Forecasting Cryptocurrency Prices Using Ensembles-Based Machine Learning Approach

Vasily Derbentsev
Informatics and Systemology Department
Institute Information Technologies in Economics
Kyiv National Economic University named after Vadim Hetman
Kyiv, Ukraine
derbv@kneu.edu.ua
http://orcid.org/0000-0002-8988-2526

Natalia Datsenko
Department of Economic Mathimatical Modelling
Institute Information Technologies in Economics
Kyiv National Economic University named after Vadim Hetman
Kyiv, Ukraine
d_tashakneu@ukr.net
http://orcid.org/0000-0002-8239-5303

Vitalina Babenko
Department of International e-Commerce and Hotel&Restaurant Business
V. N. Karazin Kharkiv National University
Kharkiv, Ukraine
vitalinababenko@karazin.ua
http://orcid.org/0000-0002-4816-4579

Olha Pushko
Economic Cybernetics Department
Sumy State University
Sumy, Ukraine
o.syniavska@uabs.sumdu.edu.ua
https://orcid.org/0000-0002-7507-3541

Oleg Pursky
Department of Computer Science
Kyiv National University of Trade and Economics
Kyiv, Ukraine
pursky_o@ukr.net
http://orcid.org/0000-0002-1230-0305

2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T) | 978-1-7281-9177-5/20/$31.00 ©2020 IEEE | DOI: 10.1109/PICST51311.2020.9468090

***Abstract* — This paper is devoted to the problems of the short-term forecasting cryptocurrencies time series using machine learning approach. We applied two the most powerful ensembles methods: Random Forests (RF) and Gradient Boosting Machine (GBM). For testing models we used the daily close prices of three the most capitalized coins: Bitcoin (BTC), Ethereum (ETH) and Ripple (XRP), and as a features were selected the past price information and technical indicators (moving average). To check the efficiency of these models we made out-of-sample forecast for three cryptocurrencies by using one step ahead technique. As the accuracy rate for our models we were selected Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) metrics. According to comparative analysis of the predictive ability of the RF and GBM both models showed the same order of accuracy for the out-of-sample dataset prediction, although boosting also was somewhat more accurate. Computer experiments have confirmed the feasibility of using the machine learning ensembles approaches considered for the short-term forecasting of cryptocurrencies time series. Built models and their ensembles can be used as the basis algorithms for automated Internet trading systems.**

Keywords—cryptocurrency prices, short-term forecasting, Machine Learning, Random forest, Gradient boosting.

I. Introduction

Over the last decade, new assets have appeared in the financial market - digital currencies based on blockchain technology. Although the cryptocurrency market cannot be compared with traditional financial markets (stocks, currencies) in market capitalization and trading volume yet, many traders and investors consider them as an alternative short-term speculative financial assets.

Therefore, to predict the prices of cryptocurrencies, it is necessary to develop adequate forecasting tools. An analysis of recent papers shows that the price dynamics of cryptocurrencies is influenced by many latent factors that are still not fully identified and investigated. The majority of researchers believe that the dynamics of cryptocurrency prices is determined mainly by the balance of supply and demand and consider them as speculative financial assets [1-5].

It should be noted that common way to forecasting financial time series is to use different statistical and econometric approaches, first of all, ARIMA-GARCH time series models and their modifications [6].

However, the use of these models for forecasting cryptocurrency prices has limited applicability, since these time series characterized by non-stationarity, non-Gaussian distributions, heavy tails, high volatility and the presence of critical events [7, 8].

Recently, machine learning (ML) approaches have been actively applied for the forecasting financial time series, which have shown their efficiency in many other areas: texts and images recognition, detection of anomalies, and search for hidden relationships in data [9-13].

978-1-7281-9177-5/20/$31.00 ©2020 IEEE

II. Literature Review and Problem Statement

A. Analysis of literary data and problem statement

Among machine learning methods, artificial neural networks (ANN) of various architectures and support vector machine (SVM) are the most widespread. Resent time they have been widely used for financial time series forecasting.

Some papers shown that these approaches have better accuracy for forecasting both "traditional" financial assets (stock indices, currency exchange rates, commodity prices) and cryptocurrencies [14-21].

Thus in [11-17] it is shown that ANNs have better predictive properties than time series models and other ML algorithms for financial time series forecasting problems.

