УДК 004.852.34 Вестник УрФО № 2(48) / 2023, с. 30–41

Андреев А.М., Поршнев С.В.

DOI: 10.14529/secur230202

ОПЫТ РАЗРАБОТКИ ПРОГРАММНОЙ РЕАЛИЗАЦИИ КЛАССИФИКАТОРА ДЛЯ АВТОМАТИЗАЦИИ ОБНАРУЖЕНИЯ ФАКТОВ ИСПОЛЬЗОВАНИЯ КРИПТОКОШЕЛЬКОВ В ПРОТИВОПРАВНОЙ ДЕЯТЕЛЬНОСТИ

В статье обсуждается методика построения бинарного классификатора для выявления криптокошельков, связанных с программами-вымогателями. Был создан датасет из 41698 адресов, из них 20849 адресов, связанных с программами-вымогателями и 20849 адресов, несвязанных с программами вымогателями. Для каждого из кошельков было выделено 53 признака. Для построения классификаторов использовались следующие алгоритмы: логистическая регрессия, k-ближайших соседей, деревья принятия решений, случайный лес, градиентный бустинг. Был осуществлен подбор гиперпараметров классификаторов. Для оценки качества классификаторов использовались метрики: accuracy, precision, recall, F_1, ROC-AUC и кривые ROC. По метрикам accuracy (95.54%), precision (92.40%), F_1 (94.24%) лучший результат показал градиентный бустинг, по метрике recall (99.15%) – логистическая регрессия, по метрике ROC-AUC (98.85%) – случайный лес. По кривым ROC – случайный лес и градиентный бустинг.

Ключевые слова: *биткоин, криптовалюты, криптокошельки, криптоадреса, программы-вымогатели, машинное обучение, противоправная деятельность.*

EXPERIENCE OF DEVELOPING SOFTWARE IMPLEMENTATION OF A CLASSIFIER FOR AUTOMATION OF DETECTION OF THE FACTS OF USE OF CRYPTOWALLETS IN ILLEGAL ACTIVITIES

The article discusses a methodology for constructing a binary classifier to identify crypto wallets associated with ransomware. A dataset of 41,698 addresses was created, of which 20,849 were ransomware-related and 20,849 were non-ransomware. For each of the wallets, 53 features were identified. The following algorithms were used to build classifiers: logistic regression, k-nearest neighbors, decision trees, random forest, gradient boosting. The hyperparameters of the classifiers were selected. To assess the quality of the classifiers, the following metrics were used: accuracy, precision, recall, F_1, ROC-AUC and ROC curves. According to the accuracy (95.54%), precision (92.40%), F_1 (94.24%) metrics, gradient boosting showed the best result, according to the recall metric (99.15%) – logistic regression, according to the ROC-AUC metric (98.85%) – random forest. According to ROC curves – random forest and gradient boosting.

Keywords: *bitcoin, cryptocurrencies, crypto wallets, crypto addresses, ransomware, machine learning, illegal activity.*

Введение

Мониторинг и анализ финансовых потоков с целью противодействия отмывания денег (ОД) и финансирования терроризма (ФТ) в соответствие с Федеральным законом №115-ФЗ [1] возложено на Федеральную службу по финансовому мониторингу (Росфинмониторинг). Также данный закон предусматривает обязательное предоставление информации в Росфинмониторинг организаторами торговли, клиринговыми организациями, центральными контрагентами и профессиональными участниками рынка ценных бумаг, осуществляющими деятельность исключительно по инвестиционному консультированию, иными кредитными и финансовыми организациями в случае обнаружения признаков ОД/ФТ клиентами этих организаций.

Требования №115-ФЗ, также распространяются на криптовалюты, используемые сегодня в качестве платежного средства, способа заработка или инструмента сохранения капитала, популярность которых непрерывно возрастает, в том числе, и в РФ (особенно, исторически первой криптовалюты – биткоина). Например, в 2021 г., капитализация рынка цифровых валют впервые превысила $3 трлн [2]. В России оборот криптовалют легализован Федеральным законом «О цифровых финансовых активах, цифровой валюте и о внесении изменений в отдельные законодательные акты Российской Федерации» от 31.07.2020 №259-ФЗ.

