Abstract. This paper is devoted to the development of a reliable abnormal bitcoin transaction detection that may be involved in money laundering and illegal traffic of goods and services. The article proposed an algorithm of abnormal bitcoin transaction detection based on machine learning. For training and evaluation of the model, the Elliptic dataset is used, consisting of 46564 Bitcoin transactions: 4545 of “illegal” and 42019 of “legal”. The proposed algorithm for detecting abnormal bitcoin transactions is based on various machine learning algorithms with the selection of hyperparameters. To evaluate the proposed algorithm, we used the metric of accuracy, precision, recall, F1 score and index of balanced accuracy. Using the resampling algorithm in conditions of class imbalance, it was possible to increase the reliability of the classification of abnormal bitcoin transactions in comparison to the best known result on the Elliptic dataset.

Keywords: Bitcoin transactions · Classification · Detection of abnormal transactions · Machine learning

1 Introduction

The launch of first bitcoin in 2008 drew technological and business interest to payments through cryptographic methods (digital signature and hash function) and distributed transaction retention (blockchain). The main advantage of BTC is to provide anonymous and cheap money transactions both in the native country and abroad. However, criminals started actively using anonymity of BTC for illegal trade.

Anti-money laundering, AML plays the key role not only in providing safety for financial systems, but also determining illegal trade. BTC launch aroused a paradox: anonymity makes it possible for criminals to hide, though open BTC transaction database allows to perform forensic analysis or AML analytics. The objective of AML analytics is determining abnormal transactions which might
take place via reliable classification of minor illegal transactions in massive constantly increasing datasets. Manual or semi-automatic transaction analysis produces a high error rate. Meanwhile, success in machine learning shows great perspectives to use it for AML analysis [1].

The current paper is devoted to the development of a reliable abnormal bitcoin transaction detection that may be involved in money laundering and illegal traffic of goods and services. The Elliptic dataset is used [2], consisting of over 200 kt bitcoin transactions (nodes), 234 kt directed payments (edges) and 166 nodal functions, incl. the ones based on secretive data.

Methods of machine learning for binary classification which forecasts illegal transactions via logistic regression, random forest, multilayer perceptrons and convoluted network charts are used in the article [1] to detect abnormal bitcoin transactions. Convoluted network charts have appeared as a potential tool for AML analysis and it is especially attractive as a new method to gather and analyze transactions [1]. The results of work [1] reflect the advantage of random forest, though F1 score equal to 0.796 for the classifier set up on random forest is not sufficient to speak about reliable classification for bitcoin abnormal transactions.

To detect abnormal bitcoin transactions the following algorithms of machine learning were used in this article: linear regression, quadratic regression, logistic regression, k-nearest neighborhood, decision trees, random forest, naive Bayesian classifier, Support Vector Machines method, classifier based on multilayer perceptrons, linear discriminant analysis, quadratic discriminant analysis, adaptive boosting. Besides, methods of hyperparameter optimization for machine learning algorithms are used in this work which allowed to increase classification reliability.

It was important to observe that the Elliptic dataset is imbalanced (4545 “illegal” and 42019 “legal”). Therefore experiments on classification reliability for abnormal bitcoin transactions were performed with resampling algorithms under conditions of imbalanced classes. Due to it, it was possible to increase classification reliability for bitcoin abnormal transactions as compared to the best result in the Elliptic dataset [1].

2 Review of Approaches to Bitcoin Transaction Analysis

A set of tools for forensic analysis in bitcoin transactions is described in the article [3]. The set of tools comprises modular scalable framework including: Bitcore Node—“completely nodular” Bitcoin client which keeps information about transactions; Bitcoin Addresses Scraper, a tool for searching Bitcoin addresses on web pages to deanonymize their owners; Mixing Services Detector, a detector of mixing transactions services; BlockChainVis, a tool for transaction data visualization to simplify criminalists job, and also Bitcoin Addresses Clusteriser which is necessary to search group of clusters belonging to the same user. Bitcoin Addresses Clusteriser uses in its work a number of heuristics and their combinations. The dataset under research is not submitted.
The authors [4] describe algorithm of treatment major non-structured data within the task of detecting an illegal cycle of money. However, the dataset under research is not submitted.

