What factors influence Bitcoin's daily price direction from the perspective of machine learning classifiers?

Tea Kalinic Milicevic[1,*] **and Branka Marasovic**[1]

[1] *Faculty of Economics, Business, and Tourism, University of Split, Cvite Fiskovića 5, Split, Croatia*
E-mail: ⟨{tkalinic, bmarasov}@efst.hr⟩

Abstract. The paper examines the factors that influence Bitcoin price direction from the perspective of machine learning (ML) models. The observed factors cover Bitcoin market data, technical indicators, blockchain variables, sentiment analysis, and other macro-financial variables. Logistic Regression (LR), Random Forest (RF), and Support Vector Machine (SVM) classifiers are employed. Three train-test ratios are considered. Grid search and blocking time series cross-validation are used to adjust the hyperparameters of the proposed ML algorithms resulting in the three most accurate models for each train-test ratio. Variables that affect the next-day price direction are ranked using LR and RF best models. For each method and train-test ratio, the smallest subsets of independent variables with the highest test set accuracy were chosen to reduce dimensionality. Models show that technical indicators influence daily Bitcoin price direction the most, followed by blockchain and Bitcoin market variables. Contrarily, models disagree on the importance of Tweets and macro-financial variables. Finally, SVM performed better on the test set when the LR optimal sets of independent variables were considered, indicating that the analysis of individual factors' influence on the Bitcoin price is not important only for corresponding model. Combining only influential independent variables and 90:10 train-test ratio yielded the greatest accuracy of 58.18% achieved by RF model.

Keywords: Bitcoin, feature selection, machine learning classifiers, price direction

Received: September 13, 2023; accepted: December 12, 2023; available online: December 19, 2023

DOI: 10.17535/crorr.2023.0014

1. Introduction

Cryptocurrencies achieved great popularity in 2017 after several consecutive months of exponential growth in their market capitalization. After the big cryptocurrency crash that occurred at the beginning of 2018, also known as the Bitcoin Crash, interest in cryptocurrencies increased drastically [14]. The estimated market capitalization places Bitcoin at the highest point of the cryptocurrency market. High market capitalization is connected with high trading volumes as well as high Bitcoin prices. In 2017, the price of Bitcoin rose from $900 to an incredible $20,000. The second period that saw a noticeable spike in the price and volume of Bitcoin is the COVID-19 crisis. In a period of general uncertainty in the market due to the sudden pandemic crisis, Bitcoin shone brighter than ever before in the next two pandemic years, despite a slight drop in price and volume in the first quarter of 2020. Exponential increases in Bitcoin prices represented an opportunity to achieve high gains in a very short period. Given that the Bitcoin market, and the cryptocurrency market in general, is highly volatile and does not have seasonal effects like the stock market, an accurate price prediction is extremely challenging.

Previous research on Bitcoin price prediction (and cryptocurrencies in general) has utilized methods that can be categorized into two main categories: traditional and machine learning

*Corresponding author.

(ML) methods [14]. The most commonly used traditional models are the multivariate linear regression, multivariate vector autoregressive (VAR) model, extended VAR model, and generalized autoregressive conditional heteroscedasticity (GARCH) model [8]. With the development of artificial intelligence (AI) and Big Data science, ML has become more popular in scientific research. Motivated by the effective applications of these models in a variety of financial markets, researchers began to apply them more frequently in the study of the cryptocurrency market. Generally, AI is used to address a variety of issues related to the crypto market, including security, fraud detection, automated trading, privacy, mining, as well as price forecasting which is the first step toward profitable trade for traders and market observers [26].

Some studies compared the performance of traditional and ML methods for Bitcoin price forecasting. Ibrahim et al. [12] evaluated various ML and statistical models, including Autoregressive Integrated Moving Average (ARIMA), Prophet (developed by Facebook), Random Forest (RF), RF Lagged-Auto-Regression, and Multi-Layer Perceptron (MLP) Neural Networks (NNs) for Bitcoin direction prediction within a 5-minute time frame. The MLP deep neural network exhibited the highest accuracy rate of 54%. Chen et al. [6] compared five ML models (Support Vector Machine (SVM), RF, Quadratic Discriminant Analysis (QDA), Long Short-term Memory (LSTM)) with two traditional statistical models (Logistic Regression (LR), Linear Discriminant Analysis (LDA)) and their results showed that statistical methods outperform the ML methods for daily predictions, with an average accuracy of 65.0%, while the average accuracy of the ML models was 55.3%. On the other side, in the case of a 5-min time interval, their results showed that the ML models outperformed the statistical methods, with LSTM achieving the best result (67.2% accuracy). Furthermore, results of predicting binary values of 15,30,60-min returns, in the study of Akyildirim et al. [2] showed that SVM outperforms all other observed ML models (LR, RF, NN, and Ensemble model of five ML models) as well as traditional ones (ARIMA and random walk). In contrast to [6], in the case of daily predictions, SVM also outperformed all other models. Khedr et al. [14] provided a comprehensive evaluation of papers that utilized either traditional or ML models to predict cryptocurrency prices. In addition to comparing the performance of ML and statistical models, researchers have also started merging AI with traditional models, resulting in the development of hybrid models, which have primarily been utilized for forecasting cryptocurrency price volatility [11, 15, 21].

In general, cryptocurrency price forecasting can be formulated as classification problem and regression problem. The base of regression problems lies in the forecasting of cryptocurrency returns, prices, or volatility, while classification problems primarily aim to forecast the direction or trend of cryptocurrency prices.