Hitam and Ismail [21] compared forecasting performance of different ML algorithms by using cryptocurrency time series (prices). They tested ANNs, SVM and Boosted NN for top-six cryptocoins. As to their results SVM has better predictive accuracy (in the terms of MAPE).

For instance, Peng et al. [22] provided examples of effective use of SVM in forecasting volatility fiat- and crypto currencies.

Other powerful class of ML based on using ensembles of regression and classification trees: Random Forest (RF) [23-24] and Gradient Boosting Machine (GBM) [25-26]. Through the use of bagging (RF) and boosting (GBM) these techniques have been shown high performance in many areas of ML applications.

There are some empirical research papers devoted to application of ensembles approaches for financial time series analysis (see, for example [19-20, 27-33]).

Thus, Varghade and Patel [20] tested RF and SVM to forecasting stock market index S&P. They showed that the Decision Trees model outperforms the SVR, although RF at times is found to overfit the data.

Madan et al. [31] explored daily Bitcoin data. They used three ML methods: logistic regression, SVM and RF. As to their results RF provide forecast close to the original data.

Sun et al. [32] applied GBM for prediction changes in the price trend of the cryptocurrencies. As to their results GBM have powerful predictive ability as other ML and statistical approaches.

B. The purpose and objectives of the study

The purpose of our study is to compare the prognostic properties for the short-term prediction cryptocurrency prices such ML methods as Random Forest (RF) and Gradient Boosting Machine (GBM).

III. Methodology

A. Random forest

RF developed by Breiman et al. [23-24] is one of the most powerful ML approach. This method is based on using compositions of weak classifiers. At the same time, due to large number of classifiers, the final result of classification or regression task is satisfactory. RF based on bagging and bootstrap sampling.

Training dataset is divided into many random subsets with replacement (bootstrap samples) and individual classifiers are trained on different sub-datasets.

The final classifier $a_N(\mathbf{x},\theta)$ is built as the average of the basic algorithms $h_i(\mathbf{x})$ (for regression):

$$a_N(\mathbf{x},\theta)=\frac{1}{N}\sum_{i=1}^{N}h_i(\mathbf{x},\theta), \tag{1}$$

where N – is the number observations (samples) in training set $(\mathbf{x}_i, y_i)_{i=1,2,\ldots,N}$, θ - is vector of unknown parameters.

It should be noted that RF uses bagging and all elementary classifiers are trained independently. Basic classifiers must be uncorrelated. In this case, we can use classifiers based on different groups of methods: logistic regression, classification and regression trees (C&RT), ANNs and so on. On the other hand, we can train models on independent data sets, in this case we can use the same base algorithm.

Another aspect that determines the efficiency of RF is that the feature used for branching in the certain node is not selected from all possible features set, but only from their random subset of size *m*.

Moreover, they selected randomly, typically with replacement. Tthat's why the same feature can appear several times, even in one branch. Number of features for training each tree recommended to choose as $m\approx\frac{M}{3}$ (for regression task), where *M* is total number of features.

RF uses fully grown decision trees. It tackles the error reduction task by reducing variance. The trees are made uncorrelated to maximize the decrease in variance, but the algorithm cannot reduce bias (which is slightly higher than the bias of an individual tree in the forest).

It should be noted that in RF are used deep trees because basic algorithms require low bias; the spread is eliminated by averaging the responses of various trees.

B. Gradient Boosting Machine

Other popular ML ensembles approaches which have been widely used over past two decades is gradient boosting. CBM does not use simple voting but a weighted one (in contrast to bagging).

GBM like RF is also based on ensembles of weak classifiers, as a rule, these are shallow decision trees, sometimes even as small as decision stumps (trees with two leaves).

Following Friedman [25-26] let's consider the basic steps of GBM. By analogy with RF we can use a weighted sum of N basic algorithms

$$a_N(\mathbf{x})=\frac{1}{N}\gamma_N\sum_{i=0}^{N}h_i(\mathbf{x},\theta). \tag{2}$$

We will also assume that initial algorithm $h_0(\mathbf{x},\theta)$ has been defined, for example, in such form:

$$h_0(\mathbf{x},\theta)=\frac{1}{l}\sum_{i=1}^{l}y_i, \qquad (3)$$

where – l is the number of samples in training set ($l<n$, as a rule $l\approx 0.7\div 0.8n$).

