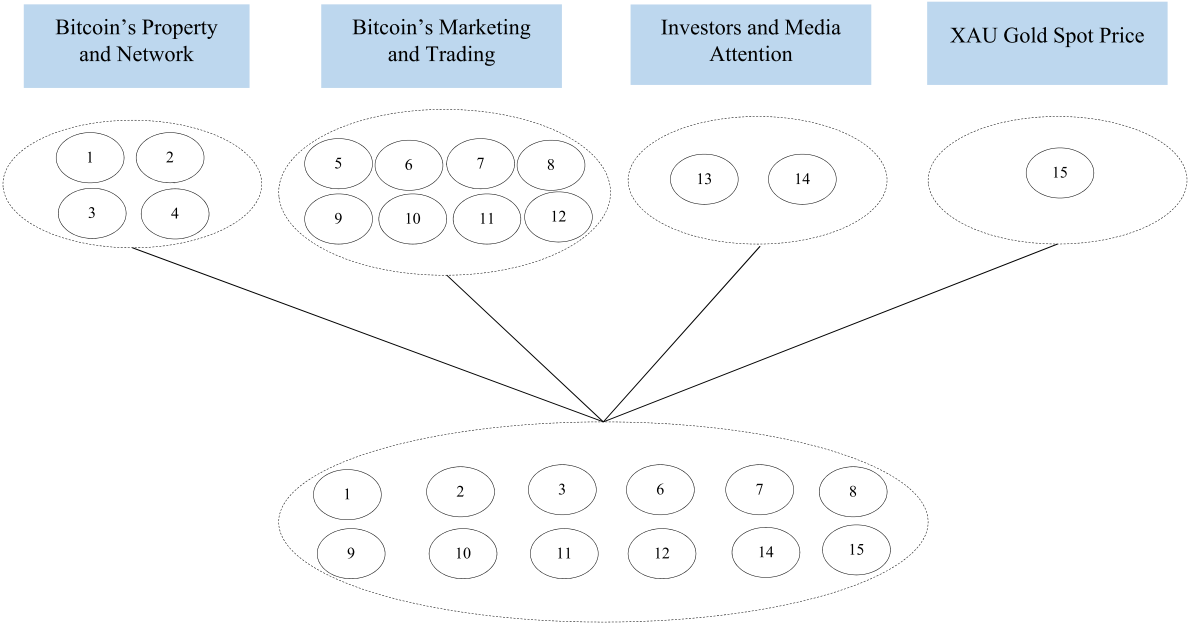

Fig. 2. Features engineering.

4. Implementation

4.1. Experimental design

Two datasets are employed. The first includes the aggregated Bitcoin daily price, with a big interval and small scale, from CoinMarketCap.com. It also includes property and network data, trading and market data, media and investor attention and gold spot price, for the period from February 2, 2017, to February 1, 2019. Fig. 3 plots the distribution of the Bitcoin daily price. A complete cycle for Bitcoin price rise and fall is considered. The price continued to rise from February 2017 and crashed from January 2018 to February 2019.

Table 2
Summary statistics of features used for bitcoin daily data prediction.

Feature	Count	Mean	SD	Min	Max
Block size	740	819710.5	157884.5	361626.6	998175.2
Hash rate	740	23412238	17352387	2917084	61866256
Mining difficulty	740	3.18E+12	2.41E+12	4.22E+11	7.45E+12
Number of transactions	740	255215.8	57038.29	131875	490644
Confirmed transactions per Day	740	255694.5	57012.15	131875	490644
Mempool transaction Count	740	255215.8	57038.29	131875	490644
Mempool size	740	26489513	35357624	35369.5	1.37E+08
Market capitalization	740	9.87E+10	6.1E+10	1.53E+10	3.23E+11
Estimated transaction value	740	193095	96494.26	37558.21	629491.3
Total transaction fees	740	165.9894	192.5094	11.2287	1495.946
Google trend search volume index	740	8.452703	10.53606	2	100
Gold spot price	740	1268.682	43.20863	1174.2	1357.91

Fig. 3. Bitcoin daily price distribution.