The method of detecting transactions used in the services of mixing transactions blockchain (Blockchain Mixer) of a special type is described in the work [5]. It is performed via building assigned acyclic transaction graph, where nodes are transactions, and edges are bitcoin streams between transactions. To address this problem, a mathematical model has been developed, which showed in the course of research that it allows detecting the fact of participation of a user in arranging a transaction which gives extra advantages in detecting transactions connected with money laundering schemes. The dataset under research is not submitted.

The authors [6] proposed a mathematical model to analyze bitcoin transactions based on a hidden Markov model, a method of determining certainty values of transitions between bitcoin addresses. Markov model is a stochastic random process in the form of a direct graph where certainty values of state transitions depend only on the previous condition. The hidden Markov model belongs to the model where conditions are either hidden or not obvious. Regarding bitcoin transactions the observations give time-series of transactions between nodes. The research did not use real data, there was generated an imitative blockchain where the required transactions always enter the block regardless the amount. The dataset under research is not submitted.

The authors [7] analyze methods of money laundering detection based on Big Data treatment technology, instantiated, for example, in complex systems of fulfilling tasks at counteracting against money legalization and laundering which was obtained illegally (SAS Anti-Money Laundering System, SAS AML). The data are submitted as a graph. The necessary program modules are instantiated. The dataset under research is not submitted.

The results of using machine learning to solve a problem of deanonymizing bitcoin users are described in the article [8]. Data by Chainalysis company are used in the research. The dataset went through preliminary treatment and included 56 categories concerning illegal activities, and 957 various categories in total. The task comes down to searching the most effective algorithm of classification for data treatment. As a result, the method of gradient boosting is considered the best. The dataset under research is not submitted.

3 The Elliptic Dataset

There are several possible ways to represent bitcoin blockchain as a graph. The simplest of them is a graph where nodes are transactions, and edges are a stream of bitcoins between one and another transaction. In this representation bitcoin blockchain is an assigned acyclic graph. Excluding bordering cases when several outputs of transaction are spent as inputs of the same transaction, the degree of node input makes the number of transaction outputs, the spent number of outputs. The only nodes without inbound edges are coinbase transactions, i.e. newly made bitcoins included into the blockchain for the first time.
Each transaction is linked with a time feature, i.e. approximate time when transaction is streamed into bitcoin net. It allows to include temporal information into graph visualization. Payoff for obtaining a new block is currently BTC 12.5 (as of April 2020). Nowadays bitcoin transaction graph consists of over 438 million nodes and 1.1 billion edges. It is a constantly increasing graph as almost every day over 350,000 new confirmed BTC transactions take place.

The Elliptic dataset is a subgraph of total bitcoin transactions consisting of 203,769 nodes and 234,355 edges. Besides graph information, the Elliptic dataset includes information about node class: “legal”, “illegal” or “unknown”. A node is considered either legal or illegal if the current transaction was arranged by a legal party (broker’s board, wallets suppliers, miners, financial services providers, and etc.) or illegal (fraud, malicious software, terrorist groups, ransomware, pyramid investment schemes, drug dealers, and etc.) class correspondingly.

The task is to classify “legal” and “illegal” transactions taking into account a set of objects and a graph topology. As not all the nodes are marked, it is possible to approach the problem in a semi-control mode which includes data transmitted via non-marked nodes.

Just 2% (4545) of nodes belong to “illegal” class, 21% (42019) belong to “legal” class, 77% (157205) other transactions are “unknown”.

Timestamp in the graph is coded with 1 to 49 pitch which measures approximate time feature of the transaction. Timestamps are evenly distributed with about a 2-week interval; each of them contains one transaction organization unit which appears in the blockchain with the interval less than three hours. Edges connecting various time pitches also can be missing in the graph. Each associated graph component consists of 1,000–8,000 nodes. Most nodes are referred as “unknown”.