Напомним, что криптовалюта – это цифровая платежная децентрализованная система с равноправными участниками, в которой банки не участвуют в проверке транзакций, что позволяет любому пользователю отправ-

лять и получать платежи в любом месте мира. Криптовалюта – это ключ (не материальный объект), который позволяет перемещать запись или единицу измерения от одного лица к другому без доверенной третьей стороны. Криптовалютные платежи существуют только в цифровом виде в онлайн-базе данных – распределенном публичном реестре, называемом блокчейном (см., например, [3]), в котором хранится информация о всех транзакциях, совершенных пользователями данной цифровой платежной системы.

Для проведения финансовых операций с криптовалютами используется следующая технология. Пользователь генерирует открытый ключ, на основе которого создается криптовалютный адрес (идентификатор, используемый для совершения транзакций в блокчейне криптовалюты). При отправке криптовалюты в блокчейн пользователь создает закрытый ключ, которым подписывает транзакцию. Таким образом, финансовые операции биткоинкошельков на рынке сегодня с криптовалютой представляют собой операции с криптовалютным адресом, а программно-аппаратное обеспечение, которое позволяет пользователю хранить, отправлять и получать криптовалюту называют криптовалютным кошельком.

Сегодня известны следующие способы перевода (обналичивания) криптовалюты в реальные деньги:

1. Сайты-обменники криптовалюты на реальные деньги (фиат) (например, [4]).

2. P2P-обменники (например, сайт биржи BINANCE [5]).

3. Платежные системы, предоставляющие сервис для автоматической конвертации криптовалюты в фиат по курсу, установленному выбранным сервисом (например, Payeer [6], Advcash [7]).

4. Криптокошельки, поддерживающие данную функцию (например, Blockchain [8]).

5. Криптоматы – аналоги банкоматов, имеющиеся в наличии, однако, не во всех странах.

6. Telegram-боты – сайты-обменники, функционирующие в сети Telegram.

Оборотной стороной криптовалют, обусловленной их анонимностью, является их активное использование для реализации противоправных действий, в том числе, для ОД/ФТ. Например, сегодня криптокошельки, в случае проведения успешной компьютерной атаки (КА) на информационную систему (ИС) с помощью вредоносной программы, блокирующей доступ к компьютерам и/или файлам, активно используются в качестве платежного инструмента для получения вымогаемых нарушителями средств у пользователей зараженной ИС за устранение последствий КА. В 2021 г. по данным Reuters, преступники отмыли с помощью криптовалюты около 8,6 миллиарда долларов США [9].

В этой связи задача мониторинга и анализа оборота криптовалют с целью выявления признаков использования криптокошельков для проведения незаконных действий и финансовых операций является актуальной. С математической точки зрения данная задача относится к задаче анализа данных, в которой требуется на основе анализа некоторого известного набора признаков, характеризующих криптовалютные платежи, совершенные с данного криптокошелька, отнести его либо к платежным инструментам, используемым только для проведения легальных финансовых операций, либо платежным инструментам, которые используются для проведения незаконных финансовых операций, точнее к задаче бинарной классификации. Результаты проведенного авторами анализа подходов, используемых для решения данного типа задач, показали, что одним из наиболее популярных сегодня является метод машинного обучения, в котором в качестве классификатора используются нейронные сети, обученные на соответствующих наборах данных (датасетах). Однако готовых программных решений, адаптированных для автоматизированного решения задачи классификации биткоинкошельков, обнаружить не удалось, что определяет необходимость разработки соответствующих инструментов, который окажется востребованным как сотрудниками Росфинмониторинга, так и других организаций, ответственных за мониторинг финансовых потоков и выявления признаков ОД/ФТ.

Цель статьи – описание методики построения бинарного классификатора криптокошельков, обеспечивающего их автоматическое отнесение к соответствующим классам их вовлеченности в противоправную деятельность.

Методика построения бинарного классификатора криптокошельков

Проведенный авторами анализ ресурсов, размещенных в сети Интернет, позволил обнаружить два размеченных набора данных, содержащих информацию с метками транзак-

ций в сети биткоин [10, 11]. Здесь разметка набора данных состояла в отнесении экспертами соответствующих транзакций либо к классу транзакций, связанных с программами-вымогателями, либо к противоположному классу. Однако оказалось, набор данных [11], содержит анонимизированные данные, так как для обучения классификатора необходимо выделить признаки для конкретного адреса, что в рассматриваемом случае оказывается невозможным. В этой связи в дальнейшем исследовании был использован набор данных [10].