The second dataset consists of 5-minute interval Bitcoin real-time trading price data at high-frequency and large scale pulled from Binance, the top cryptocurrency exchange in the world. We collected tick data by building an automated real-time Web scraper that pulled data from the APIs of the Binance cryptocurrency exchange from July 17, 2017 to January 17, 2018, obtaining roughly 50,000 unique trading records including Price, Trading Volume, Open, Close, High, and Low points for use in our modeling. Fig. 4 illustrates the distribution of the Bitcoin 5-minute interval price. We can observe moderate growth during the period of January to May 2017 and a rapid rise to a peak at the beginning of 2018, which is the price turning point.

A laptop is configured to process the data for our experiments, with four cores of 3.60 GHz CPU and a total memory of 500 GB. We ordered multiple frequency Bitcoin price datasets and used the first 75% for training and the remaining 25% for testing.

4.2. Machine learning algorithms

This section presents the implementation of different machine learning techniques. For Bitcoin daily price with higher dimensional features, we implement two statistical methods: logistic regression (LR) and linear discriminant analysis (LDA), and the 5-minute interval price with a few features is predicted using machine learning models including random forest (RF), XGBoost (XGB), quadratic discriminant analysis (QDA), support vector machine (SVM), and long short-term memory (LSTM).

4.2.1. Logistic regression

Logistic regression (LR) is a traditional multiple variate regression method that can be implemented in binary classifications. The value of the binary response variable $y_i \in \{0, 1\}$, which indicates the class label, is predicted by the values of the feature x_i, where $i = 1, \ldots, K$.

The logistic regression model can be expressed as

$$\log it\left(P\left(y=1\right)\right) = \log\left(\frac{P\left(y=1\right)}{1-P\left(y=1\right)}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_K \tag{9}$$

Fig. 4. Bitcoin 5-minute interval price distribution.

where $P(y = 1)$ represents the probability of the sample belonging to class 1 and β_i represents the regression coefficient of x_i.

4.2.2. Linear discriminant analysis

Linear discriminant analysis (LDA) can be used to reduce the dimensionality of data and for classification purposes. It projects data onto a lower-dimensional space. This space provides maximum class separability. Features derived from LDA are linear combinations of the original features.

The optimal projection of features for LDA is achieved as follows. The within-class distance is minimized and the between-class distance is maximized. This results in maximum class separability. An important observation concerning two-class classification is that LDA is equivalent to linear regression. Thus, LDA can be formulated as a least squares problem.

4.2.3. Random forest

Random forest models use an ensemble of decision trees for various tasks to obtain a better classification result and are a popular approach. The use of decision trees [42,43] is one of the basic machine learning methods and is used to solve a wide range of problems in classification. Decision trees adopt a tree structure to recursively partition the feature space, with each node continuing to split to maximize purity until the nodes only contain single-class samples. These pure nodes are called leaf nodes. When a test sample is an input into a decision tree, it can be traced down to the leaf node and a class label can be assigned. By running a bootstrap aggregation (or bagging), a random subset of the whole feature space is assigned to the growth of each tree.

4.2.4. XGBoost

XGBoost is a framework and library that parallelizes the growth of gradient boosted trees in a forest [44]. It aims to minimize the time required to grow trees and speed up the process of optimizing, which makes gradient boosting decision trees (GBDTs) practical to use. A GBDT is a classifier that combines the results of many weak classifiers to make a strong prediction. It is an improved version of a decision tree because each tree is approximated by a large number of regression functions $f_i(x)$. By trying to better classify the residuals in the previous tree, the error in classification can decrease successively. Once each tree has been optimally approximated, the structure's scores and gain are calculated to determine the best split. Finally, the prediction result of the entire model is the sum of each decision tree. Like the random forest, a subset of the features is used to build each tree.

4.2.5. Quadratic discriminant analysis

Quadratic discriminate analysis (QDA) is another kind of distance discrimination method for supervised classification problems. It is assumed that measurements from each class are normally distributed. QDA is quite similar to LDA, but it allows for quadratic decision boundaries between classes. The parameters for each class can be estimated from training points with maximum likelihood estimation.