Each node is associated with 166 features. The first 94 features comprise local data on transaction including: time pitch, input and output qty, transaction fee, issue volume and aggregates, such as average bitcoin number received and spent in inputs and outputs, average number of incoming and outgoing transactions associated with inputs and outputs. Other 72 features named aggregate features are received with transaction information aggregation at one pitch backward/forward from the central node, giving max, min, standard deviation and correlation coefficients of neighboring transactions for the same data (number of inputs and outputs, transaction fee, and etc.).

The article [1] is presented by IBM research workers Weber and Domeniconi at the seminar on detection anomaly in financial flows named Knowledge Discovery and Data Mining Conference (KDD) on August 5, 2019, the Head of the Conference on digital intelligence. The article sets out results of early experiments with various algorithms at machine learning: from conventional classification models (logistic regression, random forest, multilayer perceptrons) to more complex, such as graph convoluted networks (GCN). The above-mentioned various models are thoroughly compared in the article, and the effect of graph data inclusion into conventional classification methods is studied and presented as a table. This work brings several conclusions:
1. Non-local information implication, in particular, information about central node’s neighbors always improves models’ productivity;
2. Conventional (non-graph) models of classification win from extra functions provided by GCN attachments;
3. Random forest is the best classification model for this task.

The fact that GCN model did not prove as the best one is a curious discovery. At the Elliptic dataset GCN operates much better than logistic regression, but worse than random forest. As it was proposed in the article [1], insignificant improvement of results can be reached at combining abilities learning graph algorithms (such as GCN) with decision trees or random forest. One more problem of the Elliptic dataset is temporal dynamics of appearance and disappearance of new matters in a blockchain. Dark Market closure can serve as an example, when after this event no model operates normally due to sudden changes in basic system behavior.

4 Bitcoin Abnormal Transaction Detection Based on Machine Learning

Bitcoin abnormal transaction detection can be built with supervised machine learning [9–11]. Supervised machine learning uses marked training data \((x, y)\) to determine objective function \(f\), where \(x\) is a vector of input features, and \(y\) is an output mark. Algorithm of learning generalizes correlation between the vector of features and the mark in training data for new examples to determine their marks correctly.

Assume that \(\{(x_1, y_1), \ldots, (x_n, y_n)\}\) is a marked training dataset, where \(x_i = (x_{i,1}, \ldots, x_{i,d}) \in X\) for \(1 \leq i \leq n\) which is a vector of features for \(x_i\) and \(y_i \in Y\) is its relevant mark. The goal in supervised learning is to detect an objective function for a set of marks using a training data set. Assume that \((x_i, y_i)\) are independent and equally distributed. The supposed objective function can be estimated with its forecast ability on test data. If either \(Y \in \mathbb{R}\) or \(Y \in \mathbb{R}^d\) marks are continuous actual levels or a vector above real levels, then the machine learning model is regressive. If marks come as discrete states or symbols, then the machine learning model is named classification.

The task of supervised machine learning is to search objective function \(f : X \rightarrow Y\), where \(X\) is an input dataset, and \(Y\) is a set of output variables. The process of searching this objective function \(f\) is called supervised learning or modeling. The objective function can be detected only if enough marked data are available. As it’s difficult to detect precise objective function in practically it is close to approximate function. Learning actually includes function \(h\) search which at its best approximates the unknown objective function \(f\). Model training is performed with a test set which features a representative dataset randomly taken from the general dataset. As a rule, 70% of data are used for training. Then model parameters are given with a validation set; usually 20% of training data are taken for the validation set. At last, actual forecast capacity of the model is checked with a test set, as a rule 30% of data are taken for a test set.
There is a dataset $M$ in bitcoin abnormal transaction detection which consists of $n = 46564$ transactions, and a mark is a binary variable (“illegal” or “legal”). We are not considering “unknown” class of transactions. We can apply a semi-supervised machine learning approach to bitcoin abnormal transaction detection using unknown transactions. We leave the execution of this idea as future investigation. Assume that mark 0 points at “legal” transaction, and mark 1 points at “illegal” transaction. A number of vector features consisting of 94 features of local information and 72 aggregate features are associated with each transaction $m$. The objective function $f : M \in L$ is to determine if a definite node $m$ is “illegal” 1 or “legal” 0. Function $f : M \in \{0, 1\}$ can be detected with one of machine learning algorithms for a set of $n$ marked nodes $\{(m_1; l_1), (m_2; l_2), \ldots, (m_n; l_n)\}$, where $m_i \in M$ and $l_i \in \{0, 1\}$ for $1 \leq i \leq n$.