This study employs ML classifiers to predict the direction of Bitcoin prices for the following day. For intraday price direction forecasting, deep learning (DL) and NN models outperformed other ML and statistical models, according to the findings of some previous research. However, for daily price forecasting, other ML classifiers and traditional models mostly performed better. The research conducted by Ibrahim et al. [12] and Chen et al. [6] revealed that MLP and LSTM models were more effective in predicting the movement of Bitcoin values within a 5-minute interval, surpassing other models in terms of performance. Moreover, an additional part of the analysis carried out by Chen et al. [6] revealed that NNs exhibited inferior performance compared to other models in the context of daily forecasting. Akyildirim et al. [2] demonstrated that SVM exhibits superior performance compared to NNs, other observed ML models, and traditional models, not only in daily forecast but also in forecasts for 15, 30, and 60 minutes time intervals. Furthermore, in a research by Pabucco et al. [20], RF outperformed other models in predicting the direction of Bitcoin closing prices using a continuous dataset. Neural Networks (NNs) outperformed alternative models when applied to discrete datasets. However, their accuracy was lower compared to the accuracy of the highest performing model (RF) when applied to continuous datasets. Borges and Neves [4] compared LR, RF, SVM, and Gradient Decision Tree Boosting (GTB), together with ensemble voting with the objective of predicting

the trend of BNB coin prices. Their findings indicated that, on average, the ensemble voting approach exhibited superior performance compared to the other learning algorithms, achieving an accuracy rate of 56.28%. Additionally, when considering models based on a single classifier, RF demonstrated the best results.

Following the literature, in order to build the model to forecast direction of Bitcoin prices in daily time interval, three ML algorithms, namely LR, RF, and SVM are used. The LR model has been widely used in binary classification problems, and the model's parameters can be utilized to analyze the relation between the dependent and independent variables. SVM and RF have demonstrated good performance in prior research. Due to the availability of various kernel functions, SVM can be effectively employed in scenarios where the observed variable space lacked linear separability. The RF algorithm does not make assumptions about the linearity between variables. Additionally, this model can be used to rank the independent variables based on their impact on the dependent variable. According to Pabucco et al. [20], the use of RF has gained popularity over NN due to its easy-to-use nature. The selection of previous algorithms is based on the observation that prior studies on forecasting daily bitcoin price movements have generally shown superior performance of ML models that do not rely on DL algorithms. In addition, Borges and Nuves [4] argue that simpler models are considered more favorable compared to models associated with NNs, particularly when investigating the correlation between independent and dependent variables. This preference arises from the fact that NNs involve a multi-layered process, which does not provide a clear understanding of the relationship between independent and dependent variables.

The performance of a model depends not only on the ML algorithm but also on the selected independent variables [14]. In previous research, various factors have been utilized to predict Bitcoin prices and returns. Akyildirim et al. [2] used both cryptocurrency market data and calculated technical indicators, while Pabucco et al. [20] and Borges and Nunes [4] used only technical indicators calculated from close, high, and low Bitcoin prices. Similarly, Mudassir et al. [18] used only technical indicators but calculated from a chosen set of blockchain variables. In contrast to these studies, Chen et al. [6] did not include technical indicators in the set of independent variables. Also, they constructed different sets of independent variables for minute and daily forecasting. For 5-minute price direction forecasting, they used only OHCLV data which includes open, high, close, low prices and volume. For daily time intervals, they used market capitalization, blockchain data, Google trend search volume index, and Baidu media search volume as representatives of public sentiment analysis variables, and Gold spot price as representative of macro-financial data. Furthermore, Jaquart et al. [13] observed several representatives of macro-financial data (S&P500 returns, MSCI World returns, Gold returns, VIX returns) as independent variables in addition to Bitcoin returns, number of Bitcoin Transactions, and Twitter sentiment as well as Twitter sentiment weighted with the strength of emotion. In addition, they differentiate sets of independent variables for models with and without memory.

In some studies, authors examined the impact that the observed set of independent variables has on dependent variable. Jaquart et al. [13] point out that technical indicators are more crucial for minute predictions. In addition, Goczek and Skliarov [10] found out that a number of transactions has no impact on the price of Bitcoin. Similarly, Poyser [22] has not found any relevant effect of confirmation time, hash rate, and the number of transactions per day on Bitcoin's price, which is contrary to results from [16] and [9] that show a positive, albeit small, effect of blockchain variables on Bitcoin's price. When it comes to sentiment analysis measures, most researchers agree that attractiveness is the main driver of Bitcoin price ([22], [10], [24], [23]). Even though there is no consensus regarding the impact of macro-finance factors on Bitcoin prices, the macro-finance variables most usually associated with Bitcoin prices are the S&P500, gold, oil, real estate, VIX, and exchange rates. In this paper, the chosen set of independent variables consists of one or more representatives from each of the presented

categories i.e. market data, blockchain variables, technical indicators, sentiment analysis, and macro-finance factors.

The primary objective of this paper is to examine the influence of the chosen independent variables on the next-day Bitcoin price direction in order to simultaneously improve the model's accuracy while decreasing the dimensionality of the problem. To evaluate the influence of independent variables on the dependent variable, LR and RF models are used to rank independent variables according to their significance for the observed dependent variable. Furthermore, different subsets of ranked independent variables are employed to identify the optimal number of independent variables that yielded the highest accuracy on the test set. The result includes two sets of independent variables that are considered optimal in terms of their significance, as determined by the LR and RF algorithms. Finally, the SVM classifier is used to determine if the sets of important independent variables have an impact on the accuracy of models based on a different algorithm. In addition, the analysis is performed using several train-test ratios, specifically 90:10, 80:20, and 70:30, to assess the influence of dataset partitioning on model performance.

The remainder of the paper is organized as follows. Section 2 describes the data and methodology. Section 3 presents empirical findings with discussion of the results. Conclusions is provided in Section 4.

2. Data and Methodology

Since most researchers agreed that it is difficult to make predictions based solely on historical data, the set of independent variables whose influence on the direction of the Bitcoin price is examined in this paper is divided into five categories shown in Table 1.