4.2.6. Support vector machine

A support vector machine is a kind of machine learning methodology that is applied in binary classification problems [45]. The basic principle of the SVM method is finding a separator hyper plane through nonlinear mapping in feature space with a maximum margin to classify data samples into two classes.

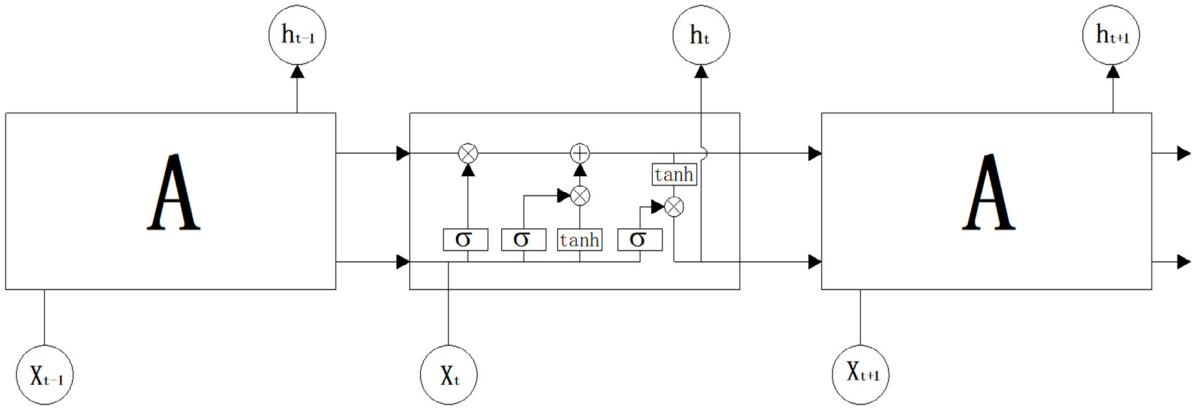

Fig. 5. Structure of long short-term memory model (see text for definitions of the elements of the model)

The class of each point x_i can be denoted by $y_i \in \{1, -1\}$. A linear SVM searches for an optimal decision hyper plane defined as

$$wx + b = 0 \tag{10}$$

where $w \in \mathrm{R}^n$, $b \in \mathrm{R}$, w is normal to the optimal hyper plane, termed as weight vector, and b is bias. The optimal hyperplane can be obtained by minimizing $\|w\|$. An equivalent mathematical formulation can be given by the quadratic program

$$\begin{aligned} &\min \ \frac{1}{2}\|w\|^2 + v\sum_{i=1}^{m}\xi_i \\ &\text{s.to} \ \ y_i(wx_i - b) \geq 1 - \xi_i, \\ &\qquad \xi \geq 0, \end{aligned} \tag{11}$$

where v is the regularization parameter and ξ_i is the non-negative slack variables. By using a standard QP solver, the minimization problem can be solved. The solution is a linear combination of a subset of sample points called support vectors.

4.2.7. Long short-term memory

To solve the problem of the gradient disappearance of an RNN, the long short-term memory (LSTM) model, which uses a memory cell and gate structure, was proposed [46]. As shown in Fig. 5, an LSTM unit consists of a memory cell that stores information that is updated via three special gates: the input gate, the forget gate, and the output gate. It performs better in representing history information and future information and in extracting long-distance dependencies of elements in sequence data.

LSTM is expressed as

$$X = \begin{bmatrix} x_t \\ h_{t-1} \end{bmatrix} \tag{12}$$

$$f_t = \delta\left(W_f \cdot X + b_f\right) \tag{13}$$

$$i_t = \delta\left(W_i \cdot X + b_i\right) \tag{14}$$

$$o_t = \delta\left(W_o \cdot X + b_o\right) \tag{15}$$

$$\tilde{C}_t = \tanh\left(W_c \cdot [h_{t-1}, x_t] + b_c\right) \tag{16}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{17}$$

$$h_t = o_t \odot \tanh\left(C_t\right) \tag{18}$$

where x_t is the input at time t; h_t is the hidden state at time t; W_t, W_i, W_o, and W_c are the weight matrix of LSTM, b_f, b_i, b_o, and b_c are the offset of the LSTM and all are the training parameters of the model; δ is the activation function; and $\odot$ is the dot multiplication operation.