The following machine learning algorithms are used in the article to detect bitcoin abnormal transactions: linear regression; quadratic regression; logistic regression; $k$-nearest neighbourhood, decision trees, random forest, naive Bayesian classifier, Support Vector Machines method, classifier based on multi-layer perceptrons, linear discriminant analysis, quadratic discriminant analysis, adaptive boosting, gradient boosting.

Consider general evaluation metrics used in the machine learning models in case the classification is binary as positive (“illegal”) and negative (“legal”) class.

One important observation took place concerning the data in the Elliptic dataset, which proves imbalanced (4545 “illegal” and 42019 “legal”). Assume this class minority, when example share of a certain class in the dataset is too small, and the other class is majority which is broadly represented in the dataset.

Among the approaches to solve the imbalance problem is to apply various resampling strategies. There exist two ways to restore class balance. In the first case some examples of major class are removed (undersampling), in the other the examples of minor class are increased with synthetic data (oversampling).

TomekLinks [12] algorithm was selected to solve the imbalance problem, where all major records incoming in TomekLinks must be removed from the dataset. TomekLinks can be determined as follows. Assume that $E_i = (x_i, y_i)$ and $E_j = (x_j, y_j)$ are two examples from different classes, where $y_i \neq y_j$, and $d(E_i, E_j)$ is the distance between $E_i$ and $E_j$, then a pair $(E_i, E_j)$ is defined as TomekLinks if there is no such example $E_l$, as either $d(E_i, E_l) < d(E_i, E_j)$ or $d(E_j, E_l) < d(E_i, E_j)$. TomekLinks algorithm succeeds in noisy record removing.

Geometric mean (Gmean) is a parameter $\sqrt{TP + TN}$, which is used to maximize correct and positive, and correct and negative classification results at keeping the balance between them. Where TP is a number of correct and positive results when the example of a positive class is correctly forecast with the model, and belonging to a positive class; and TN is a number of correct and negative results when the example of a negative class is correctly forecast with the model as belonging to a negative class. It’s worth mentioning that geometric mean leads to minimizing a negative impact of class distributions errors, though it is unable to explain the contribution of each class to overall index, giving the same result for various TP and TN combinations.
Dominance can be defined as $TP - TN$, which is used to estimate relationship between TP and TN.

The article [13] proposes Index of Balanced Accuracy (IBA) to estimate binary classifier in case with data imbalanced which can be obtained as

$$IBA = (1 + Dominance) \cdot Gmean^2.$$  

At replacement of Dominance and Gmean the final equation gives useful information for better understanding how IBA supports a compromise between Dominance and Gmean, besides a weighted parameter $0 \leq \alpha \leq 1$ can be added for Dominance

$$IBA = (1 + \alpha(TP - TN)) \cdot TP \cdot TN.$$  

However, if $\alpha = 0$, then IBA turns exactly into $Gmean^2$.

5 Experiments

To implement the algorithm of bitcoin transaction classification with machine learning Python 3 programming language was used as it is the best suitable one for fulfilling this task, and it has a perfect productivity at data treatment. Besides, Python contains many frameworks and libraries which simplify the process of writing a code and reduce the time for implementation.

Repository Pandas with high-level data structure was selected for data handling. It possesses in-built methods to group, combine and filter data. Pandas makes it possible to take data from different sources, such as SQL dataset, CSV, Excel, JSON files, and manipulate with the data to operate them.

Algorithms of machine learning from repository SciKit-Learn (SKlearn) [14] are used for abnormal transactions classification. The following algorithm types of machine learning were used to classify abnormal transactions: LogisticRegression, RandomForestClassifier, DecisionTreeClassifier, SVC, KNeighborsClassifier, MLPClassifier, AdaBoostClassifier.

Some part of machine learning algorithms gave poor results, therefore further on results of the following machine learning algorithms are not presented: linear regression, quadratic regression, linear discriminant analysis, quadratic discriminant analysis, naive Bayesian classifier.