Category	Variables	Source
Bitcoin market	OHCLV, market capitalization	CmcScraper
Blockchain	number of transactions, average block size/block time/hash rate, average mining difficulty/transaction fee	https://bitinfocharts.com
Macro-financial	TNX, VIX, S&P500	https://fred.stlouisfed.org
Sentiment analysis	Tweets	https://bitinfocharts.com
Technical indicators	MA, MACD, RSI, STOCH, ADX	calculated in Python

Table 1: *Categories of independent variables*

Market variables are the most common financial variables used in time series analysis. The Bitcoin blockchain category includes six different variables: the number of confirmed transactions in blockchain per day, the average block size in megabytes (MB), average block time (time needed to mine a new block) in minutes, average hash rate (estimated number of tera hashes per second performed by the network) per day, average mining difficulty per day, and average transaction fee paid to miners, in USD. The three indices related to traditional asset classes are also included in the set of independent variables: Standard & Poor's 500 Index (S&P500), The Chicago Board Options Exchange (CBOE) Volatility Index (VIX) , and 10-year Treasury Note Yield Index (TNX). As the representative of sentiment analysis variables, a number of Tweets per day is used in this paper. Various technical indicators are used in different papers. In this paper, five technical indicators for closing price are used: simple moving average (MA), moving average convergence/divergence (MACD), relative strength index (RSI), stochastic oscillator (STOCH), and average directional movement index (ADX). The dependent variable in

the study is the next-day price direction. More formally, if $P_t, t \in T$ are Bitcoin close prices, then the next-day price direction y_t is calculated with the following rule:

$$y_t = \begin{cases} 0, & ln\frac{P_{t+1}}{P_t} < 0 \\ 1, & \text{otherwise} \end{cases} \tag{1}$$

Moreover, prices $P_t, t \in T$ are used to calculate five technical indicators. The most common and simplest indicator is moving average (MA) which represents the average price change over a specified period and thus indicating the general trend direction [20]. In this paper, six different lengths of the period are observed $n = 5, 10, 15, 30, 90, 180$, in order to test whether a trend from the recent or distant past has a greater impact on the direction of the bitcoin price. The Moving Average Convergence/Divergence (MACD) indicator displays the relationship between short and long Exponential Moving Averages (EMA) of security prices and thus indicating bullish/bearish shift in supply/demand lines [20]. In this paper, following Borges and Neves [4], the MACD is calculated with default parameters for short and long EMA, i.e. by subtracting a 26-day MA of a security's price from a 12-day MA of its price. Relative strength index (RSI) measures the speed and magnitude of recent price fluctuations to indicate overbought or oversold levels of securities prices [20]. A stochastic oscillator (STOCH) is a momentum indicator that compares a close price of security to its price range over a specified period and indicates how highest or lowers security's closing price was in comparison to the preceding n periods [1]. Finally, the Average Directional Movement Index (ADX) was constructed by Wilder Jr. [25] and it is used to measure the strength of the trend.

After obtaining all data and calculating each of the previously mentioned technical indicators, the initial dataset can be defined in the form of supervised dataset $D' = \{(x_t, y_t)\}_{t \in T'}$, where $x_t \in \mathbf{R}^N$ is a vector of $N = 27$ independent variables, and y_t is dependent variable i.e. the next-day close price direction. The data preprocessing phase is also conducted because the initial dataset contains data gaps, which are primarily caused by the absence of non-working day data for traditional indices such as the S&P500, VIX, and TNX. Following Mudassir et al. [18] linear interpolation is used to fill the gaps. In addition to dealing with missing data, the data preprocessing phase includes the calculation of percentage change and the data scalling. To improve training efficacy, the existing values for all observed independent variables $x^i, i = 1, 2, ..., N$, except one lag return, are replaced with corresponding percentage changes i.e. $x_t^i \leftarrow (x_t^i - x_{t-1}^i)/x_t^i, t \in \mathbf{T}'$. Finaly, the min-max scaler is employed to rescale variables, ensuring they fall inside the range of [0,1]. ML algorithms exhibit improved performance when applied to scaled data due to the mitigation of issues arising from different units of measurement among independent variables. Scaling the data prevents the dominance of certain variables purely based on their larger values compared to others [4]. Following the preparation of the initial dataset $D' = \{(x_t, y_t)\}_{t \in T'}$, final dataset $D = \{(x_t, y_t)\}_{t \in T}$ consists of 1642 days, with indices $t = 0$ and $t = 1641$ corresponding to 2017-07-05 and 2022-01-01, respectively.

Three ML algorithms are considered in the analysis: Logistic Regression, Random Forest Classifier, and Support Vector Classifier.

LR is a widely used classification algorithm which can also be considered as a single layer NN with binary response variable [2]. The model belongs to the group of linear discriminative models, meaning that it assumes the existence of a linear boundary between classes, which is characterized by the model's parameters. In binary LR, the values of the independent variables $x^i, i = 1, ..., N$, are used to predict the value of the binary dependent variable $y_i \in \{0, 1\}$, which indicates the class label. For $(x, y) \in \mathbf{R}^N \times \{0, 1\}$, the logistic regression model can be expressed as:

$$logit(P(y = 1|x)) = log\left(\frac{P(y = 1)|x}{1 - P(y = 1|x)}\right) = w^T x + w_0, \tag{2}$$

where $P(y = 1|x)$ is the probability that the sample x belongs to class 1, $w^T = [w_1, w_2 \dots w_N$

is the weight vector, and w_0 is the threshold, i.e. w_0, w are regression coefficients of x^i. The inverse of $logit(P(y=1|x))$ is the logistic function, also called sigmoid function:

$$P(y=1|x) = sigmoid(w^T x + w_0) = \frac{1}{1+e^{-(w^T x+w_0)}}. \tag{3}$$

The magnitude of the weight w_i shows the importance of x^i on final output y, and its sign indicates if the effect is positive or negative [3]. The main advantage of the LR model is in its simplicity and efficiency in implementation. Compared to NNs, this model is less prone to overfitting due to its smaller number of parameters to estimate [2]. Additionally, these parameters can be used to examine relation between independent and dependent variables. A limiting characteristic of LR is the assumption of linear separability of the observed space of independent variables.

RF [5] is a method of ensemble learning where multiple decision trees are independently constructed using random samples drawn with replacement (known as a bootstrap sample) of the dataset and their results are aggregated. This bootstrap aggregation, known as bagging, enables reduction of overfitting while increasing the accuracy of unstable models [4]. Moreover, by parallel training multiple classifiers, required training time is also reduced [2]. RF has the additional benefit of being able to rank independent variables according to their impact on the dependent variable. At each node of the RF's binary trees, the optimal split is determined using the Gini impurity, which measures how effectively the potential split divides the samples of the two classes at this node. The variable and threshold that maximizes the decrease in Gini impurity are found by exhaustively searching all independent variables at the node (the RF reduces this search to a random subset of independent variables) and all thresholds. The decrease in Gini impurity resulting from this optimal division is recorded and accumulated for each node and tree in the forest and for each variable. This accumulated value is called the Gini importance, and it shows how often a certain independent variable was used for splitting and how useful it was overall for the given classification problem [17].

