Building of Regression Models for Cryptocurrency Price Prediction

Kirill Smelyakov [1], Oleksandr Bizkrovnyi [1], Natalia Sharonova [2], Serhii Smelyakov [1], Anastasiya Chupryna [1]

[1] *Kharkiv National University of Radio Electronics, 14 Nauky Ave., Kharkiv, 61166, Ukraine*
[2] *National Technical University "KhPI", Kyrpychova str. 2, Kharkiv, 61002, Ukraine*

Abstract

This article is an investigation of factors that can affect cryptocurrency price and their usage in regression models to determine which model type and the algorithm itself is best-suited for predicting crypto price. The determination of the best algorithm is based on an experiment that includes training and validation of models. Comparative analysis of models validation results defines the best-suited algorithm; the type of cryptocurrency that is being analyzed is Defi, namely Ethereum; the study is based on a one-year time frame; the paper does not consider political factors and factors of infrastructure destruction that may affect cryptocurrency prices. The factor types, which are used to create regression models, consist of fundamental factors. The technical factors were omitted and can be investigated in other works. Factors include: network statistics, exchange statistics, mining statistics, social statistics, transactions data, etc. The models performance is calculated by regression metrics. The JMH is used to calculate models time to train.

Keywords

Cryptocurrency, machine learning, price forecasting, prediction model, impacting factors for cryptocurrency price

1. Introduction

Cryptocurrency and bitcoin in particular, has demonstrated its value in recent years, and there are now 14 million bitcoins in circulation. Investors speculating on the future possibilities of this new technology have provided much of the current market capitalization, and this will likely continue until a certain degree of price stability and market acceptance is achieved. Beyond the announced price of a cryptocurrency, those who invest in it rely on the perceived "intrinsic value" of the cryptocurrency. This includes the technology itself and the network, the integrity of the cryptographic code and the decentralized network. Blockchain public ledger technology (the underlying cryptocurrency) is capable of disrupting a range of transactions beyond the traditional payment system. These include stocks, bonds and other financial assets whose records are stored digitally and for which there is currently a need for a trusted third party to validate the transaction. At present, a huge number of models, algorithms and technologies have been developed to improve the speed of fraud detection, mining efficiency, cybersecurity and privacy, as well as to improve the efficiency of price forecasting, volatility, portfolio volume and structure, etc. At the same time, algorithms to solve these problems are often unsustainable because they do not take into account a number of important influencing factors. In this regard, it is now relevant to make a deeper analysis of the factors that have an impact on the price formation of cryptocurrency in order to build regression models to predict prices.

COLINS-2022: 6th International Conference on Computational Linguistics and Intelligent Systems, May 12–13, 2022, Gliwice, Poland
EMAIL: kyrylo.smelyakov@nure.ua (K. Smelyakov); oleksandr.byzkrovnyi@nure.ua (O.Bizkrovnyi); nvsharonova@ukr.net (N. Sharonova); serhii.smeliakov@nure.ua (S. Smelyakov); anastasiya.chupryna@nure.ua (A. Chupryna)
ORCID: 0000-0001-9938-5489 (K. Smelyakov); 0000-0001-9335-442X (O. Bizkrovnyi); 0000-0002-8161-552X (N. Sharonova); 0000-0002-5791-2479 (S. Smelyakov); 0000-0003-0394-9900 (A. Chupryna)

2. Related Works

A cryptocurrency is a software with a specific way to use it which allows including them into the currencies market and do trading. The papers [1-3] present a modern review of Cryptocurrency systems, Models and algorithms. In particular, the work [1] has comparative analysis of mining algorithms, in [2, 3] the main Challenges and Opportunities are formulated, as well as an analysis of the basic methods of artificial intelligence, which are used to solve the most important problems of cryptocurrency, related to price forecasting, risks, cybersecurity threats and a number of others. The main idea of the "coin" type of cryptocurrency is the ability to prepare anonymous transactions [4]; work [5] shows the features of the model of decentralized confidential payment system. Furthermore (major restriction regarding the type of cryptocurrency), other types of cryptocurrencies exist, but this research orients just to the investigation of the factors that make an impact on the "coin" type of cryptocurrency [6-8].

All of the cryptocurrencies are software and their price is a difficult analysis of many factors [9]. Globally, there are two parts which contain their own factors: product itself (cryptocurrency as product and their mechanism) and trading market. Nothing lives outside the environment and crypto is not an exclusion. The relationship between the choice of factors, models and algorithms of blockchain cryptocurrency ecosystem functioning are described in papers [10-12]. Crypto is a common software, so general rules that affect any product in this area can affect cryptocurrency too. Examples of such functions are described in [13-15]: each product has competitors; each product should give some specific features to survive; each product depends on the buying ability of potential customers, etc.

A cryptocurrency is a specific software with a peculiar mechanism of their work. In general, all of the cryptocurrencies have users and transactions validators, the role that validators play miners, which mine each of the next blocks of the blockchain. As a result, there are two role needs of which need to be addressed. If users will not have the ability to use crypto coins, then crypto will die [16]. Another part of that, If the miners will not have required profit to cover all of the costs, then transactions will be approved with huge delay, which leads to decreasing popularity of crypto and as a result, this reason might be as root cause of cryptocurrency to go from the market [17-19].

The solution of these problems is decisive for the effective application of private models of artificial intelligence, machine learning and computer vision [20, 21], including using these models to improve efficiency algorithms for the formation and processing of network information [22-24].

There are many approaches that use different factors to predict the crypto price. There is an attempt to predict price using GRU, LSTM and bi-LSTM Machine Learning Algorithms [25]. This approach uses the following factors for training data set:

- Open price;
- High price;
- Low price;
- Close price;
- Date.

Factors selection is not enough confident, because the price factors do not give the root cause of values for given factors for particular date, in other words, the factors are not descriptive. Despite on this assumption the model validation process shows the following results (Figure 1, Figure2).

There is no description in the article of how the models were trained and validated, but historic price values never can be used for future price prediction.

Another work [26] uses technical metrics of cryptocurrency for price prediction. All of the article frameworks attempt to predict the Bitcoin prices starting from five technical indicators:

- Simple Moving Average (SMA);
- Exponential Moving Average (EMA);
- Momentum (MOM);
- Moving Average Convergence Divergence (MACD);
- Relative Strength Index (RSI).

One of the ML frameworks is described below (Figure 3).