Авторы использовали следующие популярные сегодня алгоритмы классификации:

– логистическая регрессия [12];

– k-ближайших соседей [13];

– деревья принятия решений [14];

– случайный лес [15];

– градиентный бустинг [16],

программные реализации которых включены в свободно распространяемую библиотеку scikit-learn [17], написанную на языке Python.

Для разработки программной реализации классификатора была использована методика, реализующаяся следующей последовательностью действий:

1. Скачивание набора данных с [10] в формате csv.

2. Подключение библиотек: pandas версии 1.4.4, numpy 1.21.5, requests 2.28.1, json 2.0.9, sklearn 1.0.2, matplotlib 3.5.2, pickle 4.0. (начиная с п. 2 методики и далее все действия выполняются в jupyter notebook, версия python 3.9.13).

3. Загрузка скачанного набора данных в jupyter notebook (рис. 1).

	address	year	day	length	weight	count	looped	neighbors	income	label
0	111K8kZAEnJg245r2cM6y9zgJGHZtJPy6	2017	11	18	0.008333	1	0	2	1.000500e+08	princetonCerber
1	1123pJv8jzeFQaCV4w644pzQJzVWay2zcA	2016	132	44	0.000244	1	0	1	1.000000e+08	princetonLocky
2	112536im7hy6wtKbpH1qYDWtTyMRAcA2p7	2016	246	0	1.000000	1	0	2	2.000000e+08	princetonCerber
3	1126eDRw2wqSkWosjTCre8cjjQW8sSeWH7	2016	322	72	0.003906	1	0	2	7.120000e+07	princetonCerber
4	1129TSjKtx65E35GiUo4AYVeyo48twbrGX	2016	238	144	0.072848	456	0	1	2.000000e+08	princetonLocky
...	...	...	...	...	...	...	...	...	...	...
2916692	12D3trgho1vJ4mGtWBRPyHdMJK96TRYSry	2018	330	0	0.111111	1	0	1	1.255809e+09	white
2916693	1P7PputTcVkhXBmXBvSD9MJ3UYPsiou1u2	2018	330	0	1.000000	1	0	1	4.409699e+07	white
2916694	1KYiKJEfdJtap9QX2v9BXJMpz2SfU4pgZw	2018	330	2	12.000000	6	6	35	2.398267e+09	white
2916695	15iPUJsRNZQZHmZZVwmQ63srsmughCXV4a	2018	330	0	0.500000	1	0	1	1.780427e+08	white
2916696	3LFFBxp15h9KSFtaw55np8eP5fv6kdK17e	2018	330	144	0.073972	6800	0	2	1.123500e+08	white

2916697 rows × 10 columns

Рис. 1. Фрагмент данных, скачанных из репозитория для машинного обучения

4. Автоматизированная выборка адресов (оставляем только столбцы «address» и «year»), связанных с программами вымогателями (значение поля label неравно «white»).

5. Удаление дубликатов в выборке адресов с шага № 4.

6. Сохранение очищенной выборки из 20849 адресов, связанных с программами вымогателями (рис. 2).

7. Автоматизированная выборка белых (значение поля label равно «white») адресов (оставляем только столбцы «address» и «year»).

8. Удаление дубликатов в выборке с шага № 7.

9. Формирование случайной выборки из выборки белых адресов размером 20849.

10. Сохранение очищенной выборки из 20849 белых адресов (рис. 3).

11. Загрузка выборки адресов, связанных с программами вымогателями, сохраненной на шаге № 6.

12. Загрузка списка транзакций для каждого адреса, связанного с программами вымогателями из выборки, загруженной на шаге № 11.

13. Загрузка выборки белых адресов, сохраненной на шаге № 10.

14. Загрузка список транзакций для каждого белого адреса из выборки, загруженной на шаге №13.

15. Выделение из списка транзакций для каждого белого адреса признаков (список признаков приведен в приложении 1).

16. Добавление столбцов выделенных признаков к выборке белых адресов.