SVM, proposed by Vapnik in 1995, is based on a process that minimizes structural risk by maximizing the margin between samples from different classes. This is not a stochastic model and it always gives the same results when the same dataset is processed at any given time [20]. In binary classification problems, SVM separates the two classes (positive and negative class) of the real problem by constructing a hyperplane such that distance of separation is maximum ([20],[18]). A maximum margin optimization problem with possible error is stated as:

$$argmin_{w,w_0,\xi} \left\{ \frac{1}{2}\|w\|^2 + C\sum_{t\in T}\xi_t \right\}, \tag{4}$$

s.t.

$$y_t(w^T x_t + w_0) \geq 1 - \xi_t, t \in T, \tag{5}$$

$$\xi_t \geq 0, t \in T. \tag{6}$$

In the previous formulation of the optimization problem, the vector (w_0, w) defines the hyperplane, while variables ξ_t are known as slack variables and they indicate how much each sample entered inside the margin. If errors are allowed in the sense that examples can be found within the margins, the problem is subject to soft margins. Otherwise, it is subject to hard margins. C is a hyperparameter that determines the compromise between margin width and hardness [3]. Most real-world situations contain non-separable data, hence no hyperplane can separate positive and negative examples from the train set. The inseparability problem can be solved by mapping the independent variables from train set into a higher-dimensional space and establishing a separating hyperplane. Selecting an appropriate kernel function is critical for improved outcomes in training set classification, since it specifies the transformed space of independent

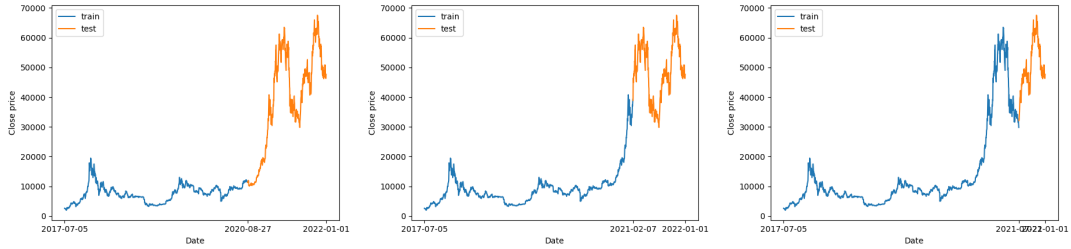
variables [19]. Linear, polynomial, RBF, and Sigmoid kernel functions are common. The advantage of SVM classification is that it produces globally optimal values. However, the outputs of SVM are still dependent on the kernel functions used [2].

Three ML algorithms, previously introduced, are employed to construct models. These models are built using three different train-test ratios. The hyperparameters of the models are fine-tuned through a grid search technique, utilizing a blocking time series split with three splits for cross-validation. The performance of the cross-validated models is evaluated based on accuracy. Tuning hyperparameters resulted in three models, for each train-test ratio, with the highest accuracy respectively named, LR best model, RF best model, and SVM best model. The LR best model is trained on the train set and the weights assigned to each of the independent variables are used to rank those variables by importance. In addition, the Gini importance derived from the structure of the RF best model, also trained on the train set, is used to rank the independent variables. Thus, the independent variables are ranked in two ways using different models and by comparing the obtained rankings, it was analyzed what variable affect the most next-day Bitcoin price direction as well as whether there is a similarity in the importance assigned to individual variable from the perspective of different models evaluated for different train-test ratios.

Furthermore, the effect of the cardinality of the set of independent variables on the accuracy of the model on the test set is analyzed for each train-test ratio. As part of that, to reduce the dimensionality of the problem, the smallest subsets of a set of independent variables that resulted in the highest test set accuracy were determined for both ranking algorithms. In the case of the LR best model, the model is fitted to the train set and tested on the test set for each size of the ranked set of independent variables, beginning with the set containing the three most important independent variables and progressing to the set containing all variables. The set of first k independent variables with the highest accuracy on the test set is further called the LR optimal set of independent variables. Similarly, with RF best model and the set of independent variables that yielded the highest accuracy on the test set, RF optimal set of independent varibles is obtained. The impact of dimensionality reduction on the prediction accuracy is assessed by comparing the accuracies of the LR and RF best models on the complete test set and the test set represented by the corresponding optimal set of independent variables. The SVM best model is used to examine if the optimal sets of independent variables, determined by the LR and RF best models, improve the predictive accuracy of the model based on another type of algorithm.

3. Results and Discussion

Following the data collection and preprocessing phase described in Section 2, the final dataset $D = \{(x_t, y_t)\}_{t \in T}$ consists of 1642 time points, covering the time period from 2017-07-05 to 2022-01-01. Given that the observed set D is a time series with varying levels of volatility in specific segments, the ratios 70:30, 80:20, and 90:10 are considered in the analysis. Figure 1 shows the movement of the Bitcoin daily close prices in an observed time interval.



(a) 70:30 ratio (b) 80:20 ratio (c) 90:10 ratio

Figure 1: *Bitcoin daily close price movement in train and test set*

Figure 1 shows that in case of a 70:30 ratio, the train set covers only the first few months of the COVID-19 crisis, while the rest of the pandemic period falls inside the test set. In the case of an 80:20 ratio, the train set covers the first sharp rise in price. Furthermore, in the case of a 90:10 ratio, the train set covers the price trend characterized by a sharp rise in prices followed by a sharp decrease in Bitcoin price, and a similar trend occurs in the remaining 10% of the dataset, which corresponds to the test set. Additionally, to ensure that models have sufficient input data to gain signals, the train set includes periods before, during, and after the Bitcoin Crash.