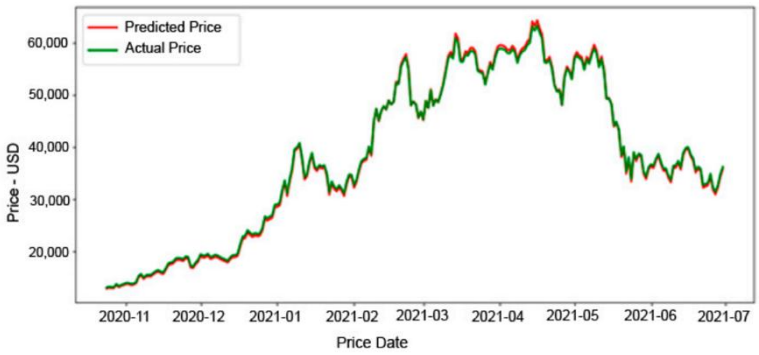

Figure 1: Actual and predicted price of BTC using the LSTM model [25]

Figure 2: Actual and predicted price of BTC using the GRU model [25]

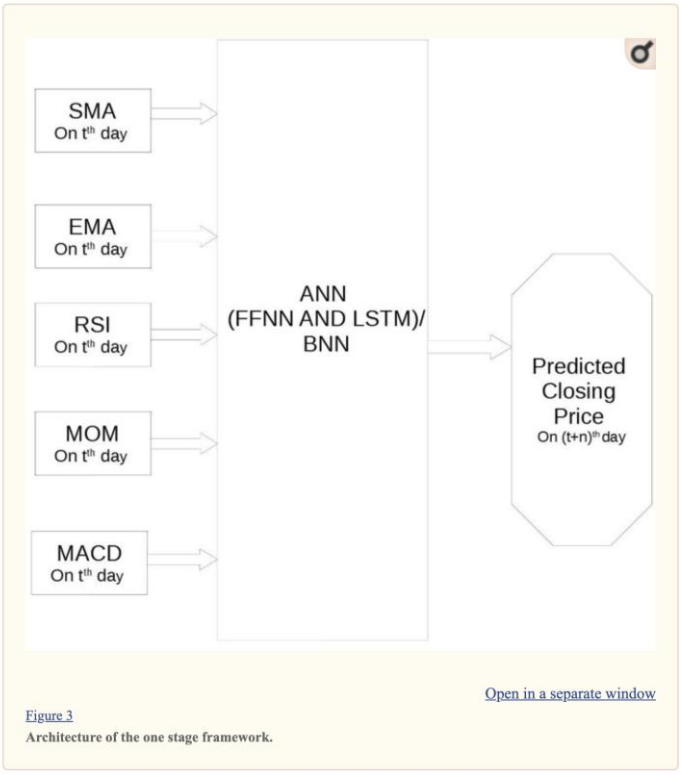

Figure 3: Architecture of one stage framework [26]

Technical indicators also cannot be a comprehensive data source for ML model training, because technical analysis does not live without fundamental analysis which includes: network statistics,

exchanges statistics, worldwide economic state, etc. The main goal of this investigation is to determine the informative factors that can be used for the price forecasting, determine the effectiveness of the usage of regression machine learning models [27] and figure out the best suited algorithm for the mentioned problem.

3. Methods and Materials

Consider initial data for the methods and experiments, metrics, factors and methods proposed to solve the problem under consideration.

3.1. Data Description

The problem is characterized by time series data because crypto metrics and worldwide economic indices are being updated each day. This dataset includes only worldwide economic data and metrics data for a particular cryptocurrency. The exclusion here is data that can be spread between countries. The crypto price is worldwide, as a result, no data per country can be used in a data set.

The factors used in the dataset are not the same dates. While regular exchanges work only 5 days a week during business hours, crypto exchanges work every day around the clock. This fact leads to a need for data preparation. Values that are found at the end of the week are used during the weekends for crypto exchanges.

Another data issue that was found is data missing in some places. This issue is resolved by deleting the full row to avoid the creation of incorrect relationships between factors. In case the dataset is large, the gaps recovering is possible, but when the dataset is small, each data row is important to create valid relationships. The dataset cannot be found on the Internet in public access. It consists of different parts that are being retrieved from different sources in CSV format and combined using "Spark" after.

Here are the examples of the data that formed the dataset (Table 1, Table 2).

Table 1
DJI index data

Date	Value
01.01.2021	495.15
02.01.2021	495.34
03.01.2021	495.15
04.01.2021	492.8
05.01.2021	495.88
06.01.2021	498.4

Table 2
Ethereum Inflow Exchanges data

Date/Time	Aggregated Exchanges	Price
01.01.2021	434993.6207041356	733.425
02.01.2021	609389.5356033011	752.49
03.01.2021	1441436.1960446832	890.94
04.01.2021	1793421.0470202232	1026.57
05.01.2021	1081812.942707662	1054.795
06.01.2021	1117149.8606023395	1136.655

The “Price” column is not used during forming the dataset, because a separate CSV file with crypto price and date exists. The dataset is available by the following link [28].

About data source. There are a huge amount of different ways for retrieving particular info regarding particular cryptocurrencies. For example, to retrieve network and mining data, users can run nodes for

a particular cryptocurrency and aggregate required information. This way is time and resource consumptions.

The social media data can be retrieved from required portals such as GitHub, Telegram, Twitter, in a direct way, but these are raw data that requires preparing for extracting data sentiment.

In general, many services avoid such resources and time costs. The "IntoTheBlock" was selected as the source data provider because this system allows a 7-day trial and already has all precomputed and aggregated data that was mentioned above.

Also, worldwide economic information for the dataset can be retrieved from the following resource [29].

3.2. Informative Factors' Selection

The mentioned above global factors give understanding which of them may signalize the price direction. There is the following list of the factors from different areas that were selected to build ML models.

3.2.1. Worldwide Data

There are factors that are not directly related to cryptocurrencies, but allegedly affect them. These are global economic factors that demonstrate global economic behavior that may affect the demand for cryptocurrency.

S&P 1200 – factor means the global economic situation in the world based on indexes of 1200 biggest companies. This factor was included as an index of investor buying ability. If the world economic situation will be better, then more investors may spend funds to buy such volatile investments as cryptocurrency.

Dow Jones Global – factor means the economic state of industrial companies. This factor also can be used as mentioned before.