Let we have already built ensemble $a_{N-1}(\mathbf{x},\theta)$ of $N-1$ classifier on the $N-1$ step. Then we can select the next basic algorithm $h_N(\mathbf{x})$ which reduced the error given by (4) as much as possible:

$$\sum_{i=1}^{l}\left[L\left(y_i,a_{N-1}(\mathbf{x}_i,\theta)+\gamma_N h_N(\mathbf{x}_i,\theta)\right)\right]\to\min_{\gamma_N,h_N}. \qquad (4)$$

If we choose $h_N(\mathbf{x}_i.\theta)$ in the following form

$$h_N(\mathbf{x},\theta)=\underset{h(\mathbf{x},\theta)}{\arg\min}\sum_{i=1}^{l}\left(h(x_i,\theta)-s_i\right)^2. \qquad (5)$$

Further by selecting the deviation s_i equal to the anti-gradient of the Loss-function $L(\cdot)$

$$s_i=-\left.\frac{\partial L(\cdot)}{\partial z}\right|_{z=a_{N-1}(x_i)} \qquad (6)$$

we will take one step of the gradient descent (GD). GD proved moving towards the fastest decrease of the Loss.

So we carry out GD in the l-dimensional space of algorithm predictions on the samples of the training dataset.

As soon as a new basic algorithm is constructed, it is possible to find its coefficient γ_N :

$$\gamma_N=\underset{\gamma}{\arg\min}\sum_{i=1}^{l}L\left(y_i,a_{N-1}(\mathbf{x}_i,\theta)+\gamma_N h_N(\mathbf{x}_i,\theta)\right) \qquad (7)$$

Note, that unlike the RF, GBM is not so susceptible to overfitting, which leads to increase errors on the test sample or on hold out data. That's why, trees with a small depth are usually used in boosting.

These trees have a large bias, but don't result to overfitting. One way to solve this problem is to reduce the step: instead of moving to the optimal point in the direction of the anti-gradient, a shortened step is taken by

$$a_N(\mathbf{x},\theta)=a_{N-1}(\mathbf{x},\theta)+\lambda\left(\gamma_N h_N(\mathbf{x})\right), \qquad (8)$$

where $\lambda\in[0,1]$ is the learning rate.

The main reason for the efficiency of boosting is that the algorithm builds a basic algorithm at each iteration, which is really effective only on a part of the subsample.

This principle can be enhanced by making a slight modification to the GBM. Namely, at each step of the algorithm, a new term is considered based not on the entire training sample, but only on a random subsample of a fixed size.

This idea is very popular and efficient. It is a combination of gradient boosting and bagging techniques.

We can also take not a random subsample of objects, but also a random subsample of the features of objects. This is called the random subspace technique.

The results of the work of such modifications are often noticeably superior in quality to various non-stochastic options.

It should be noted that both described above ensembles approaches (RF and GBM) have own advantages and disadvantages. RF is preferable for short training sets, it also allows efficient parallelization of calculations, while boosting is performed strictly sequentially.

The main advantages of RF based on bagging are:

- The ability to efficiently process data with a large number of features and classes.
- Insensitivity to scaling (and generally to any monotonous transformations) of feature values.
- Both continuous and discrete features are handled equally well. There are methods for constructing trees from data with missing feature values.
- There are methods for evaluating the significance of individual features in a model.
- Internal assessment of the generalizability of the model (out-of-bag test for unselected samples).
- High parallelizability and scalability.

GBM better applied on large training datasets. It also better reproduces the boundaries of classes of complex shape.

As to main advantages of GBM they are:

- First, different loss functions can be considered. This allows solving both classification problems and regression problems.
- Secondly, like in RF it is possible to consider any family of basic algorithms. And this, again, gives ample opportunities to take into account the features of this task.
- Boosting over decision trees is considered one of the most effective boosting options. And given that the decision trees, in turn, also use basic algorithms (for example, threshold, linear, etc.), as a result, a huge number of options for tuning are obtained.
- Third, due to the sufficient simplicity of the method and a clear mathematical justification, in each specific variation of boosting it is not difficult to carry out some mathematical and algorithmic optimizations that will significantly speed up the operation of the algorithm.