17. Сохранение выборки белых адресов с признаками.

	address	year
0	111K8kZAEnJg245r2cM6y9zgJGHZtJPy6	2017
1	1123pJv8jzeFQaCV4w644pzQJzVWay2zcA	2016
2	112536im7hy6wtKbpH1qYDWtTyMRAcA2p7	2016
3	1126eDRw2wqSkWosjTCre8cjjQW8sSeWH7	2016
4	1129TSjKtx65E35GiUo4AYVeyo48twbrGX	2016
...	...	...
41396	1zYM55sSXHgjnL7ZNFo1YCd47UFLWpjfp	2014
41397	1ZZJDU36vDwiQB4YvtJR1pym7J5AwdqZi	2016
41401	1Zzq1TordGW9WjqRtSvjB6cFZ8xVi4Z8v	2016
41408	35iCvpMMnUWcSWrYtLJLXqe9xo5CYEWRhw	2017
41412	377CY1m8W2qbQQX5HHjziimdh2faGjDeLv	2016

20849 rows × 2 columns

Рис. 2. Фрагмент выборки адресов, связанных с программами вымогателями

	address	year
684670	13oSGjio38SSJS2VGMsfuwGMhh8eXzoPnf	2012
877696	18dRxoXDF4gtd9QeiDxCe3r6ZET14bZZVx	2013
2357016	12Hrqemq4gDkX1ZVzWq6GBcToJwRWRzbBC	2017
2892649	3EEnHoovBYTjc1KwhgPWgsAdt4oTtYq5RL	2018
1773788	1GnZmBWEb3QtPjHR3eaCM5WGUMTMHVLcnh	2015
...	...	...
2808203	1JjJNfszB6YedMFY3ddPhQWJ9wVzHCfuER	2018
411580	1LCMAjERbLPBPZwmLCTPGDJttDaRW7pNhh	2012
1333466	1JmyRUBoCQNNwfbzL4f5oLfBvezL5xKsGA	2014
2866340	1PpxthcbrotH2Lsg4frrCPBijj3jMc2Fyw	2018
478853	1rEM3PTKpjnksgcyQE9t9oy97uCHasTt7	2012

20849 rows × 2 columns

Рис. 3. Фрагмент выборки белых адресов

18. Повторение шагов 15–17 для адресов, связанных с программами вымогателями.

19. Загрузка созданных наборов данных на шагах 11–18.

20. Добавление в каждом наборе данных метки класса, 0 – белые адреса, 1 – связанные с программами вымогателями.

21. Конкатенация двух наборов данных, полученных на шаге 20.

22. Сортировка набора данных, полученного на шаге 21, по столбцу «year» (рис. 4, также в приложении 2 приведены средние значения признаков, полученного набора данных).

index	Адрес	Количество транзакций	Количество исходящих транзакций	Количество входящих транзакций	Среднее количество транзакций в день	Средний размер транзакции	Медиана размера транзакций	Дисперсия размера транзакций	Средний размер исходящей транзакции	
11	1H43gAKaUiYLYyD9rmQAqQmepdjGPSumaK	2	1	1	0.000478	5.250000e+10	5.250000e+10	0.000000e+00	5.250000e+10	5.
31	1F4xVpa6rpq4kZe4biUP9yHSs8rEhpiejZ	2	1	1	0.000470	3.892100e+09	3.892100e+09	0.000000e+00	3.892100e+09	3.
35	16rSUc57eUhBTZ3HyJwENvhAziP4FHJonh	2	1	1	0.000459	1.115200e+10	1.115200e+10	0.000000e+00	1.115200e+10	1.
43	1FnphuC37jmv7QRMyHbciwGTFLxyqzTZN6	2	1	1	0.000484	3.490000e+08	3.490000e+08	0.000000e+00	3.490000e+08	3.
56	1MBroicMwbtGJicvgGjvRpHnX2tTwpj782	2	1	1	0.000481	7.510151e+09	7.510151e+09	0.000000e+00	7.510151e+09	7.
...	...	...	...	...	...	...	...	...	...	
20817	13ApsGZqToySZRpFBaeZrc8EwC1kqNFUcL	2	1	1	0.001084	3.120060e+10	3.120060e+10	0.000000e+00	3.120060e+10	3.
20822	1AfMVMxL6TkgrMExkD4wdA9DfCmjazhH4X	2	1	1	0.001099	9.590660e+08	9.590660e+08	0.000000e+00	9.590660e+08	9.
20826	3Nc82VRiXSFERSuG8Wkze64X8GPgBVLpWk	2	1	1	0.001191	3.698453e+07	3.698453e+07	0.000000e+00	3.698453e+07	3.
20830	1gFEU7sEiW3Sginsj3gHkBBgBqdoMTR1x	4	2	2	0.002142	3.430000e+08	3.430000e+08	9.000000e+12	3.430000e+08	3.
20844	338M9gbU7HgcsWXYqXjwitF6vqVymXLVJk	2	1	1	0.001117	1.500000e+08	1.500000e+08	0.000000e+00	1.500000e+08	1.