3.2.2. Ethereum Data

The following factors are closely related to cryptocurrencies and their work principles. In general, these metrics show different aspects inside cryptocurrencies during their lifecycle.

Inflow volume – total amount (in $ or tokens) entering exchange(s) deposit wallets. All exchanges refer to all supported exchanges. The sharp jumps of inflows tend to coincide with and sometimes precede periods of high volatility. This can potentially be interpreted as a sign of holders looking to sell in centralized exchanges.

Outflow volume – total amount (in $ or tokens) leaving exchange(s) withdrawal wallets. All exchanges refer to all supported exchanges. Outflow Volume often spikes following either a crash or a significant break-out. This could potentially be interpreted as users going long and opting to hold their crypto outside centralized exchanges.

ETH Price – a dependent variable in the regression model. Means Ethereum price.

ETH – BTC correlation – factor is used to display correlation between prices of largest cryptocurrencies. If some product loses the buyer's confidence, then other products in that sphere may lose it too.

Large transaction – indicator shows transactions where an amount greater than $100,000 USD was transferred.

Transaction volume is USD – large transactions are those where an amount greater than $100,000 USD was transferred. In this case, the Large Transactions Volume in USD indicator measures the aggregate dollar amount transferred in such transactions. Large transaction Volume metric shows the total amount transacted by whales players in a given day. This indicator may give an idea of changes in the cryptocurrency market if huge amounts of crypto volume transfers between addresses.

Transaction count – indicator displays activity in blockchain networks which can show the general market behavior. If the transaction count increases, then popularity of the industry or particular cryptocurrency rises.

Miners' inflows – indicators may point to general miner activity and how much they earn. Huge amount of miner inflow can mean an increased need for them.

Miners' outflows – indicator may point to miner behavior, when they sell their crypto holdings into exchanges.

Miners' reward – metric describes the miner's reward. In case if reward is low, then crypto currency may be stuck with a long transaction confirmation delay. Also, if the huge reward consists of fees that users pay, then popularity of crypto may decrease.

Average transaction fees – metric can point to increasing cryptocurrency demand. In case, when a huge amount of transactions exists in the queue, customers start to pay extra fee to up their transaction confirmation.

Average transaction volume – transaction volume can indicate both trading and non-speculative activity. Similar to trading volume observed in exchanges, transaction volume can be useful for identifying reversals and breakouts.

GitHub activity is a couple of indicators that refer to this: opened issues, closed issues, watchers count, forks count, opened and closed pull requests count. This may point to an idea of how the cryptocurrency quickly grows.

Search trends – indicates how often cryptocurrency rises to the spotlight. The increased attention may indicate upcoming market moves.

Telegram sentiment – indicator helps measure traders' emotions. In the case of bitcoin, positive sentiment on Telegram has on several occasions preceded a price movement, as seen in December 2019, April, and June 2020. At the same time, the percentage of messages perceived as negative tends to increase during market crashes.

Finally, the total number of messages is indicative of the level of activity in these group chats. It is not necessarily reflected in crypto-activity, but it is worth noting the fluctuations in it as a rough indicator of community engagement.

Twitter sentiment – indicate a measurement of the emotions of market participants. Sometimes sentiment can be a leading indicator, as was the case with Ethereum in June and July. In most cases, however, sentiment tends to be a reactive indicator. In other words, there is more positive sentiment when prices are rising and negative sentiment when prices are falling.

3.3. ML Model Validation and Metrics

The correctness of the created regression model is a relative value. Despite on how the model validness is determined for classification problems, the regression validation does not include determining count of the "false negative" values. The regression model validity is defined by the following metrics.

1. Mean absolute error

$$MAE = \frac{1}{N} \cdot \sum_{t=1}^{N} \left| Y(t) - \hat{Y}(t) \right|, \quad (1)$$

where N – count of record in the test dataset, $\hat{Y}$ – predicted value, Y – real value.

2. Mean square error

$$MSE = \frac{1}{N} \cdot \sum_{t=1}^{N} \left| (Y(t) - \hat{Y}(t))^2 \right|, \quad (2)$$

where N – count of record in the test dataset, $\hat{Y}$ – predicted value, Y – real value.

3. Root mean square error

$$RMSE = \sqrt{MSE}. \quad (3)$$

4. Explained Variance

$$VAR = 1 - \frac{VAR(y - \widehat{y})}{VAR(y)}, \quad (4)$$

where $\hat{Y}$ – predicted value, Y – real value.

The R^2 metric decided to not include into model validation metric set. Despite the same R-squared statistic produced, the predictive validity would be rather different depending on what the true

dependency is. If it is truly linear, then the predictive accuracy would be quite good. Otherwise, it will be much poorer. In this sense, R-Squared is not a good measure of predictive error.

3.4. ML Models and Methods

There are bunches of different machine learning regression algorithms that can be selected. First of all, regression model selection should be based on requirements and data specifications based on which the model will be trained and validated.

This research focuses on time series data, because cryptocurrency price changes continuously, depending on the selected time frame. There are many types of regression models, but this article focuses on a nonlinear model.

There are a few reasons for these algorithms types selection:

- Logistic regression is not suitable because that algorithm has only two values (1, 0) for dependent variable. Investigated problem has time series data;
- The relationships between data variables that was explained above is not linear, because increasing one variable in couple of decreasing another one variable may affect the dependent variable in an unpredictable way. That means that usage of linear regression algorithms can lead to incorrect data fitting;
- Polynomial regression models a non-linear dataset using a linear model. It works in a similar way to multiple linear regression (which is just linear regression but with multiple independent variables) but uses a non-linear curve. It is used when data points are present in a non-linear fashion [30]. This algorithm is not fit for the current purpose, because there is a need to make rules or decisions instead of calculating an average between all points to "draw line".

The best way here is usage of non-linear regression algorithms. There are three representers of non-linear algorithms will be used:

- Decision trees;
- Random forest;
- Gradient boosted trees.

The decision tree regression has as a main function is to split the dataset into smaller sets. The subsets of the dataset are created to plot the value of any data point that connects to the problem statement. The splitting of the data set by this algorithm results in a decision tree that has decision and leaf nodes. ML experts prefer this model in cases where there is not enough change in the data set [31].

The decision tree algorithm has hyperparameters for model tuning. One of them is tree depth. If the maximum depth of the tree is set too high, the decision trees learn too fine details of the training data and learn from the noise, i.e. they overfit (Figure 4).