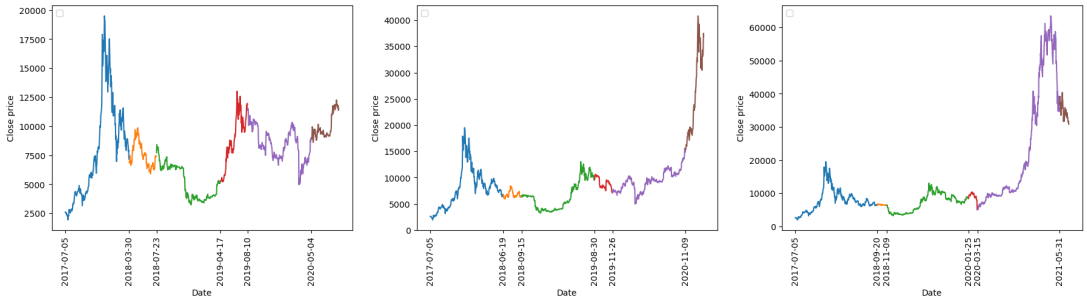
LR, SVM, and RF, are used to construct the models. The model's hyperparameters are tuned in order to achieve a higher predictive accuracy. Table 2 shows the hyperparameter values that are considered during hyperparameter optimization for each model.

Model	Hyperparameter	Values
LR	C	10^{-1}, 10^{0}, 10^{1}
	solver	newton-cg, lbfgs, liblinear, sag, saga
RF	max_features	sgrt, log2, none
	bootstrap	True, False
	max_depth	3,4,5,10,20,50
	min_samples_leaf	3,4,5
	min_samples_split	8,10,12,14,16
	n_estimators	10,30,50,100
SVM	kernel	linear, poly, rbf, sigmoid
	gamma	scale, auto

Table 2: *Models hyperparameters values*

The grid search method is used to optimize hyperparameters. This approach involves exhaustive parameter value searches for each estimator. Usually, the grid search method is combined with a cross-validation strategy. Cross-validation can prevent overfitting and help evaluate the model performance more robustly than a simple train-test approach [7]. Due to the time series nature of the observed forecasting problem, the blocking time series split is used as the cross-validation splitting strategy. This strategy has an input parameter that specifies the number of splits. Each split contains a training part and a validation part. The primary advantage is that there is no randomization during the splitting, as the timestamps of all test set elements must be exclusively after the training set elements. Furthermore, there is no overlap between splits neither between the train and validation sets. In this study, the number of cross-validation splits employed is three. Since there are only 1642 points in the complete dataset,

increasing the number of splits would result in a dataset that is insufficient for training models. Figure 2 shows the separation of the training dataset into three distinct cross-validation subsets, specifically three sets consisting of both training and validation parts, for each of the selected ratios. The proportion of the training and validation parts is equivalent to the corresponding ratio of the training and test sets.



(a) 70:30 ratio (b) 80:20 ratio (c) 90:10 ratio

Figure 1: *Bitcoin daily close price movement in cross-validation sections of train set*

For each algorithm and hyperparameter combination, the model is fitted on the training set, tested on the validation set, and the accuracy values for model performance are saved. The procedure is performed on each cross-validation section pair. For each algorithm and hyperparameter combination, the average value of the model's accuracy for three cross-validation sections is obtained. The combination of hyperparameters that yielded the highest average value of the model's accuracy is defined as the best model for each algorithm. This part of the analysis identified the three best models, specifically, the LR best model, the RF best model, and the SVM best model, for each of three observed train-test ratios, and the values of hyperparameteres are presented in Table 3. The best models are fitted on the entire corresponding train set and tested on the test set. The obtained accuracies are shown in Table 4.

Model	Train-test ratio Hyperparam.	90:10	80:20	70:30
SVC	Kernel	poly	sigmoid	linear
	Gamma	scale	scale	scale
RF	Max features	log2	log2	sqrt
	Bootstrap	TRUE	TRUE	TRUE
	Max depth	4	20	20
	Min samples leaf	4	4	3
	Min samples split	8	10	16
	N estimators	10	10	50
LR	C	0,1	10	0,1
	Solver	newton-cg	sag	newton-cg

Table 3: *Hyperparameteres of models with highest accuracy on test set*

Train-test ratio	90:10			80:20			70:30		
Model	LR	RF	SVM	LR	RF	SVM	LR	RF	SVM
Train accuracy	55,86	66,28	61,34	56,74	93,91	46,23	55,7	98,96	55,87
Test accuracy	52,73	50,3	49,09	49,85	50,15	48,94	52,94	55,17	52,94

Table 4: *Accuracies of the best models for three observed train-test ratios*

The results presented in Table 4 indicate that all three algorithms performed optimally when trained on 70% of the dataset and subsequently tested on the remaining 30% of the dataset. Additionally, the RF model exhibited the highest accuracy, achieving a rate of 55.17%. In contrast, all algorithms performed poor when evaluated with a train-test ratio of 80:20. The finding that the best result is achieved by employing a ratio of 70:30, rather than ratios of 90:10 or 80:20, confirms that incorporating excessive information, without the careful selection of appropriate variables, does not lead to improved results.

In addition, the LR and RF models are used to rank the independent variables based on their impact on the direction of the daily Bitcoin price. Table 5 shows the ranking of the independent variables based on the LR and RF best models, considering three different train-test ratios.