41698 rows × 56 columns

Рис. 4. Фрагмент набора данных перед разделением на обучающую и тестовую выборки

23. Удаление столбцов «address» и «year», т.к. они не являются информативными.

24. Разделение набора данных на обучающую и тестовую выборки в соотношении 80% (33358 адресов) обучающая выборка, 20% (8340 адресов) – тестовая.

25. Стандартизация обучающей и тестовой выборок.

26. Обучение классификаторов и вычисление метрик качества классификатора на тестовой.

27. Подбор гиперпараметров классификаторов.

28. Обучение классификаторов с подобранными гиперпараметрами и вычисление метрик качества классификатора на тестовой выборке.

Результаты реализации описанной выше методики, обсуждаются далее.

Анализ результатов обучения бинарного классификатора

Напомним, что, де-факто, в рассматривае-

мом случае бинарный классификатор проверяет статистическую гипотезу о том, что данный является белым (гипотеза H_0), против статистической гипотезы о том, что данный кошелек связан с программой-вымогателем (гипотеза H_1). Известно, что бинарный классификатор при заданном пороге принятия решения может совершить ошибку первого рода (отвергнута верная гипотеза H_0) и ошибки второго рода (принята неверная гипотеза H_0) (табл. 1).

Из табл. 1 видно, что для оценки качества

Таблица 1

Матрица возможных ошибок классификатора

		Верная гипотеза	
		H_0	H_1
Результат классификации	H_0	H_0 верно принята (*True Positive* (*TP*))	H_0 неверно принята (ошибка второго рода, *False Positive* (*FP*))
	H_1	H_0 неверно отвергнута (ошибка первого рода, *False Negative* (*FN*))	H_0 верно отвергнута (*True Negative* (*TN*))

моделей можно использовать следующие метрики.

1) Accuracy (точность глобальная) – доля правильно классифицированных объектов от общего числа объектов,

$$Accuracy = \frac{\sum(TP+TN)}{\sum(TP+FP+FN+TN)}.$$

2) Precision (точность) – доля объектов, названными классификатором положительными, и при этом действительно являющимися положительными,

$$Precision = \frac{\sum TP}{\sum(TP+FP)}.$$

3) Recall (полнота) – доля найденных объектов положительного класса из всех объектов положительного класса

$$Recall = \frac{\sum` TP}{\sum(TP+FN)}.$$

4) F-мера – среднегармоническое метрик Precision и Recall, является комбинацией метрик точности (precision) и полноты (recall), используемых в оценке качества классификационной модели.

$$F_\beta = \left(1+\beta^2\right)\frac{Precision \cdot Recall}{\beta^2 \cdot Precision + Recall},$$

где β – параметр, принимающий значения в диапазоне $0 < \beta < 1$, если приоритет отдается точности, и $\beta > 1$, если приоритет отдается полноте.

При β=1 вы получается сбалансированная F-мера, также ее называемая F_1-мерой:

$$F_\beta = 2\frac{Precision \cdot Recall}{Precision + Recall},$$

Также для оценки качества модели используют ROC-кривую (Receiver Operating Characteristic curve), а также площадь под данной кривой (ROC–AUC). ROC-кривая – график, демонстрирующий изменения соотношения между долей верно-положительных и долей ложноположительных ответов классификатора при изменении порога принятия решения.