We also used the following algorithms of machine learning for classification: CatBoostClassifier from the open repository implementing unique patented algorithm in constructing models of machine learning based on original schemes of gradient boosting, CatBoost [15]; XGBClassifier from repository of scaled gradient boosting XGBoost [16]; LGBMClassifier from the repository of gradient boosting LightGBM [17].

As test and training samples there were selected transactions marked as either “legal” or “illegal” in ratio 3:7 correspondingly. The bitcoin abnormal transactions detection was supervised with the help of selected algorithms, and to estimate the reliability of supervised models the metrics are calculated: accuracy, precision, recall, F1 Score and IBA. Table 1 provides the results of classification reliability based on a test set, where metrics precision, recall, F1 Score are
given only for “illegal” transactions, as to estimate effectiveness of transaction detection is in most cases important to analyze errors for “illegal” transactions.

From results represented in Table 1 it is obvious that the best classification algorithm is CatBoostClassifier with F1 score equal to 0.791 and accuracy 0.9780. However, the obtained results are slightly lower than the best result from article [1], where RandomForest algorithm has F1 score equal to 0.796 and accuracy 0.9780. Whereas, extra features obtained via GCN were used for a supervised model based on RandomForest in the article [1].

F1 score is equal to 0.796 and accuracy is 0.9780 which is not an acceptable result in effective detection of abnormal transactions. Our experiments with models involves testing different combinations of related hyperparameters to find the optimal response within a given set of values. To automate the process of obtaining the best combination of hyperparameters, the GridSearchCV algorithm from sklearn library is used. For the best CatBoostClassifier model, under different values of depth, iterations, l2_leaf_reg, learning_rate, border_count are analyzed. In the end, the hyperparameters of depth = 9, iterations = 1000, l2_leaf_reg = 4, learning_rate = 0.1, border_count = 10 yield the optimal response for the best CatBoostClassifier model.

Table 1. Results of algorithm classification reliability based on a test set with accuracy, precision, recall, F1 score and IBA estimation. Metrics of precision, recall, F1 score are presented only for “illegal” transaction class.

| Algorithm             | “Illegal” transactions | Accuracy | IBA  |
|-----------------------|------------------------|----------|------|
|                       | Precision | Recall | F1 Score |          |          |
| RandomForestClassifier| 0.985      | 0.653  | 0.785    | 0.9775   | 0.6688   |
| AdaBoostClassifier    | 0.873      | 0.594  | 0.707    | 0.9690   | 0.6111   |
| SVC                   | 0.862      | 0.588  | 0.699    | 0.9681   | 0.6052   |
| MLPClassifier         | 0.814      | 0.612  | 0.699    | 0.9668   | 0.6041   |
| KNeighborsClassifier  | 0.634      | 0.603  | 0.618    | 0.9531   | 0.6081   |
| DecisionTreeClassifier| 0.507     | 0.681  | 0.582    | 0.9383   | 0.6668   |
| LogisticRegression    | 0.454      | 0.633  | 0.529    | 0.9290   | 0.6168   |
| LGBMClassifier        | 0.928      | 0.589  | 0.721    | 0.9713   | 0.6056   |
| XGBClassifier         | 0.984      | 0.636  | 0.773    | 0.9764   | 0.6601   |
| CatBoostClassifier    | 0.985      | 0.661  | 0.791    | 0.9780   | 0.6801   |
| RandomForest [1]      | 0.971      | 0.675  | 0.796    | 0.9780   | –        |

To solve the imbalance problem the following resampling algorithms were used from the imbalanced-learn library [18]: TomekLinks, ClusterCentroids, RandomUnderSampler, SMOTE, RandomOverSampler, SMOTETomek. TomekLinks was selected as the best one.

Table 2 holds results of algorithm classification based on a test set, where precision, recall, F1 Score are given only for “illegal” transactions. Preliminary
TomekLinks algorithm was implemented to classify a dataset, and it removed 226 “legal” transactions. Then the dataset was randomly split into a test and a training part in ratio 3:7 correspondingly.