Train-test ratio	90:10		80:20		70:30			
Ind. var. / Model	LR	RF	LR	RF	LR	RF	Perc.	Ranks
Open	8	**9**	**13**	**2**	10	**3**	50%	8
High	19	26	**2**	**21**	**19**	**19**	50%	8
Low	22	**11**	27	**18**	26	**16**	50%	8
Volume	13	**12**	**10**	**5**	**2**	24	67%	4
Market Cap	**4**	**10**	**7**	**12**	**4**	25	83%	1
Av. Block Size	11	**6**	22	**1**	18	**13**	50%	8
Av. Block Time	14	27	26	26	22	**17**	17%	27
Av. Hash Rate	25	18	**5**	24	21	**1**	33%	20
Av. Mining Difficult	**2**	21	**1**	27	**3**	27	50%	8
Av. Transaction Fee	23	**8**	19	**9**	17	**8**	50%	8
Nmb. Of Transactions	21	25	23	**10**	15	**20**	33%	20
Tweets	20	**3**	18	**8**	14	**14**	50%	8
TNX	9	17	**4**	23	13	**10**	33%	20
VIX	26	**4**	**11**	**11**	25	23	50%	8
SP500	10	22	**3**	**6**	9	**18**	50%	8
5_MA_Close	27	**7**	25	**14**	27	**2**	50%	8
10_MA_Close	17	24	24	**7**	16	**7**	33%	20
15_MA_Close	12	16	17	**13**	11	**11**	33%	20
30_MA_Close	15	**5**	21	**17**	23	**22**	50%	8
90_MA_Close	6	**14**	**9**	**20**	**8**	**5**	83%	1
180_MA_Close	7	**2**	12	**19**	**6**	**4**	67%	4
MACD_12_26_9	24	**1**	**6**	**15**	24	**12**	67%	4
RSI_14	**3**	15	16	**22**	**5**	26	50%	8
STOCHk_14_3_3	16	20	20	**3**	20	**15**	33%	20
STOCHd_14_3_3	18	23	15	**4**	12	**6**	33%	20
ADX_14	**1**	**13**	14	**16**	**1**	**9**	83%	1
Return_1Lag	**5**	19	**8**	25	**7**	**21**	67%	4

Table 5: *The rankings of independent variables based on the LR and RF optimal models*

According to the model with the highest accuracy on the test set, which is the RF model in the case of a 70:30 train-test split, the ten independent variables that have the highest ranks are: average hash rate, 5_MA_Close, open price, 180_MA_Close, 90_MA_Close, STOCHd_14_3_3, 10_MA_Close, average transaction fee, ADX_14, and TNX. When considering the ranking with LR, and evaluating the model with the highest accuracy on the test set, which was also examined using a 70:30 train-test split, the top ten independent variables are as follows: ADX_14, volume, average mining difficult, market capitalization, RSI_14, 180_MA_Close, Return_1Lag, 90_MA_Close, S&P500, and open price. The two models with the highest accuracies on test sets rank the open price and the three technical indicators among the top ten variables that influence the direction of Bitcoin price.

According to these results, each category of independent variables has at least one representative in the top ten ranks. The only exception is the sentiment analysis category. Both best models place Tweets at the fourteenth place in terms of their impact on the direction of the Bitcoin price. On the other hand, variables from the category of technical indicators are the most common. In more detail, ADX as well as MA variables with the two largest sizes of rolling windows are in the top ten independet variables according to both LR and RF models. Additionally, MA variables with the two lowest sizes of rolling windows also took place in the first ten ranks from the perspective of the RF model. Another common independent variables within the first ten ranks is Bitcoin's open price.

In order to determine a reasonable threshold between more and less important independent variables, for each of the two ranking methods, the smallest subsets of a set of independent variables that resulted in the highest test set accuracy are determined. This phase of analysis refers to the problem of dimensionality reduction. Table 6 presents the optimal numbers of independent variables identified by the optimal LR and RF models, for all three train-test ratios.

Train-test ratio	90:10		80:20		70:30	
Model	LR	RF	LR	RF	LR	RF
Number of best ind.var.	5	14	11	22	8	22

Table 6: *The optimal numbers of independent variables identified by the optimal LR and RF models*

From the results in Table 6, it is evident that the optimal numbers of independent variables for RF models are two or more times greater than the optimal numbers for LR models.

The independent variables that belong to optimal sets are highlighted in bold in Table 5. When examining the best independent variables for LR and RF models over various train-test ratios, it becomes evident that RF models tend to identify a higher number of market and blockchain variables as optimal, compared to LR models. Furthermore, despite the fact that the RF and LR models do not share any optimal variables in the blockchain category for the same train-test ratio, five of the six models agreed that some blockchain variable is among the top three variables in terms of its influence on the next day's Bitcoin price direction. In terms of the two models with the best test accuracies shown in Table 4, it is evident that the RF model ranked the average hash rate as the foremost influential variable, whereas the LR model ranked average mining difficulty as the third most influential variable. These findings contradict the results reported in [22], which did not find any significant impact of confirmation time, hash rate, and the number of transactions per day on the price of Bitcoin. However, they also contradict the findings of [16] and [9], which indicate a small but positive effect of blockchain variables on Bitcoin's price.

The independent variables that are common to all six optimal sets of independent variables are mainly from technical indicators category. The percentage frequency of a particular

independent variable in the optimal sets is displayed in the seventh column of Table 5. The rankings of independent variables based on their percentage frequency within the optimal sets are presented in the final column of Table 5. Based on the rankings, it can be observed that the independent variables that frequently appear in the optimal sets are two technical indicators, namely ADX_14, 90_MA_Close, along with market variable market capitalization, which are present in five out of six optimal sets. This is followed by three technical indicators, namely Return_1Lag, MACD_12_26_9, 180_MA_Close, along with market variable volume, which are present in four out of six optimal sets. Although the results of Jaquart et al. [13] indicate the importance of technical indicators for minute forecasts, the results of this study highlight their importance in daily forecasting as well.

In the context of macro-finance variables, the significance of these variables is dependent on the train-test ratio. The optimal sets for a 90:10 train-test ratio, which encompass the up and down cycle of Bitcoin prices during the onset of the COVID crisis, do not incorporate TNX and S&P500. In the case of an 80:20 train-test ratio, it appears that the models attained their lowest accuracies on the test set. It can be seen that the optimal sets of independent variables largely included macro-finance variables, with the exception of TNX, which was excluded from the optimal set of the RF model. In the case of a 70:30 train-test split, where the models obtained the highest performance, it is observed that the optimal choice of variables for LR does not include any macro-finance variable. However, the optimal set for RF includes TNX and S&P500. Despite the inclusion of a greater number of macro-finance factors in optimal sets inside RF models compared to LR models, there remains a lack of clear consensus amongst these models regarding the significance of such variables. Similary, since all optimal sets of RF models include Tweets, while neither one optimal set of LR models do not includes, the RF and LR did not meet agreement about the importance of Tweets. Although, it is contrary to previous research that sees attractiveness as the main driver of Bitcoin price ([22], [10], [24], [23]), since the models did not meet agreement about the importance of Tweets, the results of the study do not provide a general conclusion on this matter.