Построение ROC-кривой основывается на вычислении пары метрик: True Positive Rate (TPR) (доли верно определенных положительных объектов среди всех реально положительных объектов):

$$TPR = \frac{\sum TP}{\sum(FP+FN)},$$

и False Positive Rate (FPR) (доли ложноположительных ответов среди всех реально отрицательных объектов):

$$FPR = \frac{\sum` FP}{\sum(FP+TN)},$$

при различных значениях порога принятия решения.

Оценки выбранных значений метрик, вычисленные для каждого из использованных алгоритмов классификации на основе анали-

Таблица 2

Метрики классификаторов до подбора гиперпараметров

Алгоритм	*Accuracy*	*Precision*	*Recall*	F_1	*ROC-AUC*
Логистическая регрессия	88.09%	76.71%	98.51%	86.26%	93.86%
k-ближайших соседей	90.26%	80.49%	98.10%	88.43%	95.56%
Деревья принятия решений	88.57%	84.25%	85.93%	85.08%	88.06%
Случайный лес	95.35%	91.01%	97.34%	94.07%	98.76%
Градиентный бустинг	91.80%	83.13%	98.32%	90.09%	98.41%

за, выбранного датасета, представлены в таблице 2.

Из таблицы 2 видно, что по метрикам Accuracy, Precision, F_1-мере наилучшим является классификатор, на основе алгоритма «Случайный лес» (95.35%, 91.01% и 94.07%, соответственно), по метрике Recall – классификатор, построенный по алгоритму логистической регрессии (98.51%).

Для улучшения качества классификации была получена оценка оптимальных по метрике Recall значений гиперпараметров с помощью функции gridsearchcv из библиотеки sklearn [15]. (Выбор для оценки гиперпараметров данной метрики обусловлен тем, что финансовые потери в случае классификации адреса, связанного с программами-вымогателями как белый, окажутся выше, чем в случае отнесения белого адреса к адресам, которые связаны с программами-вымогателями.) Значения гиперпараметров классификаторов, оптимальные по данной метрике представлены в Приложении 3 выделены жирным шрифтом.

Из таблиц 2, 3 видно, что метрика Recall улучшилась у всех классификаторов, кроме классификатора, построенного по алгоритму «Градиентный бустинг».

Из таблицы 3 видно, что по метрике Recall классификатор, построенный по алгоритму логистической регрессии, является наилучшим.

Из рисунка 5 видно, что по кривым ROC наилучшими оказываются классификаторы на основе алгоритмов «Случайный лес» и «Градиентный бустинг». Данный вывод подтверждается сравнением соответствующих значений метрики ROC-AUC, представленных в таблице 3.

Таблица 3

Метрики качества классификации после подбора гиперпараметров

Алгоритм	*Accuracy*	*Precision*	*Recall*	F_1	*ROC-AUC*
Логистическая регрессия	85.19%	72.19%	99.15%	83.55%	95.89%
k-ближайших соседей	90.18%	80.33%	98.14%	88.35%	96.80%
Деревья принятия решений	88.98%	84.74%	86.53%	85.62%	88.50%
Случайный лес	95.34%	90.77%	97.63%	94.07%	98.85%
Градиентный бустинг	95.54%	92.40 %	96.14%	94.24%	98.78%

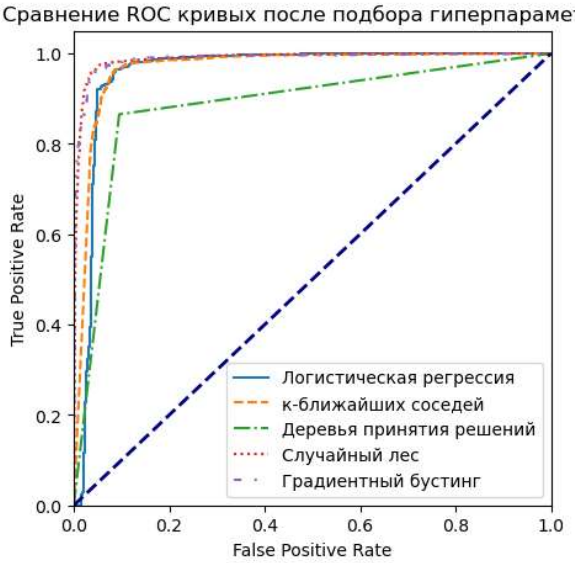

Рис. 5. Кривые ROC после подбора гиперпараметров по каждому классификатору отдельно

Заключение

Описана методика построения бинарного классификатора, предназначенного для обнаружения связи биткоин-кошелька с программами вымогателями и получены оценки метрик качества классификатора.