IV. DATA SET

Since the main purpose of our study is to get the most accurate one-step ahead forecast of daily cryptocurrencies prices based on RF and GBM approaches, we used their past values: daily close prices and trading volumes of the three most capitalized coins: (Bitcoin, BTC; Ethereum, ETH, and Ripple, XRP).

The dataset corresponds to the period from 01/01/2015 to 31/12/2019 for BTC and XRP (1826 observations), and from 07/08/2015 to 31/12/2019 for ETH (1608 observations) according to the Yahoo Finance [34].

To provide of models design, fitting and tuning their parameters dataset was divided into the training and test subsets in the ratio of 80% and 20%. Moreover, the last 92 observations (from 01/10/2019 to 31/12/2019) were reserved for validation which was performed by out-of-sample one-step ahead forecast BTC, ETH, and XRP prices.

V. EMPIRICAL RESULT

According to some empirical studies devoted forecasting cryptocurrencies prices [6, 29], there is a seasonal lag which is a multiple of 7 if we use daily observations. The results of correlation analysis also showed the existence of statistically significant lags, multiples of 7, for selected coins.

Therefore, we used the previous values of daily prices for the last 4 weeks as predictors (lagged daily prices $y_{i-1}, y_{i-2},\dots,y_{i-28}$). To take into account the changing trends, we used moving average prices of different orders: “fast” – 3,5,7 days, and “slow” – 21, 28 and 35 days.

In addition, we also included into dataset two exogenous variables: daily trading volumes and growth rates of daily trading volume with a lag of one day. Thus, the final dataset contains 36 features.

The main disadvantage of both RF and GBM methods is the difficulty in interpreting the results. Moreover, RF is more prone to overfitting.

A. Hyper-parameters tuning

Both of methods (RF, and GBM) based on partitions the data into training and testing sets by randomly selecting cases. A GBM modification that uses such a partition is called Stochastic GBM (SGBM), which we applied in this study. The training sample is used to fitting models by adding simple trees to ensembles. Testing set is used to validate their performance. For regression tasks validation is usually measured as the average error. We select 30% of the dataset as test cases for both approaches.

We tested models whith different values of hyper-parameters and chosen values that gave the smallest error on the training set.

Since the RF is not inclined to overfitting, one can choose a large number of trees for the ensemble. We designed RF model with 500 trees. At the same time, in order for the model to be able to describe complex nonlinear patterns in data, it is necessary to use complex trees. So we have been chosen 15 the maximum number of levels.

The final values of hyper-parameters setting are reported in Table 1.

Other important parameter for RF is the number of features to consider at each split. As noted in the section III for regression task it is recommended to choose this value as $m \approx \frac{M}{3}$, where M is the total number of features. We tested different RF models with value m within 8 to 12.

As stop condition for number of trees in SGBM (boosting steps) we took the number of trees at which the error on the test stops decreasing. This is necessary in order to avoid the overfitting. For boosting, unlike the RF, the simple trees are usually used. That’s why we fitted maximum number of levels in trees and number of terminal nodes by the criteria of lowest average squared error on both training and test samples.

TABLE I. FINAL HYPER-PARAMETERS SETTING

Parameters	RF	GBM
Loss-function	quadratic	quadratic
Training / test subsamples proportion, %	70/30	70/30
Random subsample rate	0.7	0.7
Number of trees in ensemble	500	250
Maximum number of levels in trees	15	4
Maximum number of features to consider at each split	12	-
Maximum number of terminal nodes in trees	150	15
Minimum samples in child nodes	5	-
Learning rate (shrinkage)		0.1

For GBM an important parameter is a learning rate (shrinkage). Regularization by shrinkage consists in modifying the update rule (8) by tuning λ . We selected this value on the on the grid search according to minimum prediction error on the test set.

B. Forecasting performance

The short-term forecasts for selected coins were made for absolute values of prices. The target variable is the prediction the value of close prices for each cryptocurrency in the next time period (day) although we used daily observation. Both SGBM and RF models were trained with the same set of features.

For testing efficiency of both approaches, we carried prediction prices of selected coins on the hold out last 91 observations by using one-step ahead forecasting technique. The final results are shown on Fig. 1-3.

Analysis of the graphs allows us to conclude that both ensembles approaches commonly well approximate cryptocurrencies time series dynamic, but we can see a certain delay in the model graphs in comparison to real data.