As a result, to make a good fitted model, the optimal count of tree depth is required. The optimal count of depth can be found experimentally. If the regression model validity metrics have a good performance for the training dataset, but on the test dataset is low, then the model overfitting happens. There is a need to decrease the tree's depth.

The random forest is also a widely-used algorithm for non-linear regression in Machine Learning. Unlike decision tree regression (single tree), a random forest uses multiple decision trees for predicting the output. Random data points are selected from the given dataset (say k data points are selected), and a decision tree is built with them via this algorithm. Several decision trees are then modeled that predict the value of any new data point. There is an example on Figure 5 of how a random forest algorithm works.

Since there are multiple decision trees, multiple output values will be predicted via a random forest algorithm. You have to find the average of all the predicted values for a new data point to compute the final output. The only drawback of using a random forest algorithm is that it requires more input in terms of training. This happens due to the large number of decision trees mapped under this algorithm, as it requires more computational power [32].

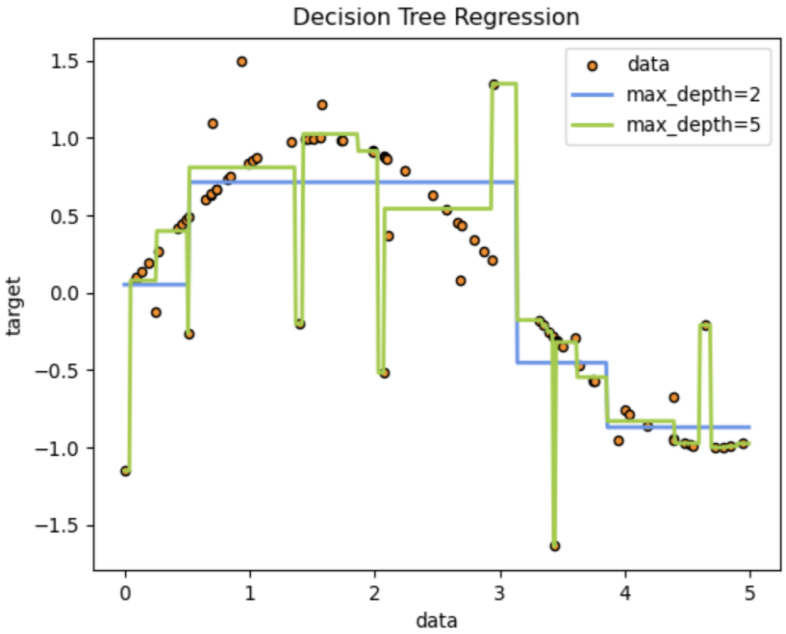

Figure 4: Representation of how the overfitting looks like [32]

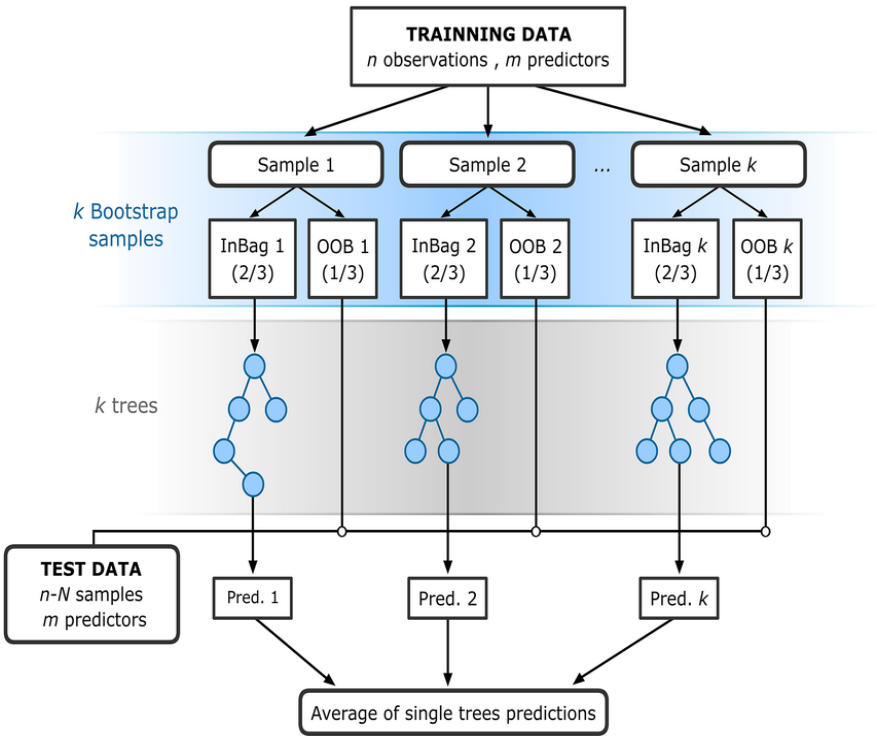

Figure 5: Explanation of random forest algorithm principle [33]

The Gradient Boosted Regression Trees (GBRT) model (also called Gradient Boosted Machine or GBM) is one of the most effective machine learning models for predictive analytics, making it an industrial workhorse for machine learning. The Boosted Trees Model is a type of additive model that makes predictions by combining decisions from a sequence of base models. For boosted trees model, each base classifier is a simple decision tree. This broad technique of using multiple models to obtain better predictive performance is called model ensembling. Unlike Random Forest which constructs all the base classifier independently, each using a subsample of data, GBRT uses a particular model assembling technique called gradient boosting [34].

4. Experiment

The main goal of this experiment is training of selected regression models, and determining which of the models is the best for small amounts of data. The experiment consists of two steps:

- Experiment planning;
- Results overview.

4.1. Experiment Planning

First of all, experiment requires creation of the dataset. This point is achieved by manually downloading the sources and making hierarchical folder structure for convenient files accessing. All of the source files have a column which describe a day when event is happened, and this column is used to merge each of the source files into full dataset.
The dataset is being created or combined from different sources using Apache Spark Framework abilities.

Apache Spark is a unified analytics engine for large-scale data processing. It provides high-level APIs in Java, Scala, Python and R, and an optimized engine that supports general execution graphs. It also supports a rich set of higher-level tools including Spark SQL for SQL and structured data processing, MLlib for machine learning, GraphX for graph processing, and Structured Streaming for incremental computation and stream processing.

The next step is splitting the dataset into two parts in the ratio of 70% and 30%. This action is required to train and test our model.