The results presented in Table 2 show that the best classification algorithm is XGBClassifier with F1 Score equal to 0.957, IBA 0.9599 and accuracy 0.9921. Obtained metrics significantly outperform random forest algorithm with F1 score equal to 0.796 and accuracy equal to 0.9780 given in the article [1].

F1 Score 0.957 and accuracy 0.9921 on a test set are acceptable for effective detection of abnormal transactions, so the conclusion comes that with Tomek-Links resampling algorithm in imbalanced-learn classes it was possible to increase reliability of bitcoin abnormal transaction classification.

### Table 2. Results of algorithm classification reliability based on a test set at estimating accuracy, precision, recall, F1 Score and IBA.

| Algorithm               | “Illegal” transactions |       |       |       |
|-------------------------|------------------------|-------|-------|-------|
|                         |                        | Precision | Recall | F1 Score | Accuracy | IBA     |
| RandomForestClassifier  | 0.997                  | 0.882  | 0.936 | 0.9885 | 0.9395   |
| AdaBoostClassifier      | 0.916                  | 0.858  | 0.886 | 0.9790 | 0.9225   |
| SVC                     | 0.908                  | 0.777  | 0.837 | 0.9712 | 0.8778   |
| MLPClassifier           | 0.901                  | 0.768  | 0.825 | 0.9646 | 0.8712   |
| KNeighborsClassifier    | 0.889                  | 0.840  | 0.864 | 0.9747 | 0.9115   |
| DecisionTreeClassifier  | 0.881                  | 0.899  | 0.890 | 0.9787 | 0.9419   |
| LogisticRegression      | 0.823                  | 0.767  | 0.794 | 0.9290 | 0.8679   |
| LGBMClassifier          | 0.994                  | 0.918  | 0.954 | 0.9916 | 0.9579   |
| XGBClassifier           | 0.995                  | 0.922  | 0.957 | 0.9921 | 0.9599   |
| CatBoostClassifier      | 0.992                  | 0.920  | 0.954 | 0.9916 | 0.9586   |
| RandomForest [1]        | 0.971                  | 0.675  | 0.796 | 0.9780 | –        |

Our experiments with models involves testing different combinations of related hyperparameters to find the optimal response within a given set of values. To automate the process of obtaining the best combination of hyperparameters, the GridSearchCV algorithm from sklearn library is used. For best XGBClassifier model, under different values of max_depth, min_child_weight, n_estimators, learning_rate are analyzed. In the end, the hyperparameters of max_depth=10, min_child_weight = 1, n_estimators = 200, learning_rate = 0.05 yield the optimal response for XGBClassifier model.

The accuracy and IBA scores in our experiments are also validated through 10-fold cross-validation. The results of 10-fold cross-validation for the best performing methods (RandomForestClassifier, LGBMClassifier, XGBClassifier, CatBoostClassifier) for various evaluation measures are shown in Table 3. For
all 10 iterations, the values of evaluation measures remain almost the same, indicating the stability of the XGBClassifier method for bitcoin abnormal transaction detection. Thus, we can say that the XGBClassifier method performed better than all other models used in this study for detecting abnormal bitcoin transactions.

### Table 3

The results of 10-fold cross-validation for the best performing methods (RandomForestClassifier, LGBMClassifier, XGBClassifier, CatBoostClassifier) for accuracy and IBA evaluation measures. Mean: mean value of various evaluation measures. SD (×10⁻⁴): standard deviation of various evaluation measures.

| Algorithm            | Accuracy | IBA   |
|----------------------|----------|-------|
|                      | Mean     | SD    | Mean | SD    |
| RandomForestClassifier | 0.9874   | 6.7864 | 0.8835 | 6.1886 |
| LGBMClassifier       | 0.9911   | 7.5099 | 0.9199 | 6.6320 |
| XGBClassifier        | 0.9917   | 7.1377 | 0.9203 | 6.6133 |
| CatBoostClassifier   | 0.9910   | 7.7603 | 0.9198 | 6.6755 |