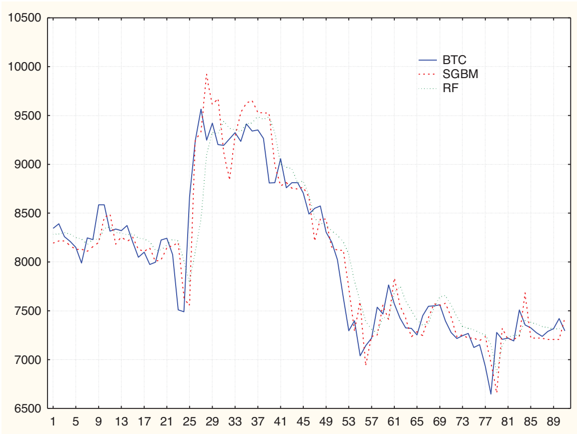

Fig. 1. Out of sample prediction BTC prices

For comparing prediction performance of different ensembles (SGBM, and RF) we applied Mean Absolute Percentage Error (MAPE) and Root Mean Square Error

(RMSE). Table 2 shows the summary of the estimation forecasting accuracy in terms of these metrics.

Thus, we can conclude that both methods have the same order of accuracy for the out-of-sample dataset prediction, although boosting also was somewhat more accurate. The best prediction performance is produced by SGBM for XRP – 0.92 %, and the best result obtained by RF is 1.84% for XRP. We also can see that SGBM outperform RF for selected coins.

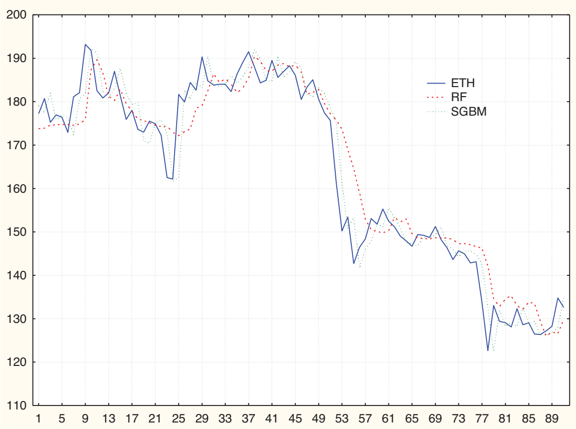

Fig. 2. Out of sample prediction ETH prices

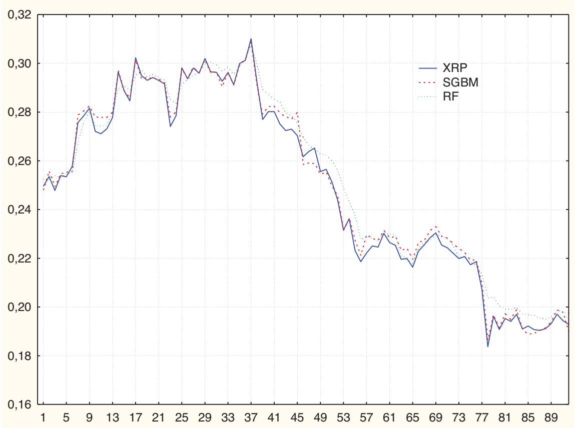

Fig. 3. Out of sample prediction XRP prices

TABLE II. OUT-OF-SAMPLE ACCURACY FORECASTING PERFORMANCE RESULTS

	SGBM		RF	
	MAPE, %	RMSE	MAPE, %	RMSE
BTC	**2.31**	263.34	2.61	305.95
ETH	3.02	5.02	**2.26**	6.72
XRP	**0.92**	0.0029	1.84	0.0057

VI. DISCUSSION OF RESEARCH RESULTS AND CONCLUSION

Our research has shown efficiency of use ML ensemble-based approaches to predicting cryptocurrencies time series. According to our results, the out of sample accuracy of short-term forecasting daily prices obtained by SGBM and RF in terms of MAPE for three the most capitalized cryptocurrencies (BTC, ETH, and XRP) was within 0.92-2.61 %.

Note, that we explored different sets of features. Our base dataset contained only past values of target variable with 14, 21 and 28 lag depth. In this case larger dataset provided better training of the SGBM and RF, and given more efficient results. Final results reported in Table 2 show that including such additional exogenios factors, as trading volumes and several technical indicators (moving averages), helped to increase accuracy rate in MAPE and RMSE metrics both on in- and out of sample datasets.