The spark.mllib supports decision trees for binary and multiclass classification, as well as for regression, using both continuous and categorical features. The implementation splits the data by rows, which allows distributed training with millions of instances.

Random forests are ensembles of decision trees. Random forests are one of the most successful machine learning models for classification and regression. They combine multiple decision trees to reduce the risk of overshoot. Like decision trees, random forests work with categorical features, extend to multi-class classification, do not require feature scaling, and can account for non-linearity and feature interaction. The spark.mllib supports random forests for binary and multi-class classification as well as regression, using both continuous and categorical features. The spark.mllib implements random forests using an existing decision tree implementation.

Gradient-Boosted Trees (GBTs) are ensembles of decision trees. GBTs iteratively train decision trees to minimize the loss function. Like decision trees, GBTs work with categorical features, extend to multi-class classification, do not require feature scaling, and are able to account for non-linearities and feature interactions.

The spark.mllib supports GBT for binary classification and regression using both continuous and categorical features. The spark.mllib implements GBT using an existing decision tree implementation. For more information about decision trees, see the decision tree guide.

Note that GBTs do not yet support multi-class classification. Use decision trees or Random Forests to solve multi-class problems.

After the dataset is created, ML models will be trained and validated using metrics that were mentioned above. An important note is a calculation of training time for each of the models during the training process to determine the fastest model for the proposed dataset.

The next step of the experiment is the validation of trained models, providing info about that and determining the correctness of assumption that selected factors have relationships with ETH price.

4.2. ML Models Training

The data that is selected to train the regression models is in a one-year time frame, because that time period has robust rules of market behavior for particular cryptocurrencies. The models training process that will be recorded below will be prepared with the best suited hyperparameters for this data. The

Decision Tree max depth: 30, The Random Forest max depth: 5, Gradient Boosted Tree: max iterations: 10, loss type: absolute, max depth: 5, subsampling rate: 0.4.

In general, data is grained by day, so there are 365 examples of how the indicators affect the Ethereum price. This is a small amount of data to make robust forecasting, but the experiment will give more valuable facts of that assumption. The Java 8, Spark framework and MacOS Monterey were selected to build and validate the regression models. Also, hardware consists of 2,6 GHz Quad-Core Intel Core i7, 16 GB 2133 MHz LPDDR3.

The microbenchmark for measuring how long the models are being trained was prepared by Java JMH Benchmark. JMH is a Java library for writing benchmarks on the JVM, developed as part of the OpenJDK project. JMH provides a very solid foundation for writing and executing benchmarks whose results will not be corrupted by unwanted virtual machine optimizations. There are the following benchmark modes, check Table 3.

Table 3
JMH Benchmark Modes

Name	Description
Throughput	Measures the number of operations per second, meaning the number of times per second your benchmark method could be executed.
Average Time	Measures the average time it takes for the benchmark method to execute (a single execution).
Sample Time	Measures how long time it takes for the benchmark method to execute, including max, min time etc.
Single Shot Time	Measures how long time a single benchmark method execution takes to run. This is good to test how it performs under a cold start (no JVM warm up).
All	Measures all of the above.

There are the following measurements of how long the models are being trained on given data, check Table 4.

The Average time mode was selected for benchmarking. This investigation uses milliseconds as a time unit. The TimeUnit class contains the following time unit constants:

- Nanoseconds;
- Microseconds;
- Milliseconds;
- Seconds;
- Minutes;
- Hours;
- Days.

Table 4
Average algorithms' training time

Time, Millis	Algorithm
5570	Decision tree algorithm
1400	Random forest algorithm
5960	Gradient boosted decision trees

The experiment shows that the quickest algorithm for that amount of data is a Random Forest regressor.

5. Results

This investigation part contains aggregated obtained results during testing of created regression models and provides it in a table view.

5.1. Decision Tree Model

The next table (Table 5) represents data that is retrieved during Decision Tree model validation.

Table 5
Decision tree validation results

Metric	Value
Root Mean Squared Error (RMSE)	285.94868249
Mean absolute error (MAE)	233.59134551
Mean square error (MSE)	81766.649019
Explained Variance	77061.375871

The Table 6 gives an example of output that Decision tree regression model produces. If the time consumption for model training is not a problem, then this algorithm can be used.

Table 6
Decision tree prediction results

Prediction	Price
4537.324	4346.08
4216.365234	4342.58
4730.384277	4283.6
4340.763672	4059.81
3970.181885	3848.18
3970.181885	3883.93
4030.908936	3960.15
4294.453612	3869.35
4269.73291	4076.1
4088.45776	4082.56
4409.93115	4057.3
4486.243164	4079.46

5.2. Random Forest Regression Model

The next table (Table 7) represents data that is retrieved during Random forest regression model validation.

Table 7
Random forest validation results

Metric	Value
Root Mean Squared Error (RMSE)	238.085738
Mean absolute error (MAE)	158.3889740
Mean square error (MSE)	56684.8187
Explained Variance	31666.02249

The Table 8 gives an example of output that Random forest regression model produces. The error values are larger than for the Decision Tree model, but the time consumption for training is less.

Table 8

Random forest prediction results

Prediction	Price
4274.41063283	4346.08
4176.54161135	4342.58
4242.195490373	4283.6
4186.625616152	4059.81
4049.135305026	3848.18
3940.053935855	3883.93
3898.710454863	3960.15
3635.556222742	3869.35
4306.760494009	4283.6
4207.974993752	4082.56
4142.932871340	4057.3
4154.965331289	4079.46

5.3. Gradient Boosted Trees

The next table (Table 9) represents data that is retrieved during Gradient Boosting Trees regression model validation.

Table 9

Gradient boosted trees validation results

Metric	Value
Root Mean Squared Error (RMSE)	261.4591287
Mean absolute error (MAE)	223.36118231
Mean square error (MSE)	68360.875985
Explained Variance	41659.384746

As we can show, the Table 10 gives an example of output that Gradient Boosted Trees regression model produces.

This algorithm is the most accurate from each other, but training time is an issue here.

Table 10

Gradient boosted trees prediction results

Prediction	Price
4374.544273549	4346.08
4374.58427354	4342.58
4374.74427354	4283.6
4281.1311509146	4059.81
4037.4451253475	3848.18
4037.609813992	3883.93
4037.622749360	3960.15
4038.030281828	3869.35
4280.553806327	4076.1

4280.3500029089	4082.56
4280.70529148001	4057.3
4651.65643435048	4079.46

6. Discussions

There are the following aggregated results in charts that display which of the algorithms is the best suited for a particular problem.