4.2.8. Evaluation indicator

From the models above, we construct a confusion matrix and divide the results into four categories. The confusion matrix contains statistics about real classification data and the predictions generated by these seven classifier algorithms.

Table 3
Confusion matrix.

Confusion matrix		Prediction	
		1	0
Real	1	True positive	True negative
	0	False positive	False negative

Table 4
Parameters and configuration for logistic regression.

Penalty	Tol	C	Fit-intercept	Intercept scaling	Class weight	Random state	Max iter	Verbose	N jobs
L2	1E−4	1.0	True	1	None	None	100	0	None

Table 5
Parameters and configuration for linear discriminant analysis.

Solver	Tol	Shrinkage	Priors	N components	Store covariance
Svd	1E−4	None	None	None	False

Table 6
Parameters and configuration for random forest.

Crite-rion	Max depth	Min samples split	Min samples leaf	Min weight fraction leaf	Max features	Max leaf nodes	Mini impurity decrease	Mini impurity split	Boot-strap	Oob score	N jobs	Random state	Wer-bose	Warm start	Class weight
Gini	None	2	1	0	Auto	None	0	None	True	False	None	None	0	False	None

Table 7
Parameters and configuration for XGBoost.

Max depth	Learning rate	N estimators	Silent bool	Objective	Booster	N jobs	Nthread	Gamma	Min child weight
Int = 3	Float = 0.1	Int = 100	True	Str = binary: logistic	Str = gbtree	Int = 1	Int = none	Float = 0	Int = 1

Max delta step	Subsample	Colsample bytree	Closample bylevel	Reg alpha	Reg lambda	Scale pos weight	Base score	Random state	Seed
Int = 0	Float = 1	Float = 1	Float = 1	Int = 0	Int = 1	Float = 1	Float = 0.5	Int = 0	Int = 0

Table 8
Parameters and configuration for quadratic discriminant analysis.

Priors	Reg param	Store covariance
None	0	Bool = False Tol = 1.0E−4

From the results in the matrix, the performance of each method can be assessed. The rows and columns in the confusion matrix show the instances of real and predicted classes.

Table 3 indicates that if both the real label and the prediction are 1, the result is a true positive; if the real label is 1 and the prediction is 0, the result is a true negative; if the real label is 0 and the prediction is 1, the result is a false positive; and if the real label is 0 and the prediction is 0, the result is a false negative. According to the confusion matrix, evaluation indicators of accuracy, precision, recall and f1-score can be calculated by the formulas below.

$$Accuracy = (tp + fn)/(tp + tn + fp + fn) \tag{19}$$

$$Precision = tp/(tp + fp) \tag{20}$$

$$Recall = tp/(tp + fn) \tag{21}$$

$$F1\text{-}score = 2 \times Precision \times Recall/(Precision + Recall) \tag{22}$$

where tp denotes a true positive, tn denotes a true negative, fp denotes a false positive, and fn denotes a false negative.

4.3. Parameter settings and configuration

We apply the default parameters in Python as our optimal values. The Keras package is applied for the LSTM algorithm and Sklearn is used for other algorithms. Tables 4–10 show the parameters and configuration that we used.

Table 9
Parameters and configuration for support vector machine.

C	Kernel	Degree	Gamma	Coef	Shrinking	Probability
1.0	Rbf	3	Auto deprecated	0.0	True	False
Tol	Cache size	Class weight	Verbose	Max iter	Decision function shape	Random state
1E−3	200	None	False	−1	ovr	None

Table 10
Parameters and configuration for long short-term memory (LSTM).

Return sequences	Input shape	Units	Activation	Loss	Optimizer	Metrics	Win
True	1,11	1	Tanh	Mse	Nadam	Accuracy	7

Table 11
Performance of different models using the bitcoin daily price.