The obtained results allow us to conclude that the use of ensembles tree-based models such as RF and GBM for short-term forecasting of cryptocurrencies time series is efficient.

From our point of view, it is possible to increase the predictive power of developed models by using extended dataset. For example, it might be useful to investigate additional features: maximum, minimum, and average prices, other technical indicators and oscillators, such as, Price rate-of-change, Relative strength index, and so on. In addition, one can try to determine the influence of fundamental variables on cryptocurrecies dynamics.

Thus, our future research should extend by investigating of the predictive power both of described above and others additional features.

Summing up, we note that a promising approach for forecasting not only time series of cryptocurrencies, but also other financial time series is the development of combined ensemble of C&RT with other powerful ML models, including ANN and SVM.

REFERENCES

[1] Krugman P., “Bits and Barbarism” [Online]. Available: http://www.nytimes.com/2013/12/23/opinion/krugmanbits-and-barbarism.html.

[2] Selmi, R., Tiwari, A., Hammoudeh, S., “Efficiency or speculation? A dynamic analysis of the Bitcoin market”, *Economic Bulletin*, 38(4), pp. 2037-2046, 2018.

[3] Cheah, E., “Speculative bubbles in Bitcoin markets? An empirical investigation into the fundamental value of bitcoin”, *Economic Letters*, 130, pp. 32-36, 2015.

[4] Ciaian, P., “The economics of BitCoin price formation”, *Applied Economics*, 48(19), pp. 1799-1815, 2016.

[5] Babenko,V., Perevozova, I., Mandych, O., Kvyatko, T., Maliy, O., Mykolenko, I., “World informatization inconditions of international globalization: Factors of influence”, *Global Journal of Environmental Science and Management*, 5 (SI), 2019, pp. 172-179. doi: http://dx.doi.org/10.22034/gjesm.2019.05.SI.19

[6] Catania L., Grassi, S., “Modelling Crypto-Currencies Financial Time-Series”, *CEIS Research Paper*, 15(8), pp. 1-39. (2017). [Online]. Available: https://ideas.repec.org/p/rtv/ceisrp/417.html.

[7] Derbentsev, L. Kibalnyk, Yu. Radzihovska,“Modelling Multifractal Properties of Cryptocurrency Market using Hurst Exponent and Detrended Fluctuation Analysis”, *Periodicals of Engineering and Natural Sciences*, 7(2), pp. 690- 701, 2019.

[8] Soloviev, V., Belinskij, A., “Complex Systems Theory and Crashes of Cryptocurrency Market”, *Communications in Computer and Information Science*, 1007, pp. 276–297, 2019. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-13929-2_14.

[9] Makridakis, S., Spiliotis, E., Assimakopoulos, V., “Statistical and Machine Learning forecasting methods”, *Plos one*, March 27, pp.1-26, 2018. doi: https://doi.org/10.1371/journal.pone.0194889.

[10] Bontempi, G., Taieb, S., Borgne, Y., “Machine Learning Strategies for Time Series Forecasting”, *European Business Intelligence Summer School eBISS* pp. 62-77, 2013. Springer-Verlag. Berlin Heidelberg.

[11] Persio, L., Honchar, O., "Multitask machine learning for financial forecasting", *International Journal of Circuits Systems and Signal Processing*, 12, pp. 444-451, 2018.

[12] Kourentzes, N., Barrow, D.K., Crone, S,F., "Neural network ensemble operators for time series forecasting", *Expert Systems with Applications*, 41(9), pp. 4235-4244, 2014. doi: https://doi.org/10.1016/j.eswa.2013.12.011.

[13] Ahmed NK, Atiya AF, Gayar NE, El-Shishiny H., "An Empirical Comparison of Machine Learning Models for Time Series Forecasting", *Econometric Reviews*, 29 (5-6), pp. 594-621, 2010. doi: https://doi.org/10.1080/ 07474938.2010.481556.

[14] Kourentzes, N., Barrow, D.K., Crone, S,F., "Neural network ensemble operators for time series forecasting", *Expert Systems with Applications*, 41 (9), pp. 4235-4244. 2014. doi: https://doi.org/10.1016/j.eswa.2013.12.011.