All charts represent a comparison of algorithms by particular regression accuracy metric. These charts include the following regression validation metrics:

- Root Mean Squared Error (RMSE);
- Mean absolute error (MAE);
- Mean square error (MSE);
- Explained Variance.

In the end of this section, the general conclusions regarding usage results of mentioned regression algorithms are extracted.

The Figure 6 shows algorithms comparison by RMSE metric, which means differences between values (sample or population values) predicted by a model or an estimator and the values observed. The RMSD represents the square root of the second sample moment of the differences between predicted values and observed values or the quadratic mean of these differences.

The smallest error is observed for Random Forest algorithm, along with the smallest training time.

The next chart (Figure 7) displays the comparison between algorithms by MAE metric which refers to the magnitude of difference between the prediction of an observation and the true value of that observation. MAE takes the average of absolute errors for a group of predictions and observations as a measurement of the magnitude of errors for the entire group. MAE can also be referred as L1 loss function. The results are the same as previous: Random Forest algorithm has highest accuracy, Gradient Boosted Trees is on the second place and Decision Tree is the last.

The next chart (Figure 8) displays the comparison between algorithms by MSE metric which defines as Mean or Average of the square of the difference between actual and estimated values.

The Random forest algorithm has the smallest value by that metric. The next chart (Figure 9) displays results for Explained variance (also called explained variation) is used to measure the discrepancy between a model and actual data. In other words, it's the part of the model's total variance that is explained by factors that are actually present and aren't due to error variance.