To analyze the connection between problem dimensionality reduction and increasing the model's performance, the LR and RF best models are trained and tested on sets including only variables from corresponding optimal set of independent variables. For each train-test ratio, the third model, SVM best model is also tested on two distinct optimal sets defined with LR and RF optimal independent variables. All obtained accuracies are shown in Table 7.

Ratio	Model	All ind. var.		LR optimal set		RF optimal set	
		Train	Test	Train	Test	Train	Test
90:10	LR	55,86	52,73	54,57	**53,33**	/	/
	RF	66,28	50,3	/	/	64,93	**58,18**
	SVM	61,34	49,09	55,38	**50,91**	57,68	48,48
80:20	LR	56,74	49,85	57,12	**54,41**	/	/
	RF	93,91	50,15	/	/	92,16	**54,41**
	SVM	46,23	48,94	46,23	48,94	46,23	48,94
70:30	LR	55,7	52,94	54,83	**53,75**	/	/
	RF	98,96	55,17	/	/	98,96	**57**
	SVM	55,87	52,94	55,27	**53,75**	54,31	51,52

Table 7: *Models performances before and after dimensionality reduction*

Results in Table 7 show that dimensionality reduction and feature engineering resulted in the higher performance of models for all types of algorithms. On average, the implementation of dimensionality reduction resulted in a 2.5 percentage point increase in test accuracy. The

greatest increase in test accuracy is observed with the RF model employing a train-test split of 90:10. By employing an optimal set of independent variables, the test accuracy is enhanced from 50.3 percent to 58.18 percent, an increase of nearly 8 percentage points. As a result, this model took the first place in terms of test set performance. This confirms finding of Pabuccu et al. [20] but it is contrary to the findings of Chen et al. [6] and Akyildirim et al. [2] where the other models performed better than the RF model. The implementation of an optimal set of independent variables also enhanced the test accuracy of the RF model, which had previously achieved the highest performance (the one with the 70:30 ratio), but is currently ranked second. Applying the third model, based on the SVM classifier, on optimal sets of independent variables, resulted in higher accuracy than when applied to the initial set of independent variables. Moreover, the SVM models achieved the highest accuracies for the LR optimal sets of independent variables. The only exception is the model employing an 80:20 ratio, wherein the accuracies remained constant across several sets of independent variables. This could be due to the train set's last part covering a time of significant growth in Bitcoin price. As a result, the algorithm may have overestimated the price movements within the test set periods when the price actually decreased. The previous results show that ranking independent variables obtained with LR and RF models can be useful for dimensionality reduction for models based on other algorithms. The obtained results regarding dimensionality reduction highlight a significant aspect within the context of price forecasting. At the beginning of the analysis, the models were trained using a dataset that encompassed all observed independent variables that may potentially be associated with the dependent variable, namely the direction of the next day's bitcoin price. Additionally, the three train-test ratios were observed. It was assumed that the model, which was trained on 90% of the available data and was tested on another 10% of data, would exhibit superior performance. Firstly, because the large train set had a broad range of dates, including the COVID-19 crisis, and contained numerous independent variables that might provide valuable information to the models and additionally, because the reduced size of the test set decreases the likelihood of errors. However, the results achieved were contrary to the expected results. The best performances were achieved by RF and LR models that were trained on 70% of the data that covered the pre-COVID-19 period. These models were subsequently tested in a period marked by the crisis, which exhibited significantly higher price levels and volatility than the period covered by train set. However, after the set of independent variables had been optimised, the expected results were achieved. The analysis produced a maximum test accuracy of 58.18%, achieved using the RF model with a train-test ratio of 90:10. This finding confirms that the most optimal outcomes can be achieved by incorporating influential variables, as determined by the relevant model, and by considering a broad time range, to train models, that encompasses a period characterized by a comparable market trend to the one being forecasted. Moreover, the fact that the generally highest level of accuracy is attained when using a 90:10 ratio supports the theory that machine learning models exhibit superior performance when predicting shorter time horizons in the future.

4. Conclusion

This study predicts the directions of Bitcoin prices for the following day using three ML classifiers: LR, RF, and SVM. Models are trained and tested for different train-test ratios, and grid-search with blocking time series split cross-validation is used to optimize hyperparameters. When considering all independent variables, all three algorithms performed best when trained on 70% of the dataset and tested on 30%. In addition, the RF model has the best accuracy rate of 55.17%. The primary objective of this research was to use LR and RF models to explore how independent variables affect daily Bitcoin price direction to improve model accuracy and to reduce dimensionality by producing optimal independent variables. Technical indicators like ADX, moving averages with 90 and 180 day rolling windows, and return influence

the daily Bitcoin price the most. Despite the fact that the RF and LR models do not share any optimal blockchain variables for the same train-test ratio, five out of six models agreed that some blockchain variable influences the next day's Bitcoin price direction. According to the RF model with the highest test accuracy, average hash rate is the most influential variable, while the LR model with the highest test accuracy ranks average mining difficulty third. Five of six models placed market variables in the top four variables influencing Bitcoin price direction for the next day. The RF and LR models did not agree on how sentiment analysis and macro-finance variables affected Bitcoin prices. LR models mostly do not consider these factors important, while RF models depend on the train-test ratio. Dimensionality reduction and feature engineering improved models' performance for all classifiers. The fact that SVM performed better on the test set when the LR optimal sets of independent variables were considered shows that the analysis of individual variables' influence on the Bitcoin price direction is not limited to the models. The RF model with a 90:10 train-test split improves test accuracy the most. An optimal selection of independent variables increases test accuracy from 50.3% to 58.18%, which is the highest accuracy achived during the analysis. This finding confirms that the optimal outcomes can be achieved by incorporating influential variables, as determined by the relevant model, and by considering a broad time range, to train models, that encompasses a period characterized by a comparable market trend to the one being forecasted. Moreover, the fact that the generally highest level of accuracy is attained when using a 90:10 ratio supports the theory that machine learning models exhibit superior performance when predicting shorter time horizons in the future.