[15] Liu W, Wang Z, Liu X, Zeng N, Liu Y, Alsaadi FE., "A survey of deep neural network architectures and their applications", *Neurocomputing*, 234 (C), pp. 11-26, 2017. doi: https://doi.org/10.1016/j. neucom.2016.12.038.

[16] McNally, S., Predicting the price of Bitcoin using Machine Learning. Doctoral dissertation, Dublin, National College of Ireland, 2016.

[17] Hamid SA, Habib A., "Financial Forecasting with Neural Networks", *Academy of Accounting and Financial Studies Journa*l, 18 (4), pp. 37-55, 2014.

[18] Boyacioglu, M., Baykan, O.K., "Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange", *Expert Systems with Applications,* 38 (5), pp. 5311-5319, 2011.

[19] Kumar, M., "Forecasting Stock Index Movement: A Comparison of Support Vector Machines and Random Forest", *SSRN Working Paper*, 2006. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=876544.

[20] Varghade, P., Patel, R, "Comparison of SVR and Decision Trees for Financial Series Prediction", *International Journal on Advanced Computer Theory and Engineering*, 1(1), pp. 101-105, 2012.

[21] Hitam, N. A., Ismail, A. R., "Comparative Performance of Machine Learning Algorithms for Cryptocurrency Forecasting". (2018). [Online]. Available: https://www.researchgate.net/publication/327415267.

[22] Peng, Y., Henrique, P., Albuquerque, M., "The best of two worlds: Forecasting high frequency volatility for cryptocurrencies and traditional currencies with Support Vector Regression", Expert Systems with Applications, 97, pp.177-192, 2018.

[23] Breiman, L., Friedman, H., Olshen, R. A., & Stone, C. J., Classification and Regression Trees. Belmont, NJ. Wadsworth International Group, 1984.

[24] Breiman, L., "Random Forests", *Machine Learning*, 45, pp. 5-32, 2001.

[25] Friedman, Jerome H., "Greedy Function Approximation. A Gradient Boosting Machine", *Annals of Statistics*, 29(5), pp. 1189-232, 2001.

[26] Friedman, Jerome H., "Stochastic Gradient Boosting", *Computational Statistics and Data Analysis*, 38, pp. 367-378, 1999.

[27] V Ji, S., Wang, X., Zhao, W., and Guo, D., "An Application of a Three-Stage XGBoost-Based Model to Sales Forecasting of a Cross-Border E-Commerce Enterprise", *Hindawi Mathematical Problems in Engineering*, 2019. doi: https://doi.org/10.1155/2019/8503252.

[28] Pavlyshenko, B.M., "Machine Learning Models for Sales Time Series Forecasting", *Data*, 4(3), pp. 1-15. 2019. doi: https://doi.org/10.3390/data4010015.

[29] Gnip, P. and Drotár, P. "Ensemble methods for strongly imbalanced data: bankruptcy prediction", IEEE 17th International Symposium on Intelligent Systems and Informatics (SISY), Subotica, Serbia, pp. 155-160, 2019. doi: https://doi.org/ 10.1109/SISY47553.2019.9111557

[30] Alessandretti, L., ElBahrawy, A., Aiello, L., Baronchelli, A., "Anticipating Cryptocurrency Prices Using Machine Learning", *Hindawi Complexity*, 2018. doi: https://doi.org/ 10.1155/2018/8983590.

[31] Madan, I., Saluja, S., and Zhao, A."Automated Bitcoin trading via machine learning algorithms", 2015. [Online]. Available: http//cs229. stanford. edu/proj2014/Isaac% 20Madan.

[32] Sun, X., Liu M., and Sima, Z."A novel cryptocurrency price trend forecasting model based on LightGBM", *Financial Reseach Letter*, 2018. doi: https://doi.org/ doi: 10.1016/j.frl.2018.12.032.

[33] Derbentsev, V., Matviychuk, A., Soloviev, V.N. "Forecasting of Cryptocurrency Prices Using Machine Learning", In: Advanced Studies of Financial Technologies and Cryptocurrency Markets, Springer, pp. 211-231, 2020. doi: https://doi.org/ 10.1007/978-981-15-4498-9.

[34] Yahoo Finance. Available: [Online]. Available: https://finance.yahoo.com.