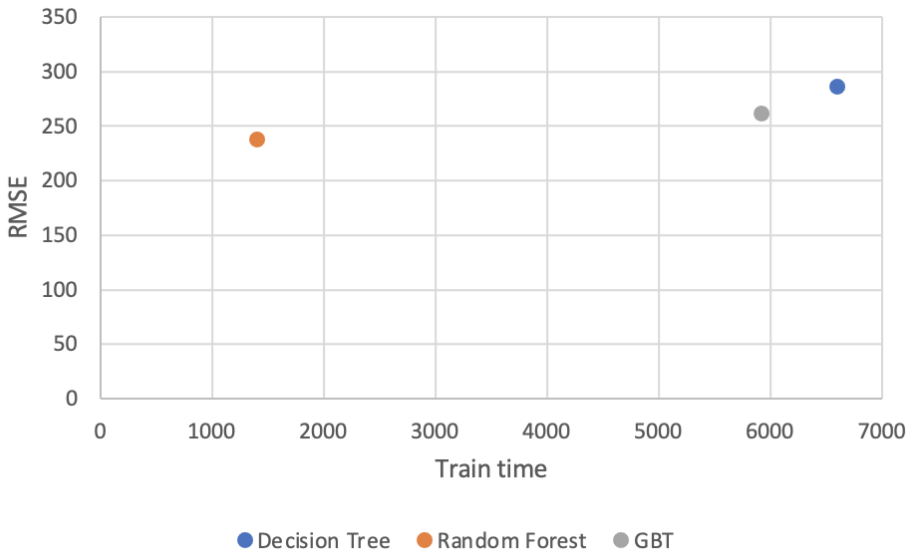

Figure 6: Result comparison between algorithms by RMSE metric

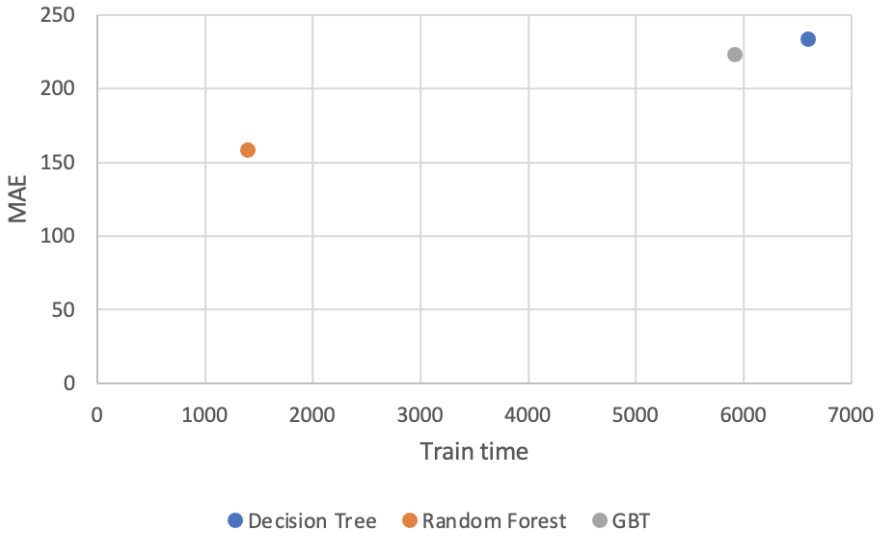

Figure 7: Result comparison between algorithms by MAE metric

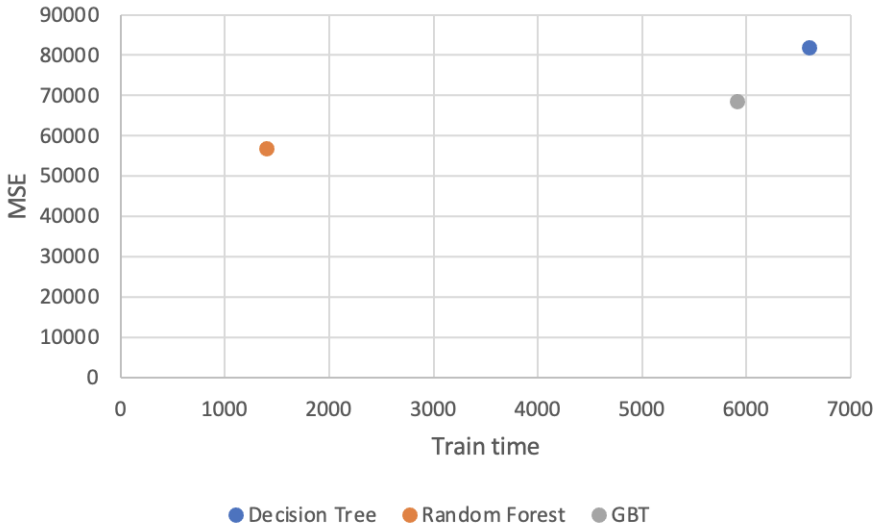

Figure 8: Result comparison between algorithms by MSE metric

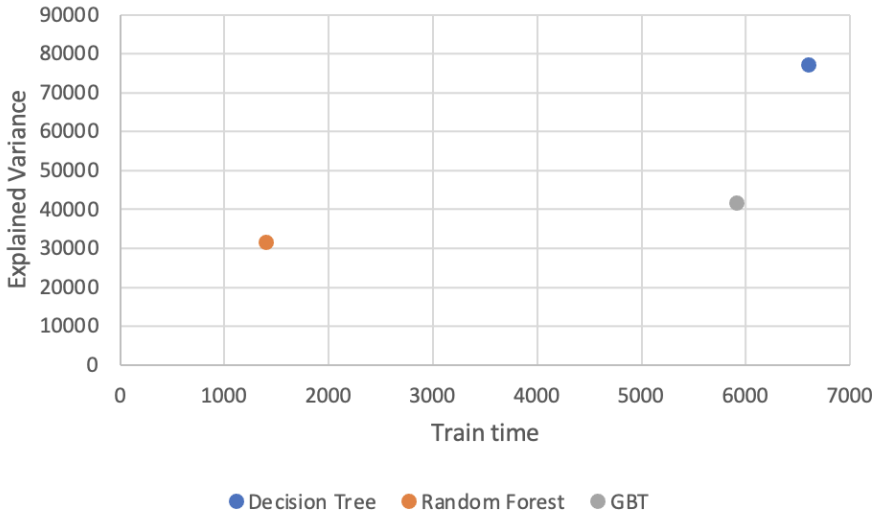

Figure 9: Result comparison between algorithms by “Explained Variance” metric

The higher value of explained variance indicates a stronger strength of association. It also means that you make better predictions.

As a result of practical experiment there are the following facts were found:

- The decision tree algorithm is the worth regression algorithm here. The biggest time-consuming gives biggest predicting results.
- The Random forest algorithm is the most accurate, and time to training is small.
- The Gradient Boosted Tree algorithm does not give expected results.

There are the following recommendations for improvement of decision tree and gradient boosting regression algorithm to make model more accurate and performance balanced:

- The decision tree model may be overfitted, which is often detrimental to the model's performance when you introduce new data. If there is no limit set of a decision tree, it will give you a zero MSE value on training set because in the worst case it will end up making 1 leaf for each observation. Thus, preventing overfitting is of major importance when training a decision tree and it can be done in 2 ways: Setting constraints on tree size (fine-tune hyperparameters) and Tree pruning.
- There are two types of parameter in Gradient boosting algorithm to be tuned– tree based and boosting parameters. There are no optimum values for learning rate as low values always work better, given that we train on sufficient number of trees. Though, GBM is robust enough to not overfit with increasing trees, but a high number for a particular learning rate can lead to overfitting. But as we reduce the learning rate and increase trees, the computation becomes expensive and would take a long time to run on standard personal computers. Keeping all this in mind, we can take the following steps to optimize model:

1. Choose a relatively high learning rate. Generally, the default value of 0.1 works but somewhere between 0.05 to 0.2 should work for different problems;
2. Determine the optimum number of trees for this learning rate. This should range around 40-70. Remember to choose a value on which your system can work fairly fast. This is because it will be used for testing various scenarios and determining the tree parameters;
3. Tune tree-specific parameters for decided learning rate and number of trees. Note that we can choose different parameters to define a tree and I'll take up an example here;
4. Lower the learning rate and increase the estimators proportionally to get more robust models.

7. Conclusions

The experiment for determining the factors which affect crypto price was setup in this work. Related works give understanding what was already done in scope of this theme. The statement regarding not exhaustive of reviewed works was extracted. The own opinion for investigation was proposed. This assumption is based on relationships between the following metrics: crypto data and worldwide metrics. The assumption of relationships existence here is based on dependencies in the world. It means dependencies between software and how this software is used in real life. Which affects software usage.

The next step was determining the best suited family of regression algorithms, and the non-linear family was selected. Also, the metrics which are required for validation were selected too.

The experiment execution was the next step. And the Apache Spark was used to create data set from source files and for creating and training the regression models. The JMH tool determined that the random forest algorithm is fastest between others in time elapsed for training perspective.

The result of validation of created models gives information that Random Forest algorithm is the most accurate between each other, also as a training time is smallest in comparison to other algorithms. The Gradient Boosted Tree algorithm stays in the middle of performance and Decision Tree algorithm does not suite for prepared data and problem.

Further investigations may be focused on including into model additional cryptocurrency factors: spreading crypto between exchanges, more financial factors for the worldwide economic and political situation like country financial institute openness which describes its ready to economic development and infrastructure failures.

8. References

[1] U. Mukhopadhyay, A. Skjellum, O. Hambolu, J. Oakley, L. Yu and R. Brooks, "A brief survey of Cryptocurrency systems," 2016 14th Annual Conference on Privacy, Security and Trust (PST), 2016, pp. 745-752, doi: 10.1109/PST.2016.7906988.

[2] F. Sabry, W. Labda, A. Erbad and Q. Malluhi, "Cryptocurrencies and Artificial Intelligence: Challenges and Opportunities," in IEEE Access, vol. 8, pp. 175840-175858, 2020, doi: 10.1109/ACCESS.2020.3025211.

[3] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll and E. W. Felten, "SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies," 2015 IEEE Symposium on Security and Privacy, 2015, pp. 104-121, doi: 10.1109/SP.2015.14.

[4] F. Béres, I. A. Seres, A. A. Benczúr and M. Quintyne-Collins, "Blockchain is Watching You: Profiling and Deanonymizing Ethereum Users," 2021 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS), 2021, pp. 69-78, doi: 10.1109/DAPPS52256.2021.00013.

[5] Yu Chen, Xuecheng Ma, Cong Tang and Man Ho Au, "Pgc: Pretty good decentralized confidential payment system with auditability", Cryptology ePrint Archive Report 2019/319, 2019, [online] Available. URL: https://eprint.iacr.org/2019/319.

[6] Understanding The Different Types of Cryptocurrency. URL: https://www.sofi.com/learn/content/understanding-the-different-types-of-cryptocurrency.