References

[1] Achelis, S. B. (2014). *Technical analysis from a to z*. McGraw-Hill.

[2] Akyildirim, E., Goncu, A., and Sensoy, A. (2020). Prediction of cryptocurrency returns using machine learning. Annals of Operations Research, 297(1-2), 3–36. doi: 10.1007/s10479-020-03575-y

[3] Alpaydin, E. (2014). *Introduction to machine learning* (3rd). The Mit Press.

[4] Borges, T. A., and Neves, R. F. (2020). Ensemble of machine learning algorithms for cryptocurrency investment with different data resampling methods. Applied Soft Computing, 90, 106187. doi: https://doi.org/10.1016/j.asoc.2020.106187

[5] Breiman, L. (2001). Random forest. Machine Learning, 45(1), 5–32. doi: 10.1023/a:1010933404324

[6] Chen, Z., Li, C., and Sun, W. (2020). Bitcoin price prediction using machine learning: An approach to sample dimension engineering. Journal of Computational and Applied Mathematics, 365, 112395. doi: https://doi.org/10.1016/j.cam.2019.112395

[7] Čorić, R., Matijević, D., and Marković, D. (2023). Pollennet - a deep learning approach to predicting airborne pollen concentrations. Croatian operational research review, 14(1), 1–13. doi: 10.17535/crorr.2023.0001

[8] Fang, F., Ventre, C., Basios, M., Kanthan, L., Martinez-Rego, D., Wu, F., and Li, L. (2022). Cryptocurrency trading: A comprehensive survey. Financial Innovation, 8(1). doi: 10.1186/s40854-021-00321-6

[9] Georgoula, I., Pournarakis, D., Bilanakos, C., Sotiropoulos, D. N., and Giaglis, G. M. (2015). Using time-series and sentiment analysis to detect the determinants of bitcoin prices. SSRN Electronic Journal. doi: 10.2139/ssrn.2607167

[10] Goczek, Ł., and Skliarov, I. (2019). What drives the bitcoin price? a factor augmented error correction mechanism investigation. Applied Economics, 51, 1–18. doi: 10.1080/00036846.2019.1619021

[11] Guo, T., Bifet, A., and Antulov-Fantulin, N. (2018). Bitcoin volatility forecasting with a glimpse into buy and sell orders. 2018 IEEE International Conference on Data Mining (ICDM). doi: 10.1109/icdm.2018.00123

[12] Ibrahim, A., Kashef, R., and Corrigan, L. (2021). Predicting market movement direction for bitcoin: A comparison of time series modeling methods. Computers & Electrical Engineering, 89, 106905. doi: https://doi.org/10.1016/j.compeleceng.2020.106905

[13] Jaquart, P., Dann, D., and Weinhardt, C. (2021). Short-term bitcoin market prediction via machine learning. The Journal of Finance and Data Science, 7, 45–66. doi: 10.1016/j.jfds.2021.03.001

[14] Khedr, A., Arif, I., P V, P., El-Bannany, M., Alhashmi, M., and Sreedharan, M. (2021). Cryptocurrency price prediction using traditional statistical and machine-learning techniques: A survey. Intelligent Systems in Accounting, Finance and Management, 28. doi: 10.1002/isaf.1488

[15] Kristjanpoller, W., and Minutolo, M. C. (2018). A hybrid volatility forecasting framework integrating garch, artificial neural network, technical analysis and principal components analysis. Expert Systems with Applications, 109, 1–11. doi: 10.1016/j.eswa.2018.05.011

[16] Kristoufek, L. (2015). What are the main drivers of the bitcoin price? evidence from wavelet coherence analysis. PLOS ONE, 10(4). doi: 10.1371/journal.pone.0123923

[17] Menze, B. H., Kelm, B. M., Masuch, R., Himmelreich, U., Bachert, P., Petrich, W., and Hamprecht, F. A. (2009). A comparison of random forest and its gini importance with standard chemometric methods for the feature selection and classification of spectral data. BMC Bioinformatics, 10(1). doi: 10.1186/1471-2105-10-213

[18] Mudassir, M., Bennbaia, S., Unal, D., and Hammoudeh, M. (2020). Time-series forecasting of bitcoin prices using high-dimensional features: A machine learning approach. Neural Computing and Applications. doi: 10.1007/s00521-020-05129-6

[19] Muhammad, I., and Yan, Z. (2015). Supervised machine learning approaches: A survey. ICTACT Journal on Soft Computing, 05(03), 946–952. doi: 10.21917/ijsc.2015.0133

[20] Pabuccu, H., Ongan, S., and Ongan, A. (2020). Forecasting the movements of bitcoin prices: An application of machine learning algorithms. Quantitative Finance and Economics, 4(4), 679–692. doi: 10.3934/qfe.2020031

[21] Peng, Y., Albuquerque, P. H., Camboim de Sa, J. M., Padula, A. J., and Montenegro, M. R. (2018). The best of two worlds: Forecasting high frequency volatility for cryptocurrencies and traditional currencies with support vector regression. Expert Systems with Applications, 97, 177–192. doi: 10.1016/j.eswa.2017.12.004

[22] Poyser, O. (2017). Exploring the determinants of bitcoin's price: An application of bayesian structural time series. https://arxiv.org/abs/1706.01437 [Accessed 14/05/22].

[23] Šestanović, T. (2021). Bitcoin direction forecasting using neural networks. The 16th International Symposium on Operational Research SOR'21, Proceedings, 557–562.

[24] Spilak, B. (2018). *Deep neural networks for cryptocurrencies price prediction* (H. W. Karl and S. Lessmann, Eds.; Doctoral dissertation).

[25] Wilder, J. W. (1978). *New concepts in technical trading systems*. Trend Research.

[26] Yamin, M. A., and Chaudhry, M. (2023). Cryptocurrency market trend and direction prediction using machine learning: A comprehensive survey. doi: 10.22541/au.167285886.66422340/v1