Models	LR	LDA	Average	QDA	SVM	RF	XGB	LSTM	Average
Accuracy	0.660	0.639	0.650	0.551	0.653	0.510	0.483	0.570	0.553
Precision	0.723	0.680	0.702	0.522	0.708	0.493	0.455	0.552	0.546
Recall	0.350	0.362	0.356	0.741	0.354	0.493	0.352	0.508	0.490
F1-score	0.472	0.472	0.472	0.612	0.472	0.493	0.397	0.529	0.501

Table 12
Performance of different models using the bitcoin 5-minute interval price.

Models	LR	LDA	Average	QDA	SVM	RF	XGB	LSTM	Average
Accuracy	0.545	0.515	0.530	0.588	0.549	0.648	0.654	0.672	0.622
Precision	0.639	0.472	0.556	0.612	0.765	0.731	0.817	0.722	0.729
Recall	0.471	0.495	0.483	0.722	0.588	0.787	0.648	0.840	0.717
F1-score	0.542	0.483	0.513	0.662	0.665	0.758	0.722	0.776	0.717

5. Results

Table 11 summarizes the performance of all of the machine learning models concerning for the Bitcoin daily price. From the results, we can make the following observations. As expected, the results of the two statistical methods are better overall. The average accuracy of the statistical methods is 65.0%, higher than the average accuracy of the machine learning models (55.3%). The LR model achieved the best results, with an accuracy of 66.0%. Among the machine learning models, XGB performed the worst, with an accuracy of 48.3%, and SVM was the best, with an accuracy of 65.3%, competitive with the statistical methods. In general, LR and LDA outperformed the other machine learning models on the daily price dataset, indicating that properly selected high-dimensional feature sets can compensate for the simplicity of models in Bitcoin daily price prediction.

Table 12 summarizes the performance of all the machine learning models concerning for the Bitcoin 5-minute interval price. As shown, the machine learning models achieved better accuracy than the statistical methods, with LSTM achieving the best result (67.2% accuracy). The average accuracy of the statistical methods was only 53.0%, worse than that of the machine learning models (62.2%). The prediction accuracies of the LR and LDA statistical methods were 54.5% and 51.5%, respectively; both results were inferior to the accuracies of the machine learning models. The machine learning models generally outperformed the two statistical methods due to the large scale of the Bitcoin 5-minute interval dataset. This result is consistent with the paradigm for the main resolution of the practical prediction of complicated models with few features.

To the best of our knowledge, few of the published works use the same methods with ours in Bitcoin price prediction till now. Our machine learning algorithms leveraged in this paper are more comprehensive. Table 13 shows a comparison of our results and the benchmark results of McNally [15], who used the proximal methods including linear ARIMA model and LSTM and RNN machine learning models to predict the Bitcoin daily price. All of our results outperform the benchmark results in both accuracy and precision except for XGB. For linear methods, ARIMA has an overwhelming advantage, with a precision of 100%. Our LR and LDA methods outperform ARIMA on accuracy.

6. Conclusion and discussion

In this study, we investigated machine learning techniques based upon sample characteristics of sample and dimension to predict Bitcoin price. While most previous works simply leverage machine learning algorithms in Bitcoin price prediction, we show that the sample's granularity and feature dimensions should be considered. The Bitcoin aggregated daily price, acquired from CoinMarketCap, facilitates the inclusion of high-dimensional features, including property and

Table 13
Performance comparison against the benchmark.

Models	LSTM*	LSTM	RNN*	QDA	SVM	RF	XGB	ARIMA*	LR	LDA
Accuracy	0.528*	0.570	0.502*	0.551	0.653	0.510	0.483	0.500*	0.660	0.639
Precision	0.355*	0.552	0.391*	0.522	0.708	0.493	0.455	1.00*	0.723	0.680

*Denotes the benchmark methods and results of McNally [15].

network, trading and market, attention and gold spot price. The Bitcoin 5-minute interval trading price is facilitated by features from the Binance exchange. Based on the Occam's razor principle and the paradigms applied in practical prediction problems using machine learning algorithms, we adopted statistical methods for Bitcoin daily price prediction and machine learning models for Bitcoin 5-minute interval price prediction. The results show that the statistical methods perform better for low-frequency data with high-dimensional features, while the machine learning models outperform statistical methods for high-frequency data. Most of our results also outperform the benchmark results of other machine learning algorithms. We envision that our approach to sampling dimension engineering using machine learning models for the prediction can be applied to other areas that have similar characteristics to Bitcoin.