[7] The 10 Most Popular Cryptocurrencies, and What You Should Know About Each Before You Invest. URL: https://time.com/nextadvisor/investing/cryptocurrency/types-of-cryptocurrency.

[8] P. Tasatanattakool and C. Techapanupreeda, "Blockchain: Challenges and applications," 2018 International Conference on Information Networking (ICOIN), 2018, pp. 473-475, doi: 10.1109/ICOIN.2018.8343163.

[9] A Guide to Cryptocurrency Fundamental Analysis. URL: https://academy.binance.com/en/articles/a-guide-to-cryptocurrency-fundamental-analysis

[10] The 7 Key Factors Influencing Cryptocurrency Value. URL: https://www.makeuseof.com/factors-influencing-the-cryptocurrency-value/

[11] S. Boshuis, T. Braam, A. Pedroza Marchena and S. Jansen, "The Effect of Generic Strategies on Software Ecosystem Health: The Case of Cryptocurrency Ecosystems," 2018 IEEE/ACM 1st International Workshop on Software Health (SoHeal), 2018, pp. 10-17.

[12] Jiangtao Ma, Yaqiong Qiao, Guangwu Hu, Yongzhong Huang, Arun Kumar Sangaiah, Chaoqin Zhang, et al., "De-anonymizing social networks with random forest classifier", IEEE Access, vol. 6, pp. 10139-10150, 2017.

[13] Vynokurova O., Peleshko D., Zhernova P., Perova I., Kovalenko A. (2021) Solving Fraud Detection Tasks Based on Wavelet-Neuro Autoencoder. In: Babichev S., Lytvynenko V., Wójcik W., Vyshemyrskaya S. (eds) Lecture Notes in Computational Intelligence and Decision Making. ISDMCI 2020. Advances in Intelligent Systems and Computing, vol 1246. Springer, Cham. https://doi.org/10.1007/978-3-030-54215-3_34

[14] T. Radivilova, L. Kirichenko, D. Ageiev and V. Bulakh, "Classification Methods of Machine Learning to Detect DDoS Attacks," 2019 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2019, pp. 207-210, doi: 10.1109/IDAACS.2019.8924406.

[15] F. A. Cahyadi, A. I. Owen, F. Ricardo and A. A. S. Gunawan, "Blockchain Technology behind Cryptocurrency and Bitcoin for Commercial Transactions," 2021 1st International Conference on Computer Science and Artificial Intelligence (ICCSAI), 2021, pp. 115-119, doi: 10.1109/ICCSAI53272.2021.9609790.

[16] How Bitcoin Works. URL: https://www.investopedia.com/news/how-bitcoin-works.

[17] S. Pillai, D. Biyani, R. Motghare and D. Karia, "Price Prediction and Notification System for cryptocurrency Share Market Trading," 2021 International Conference on Communication information and Computing Technology (ICCICT), 2021, pp. 1-7, doi: 10.1109/ICCICT50803.2021.9510122.

[18] X. Li and C. A. Wang, "The technology and economic determinants of cryptocurrency exchange rates: The case of bitcoin", Decision Support Systems, vol. 95, pp. 49-60, 2017.
[19] A. Park, J. Kietzmann, L. Pitt and A. Dabirian, "The Evolution of Nonfungible Tokens: Complexity and Novelty of NFT Use-Cases," in IT Professional, vol. 24, no. 1, pp. 9-14, 1 Jan.-Feb. 2022, doi: 10.1109/MITP.2021.3136055.
[20] K. Smelyakov, A. Chupryna, M. Hvozdiev and D. Sandrkin, "Gradational Correction Models Efficiency Analysis of Low-Light Digital Image," 2019 Open Conference of Electrical, Electronic and Information Sciences (eStream), 2019, pp. 1-6, doi: 10.1109/eStream.2019.8732174.
[21] K. Smelyakov, M. Shupyliuk, V. Martovytskyi, D. Tovchyrechko and O. Ponomarenko, "Efficiency of image convolution," 2019 IEEE 8th International Conference on Advanced Optoelectronics and Lasers (CAOL), 2019, pp. 578-583, doi: 10.1109/CAOL46282.2019.9019450.
[22] O. Lemeshko, M. Yevdokymenko, O. Yeremenko, A. M. Hailan, P. Segeč and J. Papán, "Design of the Fast ReRoute QoS Protection Scheme for Bandwidth and Probability of Packet Loss in Software-Defined WAN," 2019 IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM), 2019, pp. 1-5, doi: 10.1109/CADSM.2019.8779321.
[23] Ageyev D., Radivilova T. Traffic monitoring and abnormality detection methods for decentralized distributed networks // CEUR Workshop Proceedings. 2021. Vol. 2923. P. 283–288.
[24] K. Smelyakov, A. Datsenko, V. Skrypka and A. Akhundov, "The Efficiency of Images Reduction Algorithms with Small-Sized and Linear Details," 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), 2019, pp. 745-750, doi: 10.1109/PICST47496.2019.9061250.
[25] A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms. URL: https://www.mdpi.com/2673-2688/2/4/30/pdf.
[26] Predictions of bitcoin prices through machine learning based frameworks. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8022579.
[27] Y. Xin et al., "Machine Learning and Deep Learning Methods for Cybersecurity," in IEEE Access, vol. 6, pp. 35365-35381, 2018, doi: 10.1109/ACCESS.2018.2836950.
[28] Data Repository. URL: https://drive.google.com/drive/folders/17wcLX2VVw1cCo_6RsCEas2HSiDnZhfj4?usp=sharing.
[29] Data for Analysis. URL: https://www.marketwatch.com/investing/index/spg1200/download-data?countrycode=xx.
[30] Five Types of Regression Analysis And When To Use Them. URL: https://www.appier.com/blog/5-types-of-regression-analysis-and-when-to-use-them.
[31] Eight popular regression algorithms in machine learning of 2021. URL: https://www.jigsawacademy.com/popular-regression-algorithms-ml.
[32] DTR. URL: https://scikit-learn.org/stable/auto_examples/tree/plot_tree_regression.html.
[33] The flowchart of random forest (RF) for regression. URL: https://www.researchgate.net/figure/The-flowchart-of-random-forest-RF-for-regression-adapted-from-Rodriguez-Galiano-et_fig3_303835073.
[34] The Gradient Boosted Regression Trees (GBRT) model. URL: https://apple.github.io/turicreate/docs/userguide/supervised-learning/boosted_trees_regression.html.