### 6 Conclusion

The algorithm of bitcoin abnormal transaction detection based on machine learning was proposed in this paper. Bitcoin abnormal transactions are determined as transactions which can participate in money laundering and illegal traffic of goods and services. For training and evaluation of the proposed algorithm the Elliptic dataset is used comprising over 200,000 bitcoin transactions: 4545 are “illegal”, 42019 “legal” and 157205 “unknown”. The proposed model for bitcoin abnormal transaction detection used various algorithms of machine learning with the selection of hyperparameters, however the productivity of the proposed model equal to 0.9780 is not acceptable for effective detection of abnormal transactions. With TomekLinks resampling algorithm in imbalanced-learn conditions we managed to increase reliability of bitcoin abnormal transaction classification as compared to the best result equal to 0.9780 from the article [1]. Subsequently, the reliability of the proposed model for bitcoin abnormal transaction detection based on XGBClassifier algorithm equals to 0.9921.

### References

1. Weber, M., et al.: Anti-money laundering in bitcoin: experimenting with graph convolutional networks for financial forensics (2019)
2. Elliptic data set. [https://www.kaggle.com/ellipticco/elliptic-data-set](https://www.kaggle.com/ellipticco/elliptic-data-set)
3. Bistarelli, S., Mercanti, I., Santini, F.: A suite of tools for the forensic analysis of bitcoin transactions: preliminary report. In: Mencagli, G., et al. (eds.) Euro-Par 2018. LNCS, vol. 11339, pp. 329–341. Springer, Cham (2019). [https://doi.org/10.1007/978-3-030-10549-5_26](https://doi.org/10.1007/978-3-030-10549-5_26)
4. Kedharewsari, K., Maria Anu, M., Rajalakshmi, V.: Integration of big data & cloud computing to detect black money rotation with range - aggregate queries. Int. J. Eng. Technol. 8, 768–773 (2016)
5. Maksutov, A.A., Alexeev, M.S., Fedorova, N.O., Andreev, D.A.: Detection of blockchain transactions used in blockchain mixer of coin join type. In: 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), pp. 274–277 (2019)
6. Oakley, J., Worley, C., Yu, L., Brooks, R., Skjellum, A.: Unmasking criminal enterprises: an analysis of bitcoin transactions. In: 2018 13th International Conference on Malicious and Unwanted Software (MALWARE), pp. 161–166 (2018)
7. Plaksiy, K., Nikiforov, A., Miloslavskaya, N.: Applying big data technologies to detect cases of money laundering and counter financing of terrorism. In: 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), pp. 70–77 (2018)
8. Yin, H., Langenheldt, K., Harlev, M., Mukkamala, R.R., Vatrapu, R.: Regulating cryptocurrencies: a supervised machine learning approach to de-anonymizing the bitcoin blockchain. J. Manage. Inf. Syst. 36, 37–73 (2019)
9. Thomas, T., P. Vijayaraghavan, A., Emmanuel, S.: Machine Learning Approaches in Cyber Security Analytics. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-1706-8
10. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Berlin (2006)
11. MacKay, D.J.C.: Information Theory. Inference & Learning Algorithms. Cambridge University Press, USA (2003)
12. Tomek, I.: Two modifications of CNN. IEEE Trans. Syst. Man Cybern. 6, 769–772 (1976)
13. García, V., Mollineda, R.A., Sánchez, J.S.: Index of balanced accuracy: a performance measure for skewed class distributions. In: Araujo, H., Mendonça, A.M., Pinho, A.J., Torres, M.I. (eds.) IbPRIA 2009. LNCS, vol. 5524, pp. 441–448. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02172-5_57
14. Pedregosa, F., et al.: SciKit-learn: machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830 (2011)
15. Dorogush, A.V., Ershov, V., Gulin, A.: CatBoost: gradient boosting with categorical features support. CoRR abs/1810.11363 (2018)
16. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD 2016, New York, NY, USA, Association for Computing Machinery, pp. 785–794 (2016)
17. Ke, G., et al.: LightGBM: a highly efficient gradient boosting decision tree. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems 30, pp. 3146–3154. Curran Associates, Inc. (2017)
18. Lemaitre, G., Nogueira, F., Aridas, C.K.: Imbalanced-learn: a python toolbox to tackle the curse of imbalanced datasets in machine learning. J. Mach. Learn. Res. 18, 559–563 (2016)