Our research has several limitations in its data sources and analyses, which suggest possible extensions to this study. We acquired two kinds of data for prediction. To make a more comprehensive study of Bitcoin price prediction in the future, it is necessary to collect price data with various granularities and features with more dimensions. Secondly, we did not leverage all of the machine learning algorithms in our evaluations. To improve this study, we intend to examine more methods, such as the statistical method ARIMA, and the machine learning model RNN. Moreover, other features should be considered, and our further studies will focus on more useful elements of the sentiment using text mining and analyses of social networks.

References

[1] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system, 2008.
[2] D. Yermack, Is bitcoin a real currency? an economic appraisal, in: Handbook of Digital Currency, Elsevier, 2015, pp. 31–43.
[3] F. Mai, et al., How does social media impact bitcoin value? a test of the silent majority hypothesis, J. Manage. Inf. Syst. 35 (1) (2018) 19–52.
[4] I. Madan, S. Saluja, A. Zhao, Automated bitcoin trading via machine learning algorithms, vol. 20. URL: http://cs229.stanford.edu/proj2014/Isaac%20Madan, 2015.
[5] P.G. Nieto, et al., A comparison of several machine learning techniques for the centerline segregation prediction in continuous cast steel slabs and evaluation of its performance, J. Comput. Appl. Math. 330 (2018) 877–895.
[6] B.M. Brentan, et al., Hybrid regression model for near real-time urban water demand forecasting, J. Comput. Appl. Math. 309 (2017) 532–541.
[7] C. Ordóñez, et al., A hybrid ARIMA–SVM model for the study of the remaining useful life of aircraft engines, J. Comput. Appl. Math. 346 (2019) 184–191.
[8] R. Stefanescu, A. Moosavi, A. Sandu, Parametric domain decomposition for accurate reduced order models: applications of MP-LROM methodology, J. Comput. Appl. Math. 340 (2018) 629–644.
[9] H.H. Le, J.-L. Viviani, Predicting bank failure: an improvement by implementing a machine-learning approach to classical financial ratios, Res. Int. Bus. Finance 44 (2018) 16–25.
[10] D. Lv, et al., Selection of the optimal trading model for stock investment in different industries, PLoS One 14 (2) (2019) e0212137.
[11] L.M.Q. Abualigah, E.S. Hanandeh, Applying genetic algorithms to information retrieval using vector space model, Int. J. Comput. Sci. Eng. Appl. 5 (1) (2015) 19.
[12] L.M. Abualigah, A.T. Khader, E.S. Hanandeh, A combination of objective functions and hybrid Krill herd algorithm for text document clustering analysis, Eng. Appl. Artif. Intell. 73 (2018) 111–125.
[13] L.M. Abualigah, A.T. Khader, E.S. Hanandeh, Hybrid clustering analysis using improved krill herd algorithm, Appl. Intell. 48 (11) (2018) 4047–4071.
[14] L.M. Abualigah, et al., A novel hybridization strategy for krill herd algorithm applied to clustering techniques, Appl. Soft Comput. 60 (2017) 423–435.
[15] S. McNally, J. Roche, S. Caton, Predicting the price of bitcoin using machine learning, in: 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing, PDP, IEEE, 2018.
[16] L.M. Abualigah, A.T. Khader, Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering, J. Supercomput. 73 (11) (2017) 4773–4795.
[17] L.M. Abualigah, A.T. Khader, E.S. Hanandeh, A new feature selection method to improve the document clustering using particle swarm optimization algorithm, J. Comput. Sci. 25 (2018) 456–466.
[18] E. Pagnotta, A. Buraschi, An equilibrium valuation of bitcoin and decentralized network assets, 2018.
[19] L. Kristoufek, Bitcoin meets google trends and wikipedia: quantifying the relationship between phenomena of the internet era, Sci. Rep. 3 (2013) 3415.
[20] L. Kristoufek, What are the main drivers of the Bitcoin price? evidence from wavelet coherence analysis, PLoS One 10 (4) (2015) e0123923.
[21] A. Hayes, What factors give cryptocurrencies their value: An empirical analysis, 2015.
[22] P. Ciaian, M. Rajcaniova, d.A. Kancs, The economics of bitcoin price formation, Appl. Econ. 48 (19) (2016) 1799–1815.
[23] R.J. Barro, Money and the price level under the gold standard, Econ. J. 89 (353) (1979) 13–33.
[24] M. Balcilar, et al., Can volume predict bitcoin returns and volatility? a quantiles-based approach, Econ. Model. 64 (2017) 74–81.
[25] J. Bukovina, M. Martiček, Sentiment and bitcoin volatility, Mendel University in Brno, Faculty of Business and Economics, 2016.
[26] Y.B. Kim, et al., Predicting fluctuations in cryptocurrency transactions based on user comments and replies, Plos One 11 (8) (2016).
[27] D. Shah, K. Zhang, Bayesian regression and bitcoin, in: Communication, Control, and Computing (Allerton), 2014 52nd Annual Allerton Conference on, IEEE, 2014.
[28] G.H. Chen, S. Nikolov, D. Shah, A latent source model for nonparametric time series classification, in: Conference on Advances in Neural Information Processing Systems, 2013.

[29] I. Georgoula, et al., Using time-series and sentiment analysis to detect the determinants of bitcoin prices, SSRN 2607167, 2015..
[30] A. Greaves, B. Au, Using the bitcoin transaction graph to predict the price of bitcoin, stanford.edu, 2015.
[31] D. Gamberger, N. Lavrač, Conditions for occam's razor applicability and noise elimination, in: European Conference on Machine Learning, Springer, 1997.
[32] J. Zahálka, F. Železný, An experimental test of Occam's razor in classification, Mach. Learn. 82 (3) (2011) 475–481.
[33] P. Domingos, The role of occam's razor in knowledge discovery, Data Min. Knowl. Discov. 3 (4) (1999) 409–425.
[34] Y. Tong, et al., The simpler the better: a unified approach to predicting original taxi demands based on large-scale online platforms, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2017.
[35] J. Langford, A. Blum, Microchoice bounds and self bounding learning algorithms, Mach. Learn. 51 (2) (2003) 165–179.
[36] Y. Freund, Self bounding learning algorithms, in: COLT, Citeseer, 1998.
[37] M.K. Ebrahimpour, et al., Occam's razor in dimension reduction: using reduced row echelon form for finding linear independent features in high dimensional microarray datasets, Eng. Appl. Artif. Intell. 62 (2017) 214–221.
[38] M. Zhenin, et al., Rescoring of docking poses under occam's razor: are there simpler solutions?, J. Comput. Aid. Mol. Des. 32 (9) (2018) 877–888.
[39] L.-C. Wang, Acm, Machine learning for feature-based analytics, in: Proceedings of the 2018 International Symposium on Physical Design, Assoc Computing Machinery, New York, 2018, pp. 74–81.
[40] P. Bacao, et al., Information transmission between cryptocurrencies: does bitcoin rule the cryptocurrency world?, Sci. Ann. Econ. Bus. 65 (2) (2018) 97–117.
[41] A.H. Dyhrberg, Bitcoin, gold and the dollar–a garch volatility analysis, Finance Res. Lett. 16 (2016) 85–92.
[42] J.R. Quinlan, Induction of decision trees, Mach. Learn. 1 (1) (1986) 81–106.
[43] P. Geurts, G. Louppe, Learning to rank with extremely randomized trees, in: JMLR: Workshop and Conference Proceedings, 2011.
[44] S. Basak, et al., Predicting the direction of stock market prices using tree-based classifiers, North American J. Econ. Finance 47 (2019) 552–567.
[45] D. Kumar, S.S. Meghwani, M. Thakur, Proximal support vector machine based hybrid prediction models for trend forecasting in financial markets, J. Comput. Sci. 17 (2016) 1–13.
[46] S. Wang, et al., Bi-directional long short-term memory method based on attention mechanism and rolling update for short-term load forecasting, Int. J. Electr. Power Energy Syst. 109 (2019) 470–479.