В настоящее время разрабатывается программное обеспечение, обеспечивающее использование разработанного классификатора в автоматическом режиме.

Приложение 1

Код признака	Тип данных	Признак
ПК1	Целое число	Количество транзакций
ПК2	Целое число	Количество исходящих транзакций
ПК3	Целое число	Количество входящих транзакций
ПК4	Число с плавающей точкой	Среднее количество транзакций в день
ПК5	Число с плавающей точкой	Средний размер транзакции
ПК6	Число с плавающей точкой	Медиана размера транзакций
ПК7	Число с плавающей точкой	Дисперсия размера транзакций
ПК8	Число с плавающей точкой	Средний размер исходящей транзакции
ПК9	Число с плавающей точкой	Средний размер входящей транзакции
ПК10	Число с плавающей точкой	Медиана размера исходящей транзакций
ПК11	Число с плавающей точкой	Медиана размера входящей транзакций
ПК12	Число с плавающей точкой	Дисперсия размера исходящей транзакции
ПК13	Число с плавающей точкой	Дисперсия размера входящей транзакции
ПК14	Число с плавающей точкой	Среднее количество выходов в транзакции
ПК15	Число с плавающей точкой	Среднее количество входов в транзакции
ПК16	Число с плавающей точкой	Медиана количества выходов транзакций
ПК17	Число с плавающей точкой	Медиана количества входов транзакций
ПК18	Число с плавающей точкой	Дисперсия количества выходов транзакций
ПК19	Число с плавающей точкой	Дисперсия количества входов транзакций
ПК20	Число с плавающей точкой	Средний размер выходов в транзакциях
ПК21	Число с плавающей точкой	Средний размер входов в транзакциях
ПК22	Число с плавающей точкой	Медиана размера выходов в транзакциях
ПК23	Число с плавающей точкой	Медиана размера входов в транзакциях

ПК24	Число с плавающей точкой	Дисперсия размера выходов в транзакциях
ПК25	Число с плавающей точкой	Дисперсия размера входов в транзакциях
ПК26	Целое число	Всего пришло
ПК27	Целое число	Всего ушло
ПК28	Целое число	Текущий баланс адреса
ПК29	Категориальный	Майнинг
ПК30	Число с плавающей точкой	Процент транзакций 1к1 от общего количества транзакций
ПК31	Число с плавающей точкой	Процент транзакций 1к2 от общего количества транзакций
ПК32	Число с плавающей точкой	Процент транзакций 1кn от общего количества транзакций
ПК33	Число с плавающей точкой	Процент транзакций nк1 от общего количества транзакций
ПК34	Число с плавающей точкой	Процент транзакций nк2 от общего количества транзакций
ПК35	Число с плавающей точкой	Процент транзакций nкn от общего количества транзакций
ПК36	Категориальный	Количество транзакций классы от 0 до 5 – 0 от 5 до 10 – 1 от 10 до 500 – 2 от 500 до 1000 – 3 более 1000 – 4
ПК37	Категориальный	Всего пришло классы От 0 до 1000000 – 0 От 106 до 108 – 1 От 108 до 1010 – 2
ПК38	Число с плавающей точкой	Процент входящих транзакций
ПК39	Число с плавающей точкой	Всего ушло в процентах от прихода
ПК40	Число с плавающей точкой	Процент транзакций вида 1 к 1 (1 вход 1 выход) во входящих транзакциях
ПК41	Число с плавающей точкой	Процент транзакций вида 1 к 2 во входящих транзакциях
ПК42	Число с плавающей точкой	Процент транзакций вида 1 к n во входящих транзакциях
ПК43	Число с плавающей точкой	Процент транзакций вида n к 1 во входящих транзакциях
ПК44	Число с плавающей точкой	Процент транзакций вида n к 2 во входящих транзакциях
ПК45	Число с плавающей точкой	Процент транзакций вида n к n во входящих транзакциях
ПК46	Число с плавающей точкой	Процент транзакций вида 1 к 1 в исходящих транзакциях
ПК47	Число с плавающей точкой	Процент транзакций вида 1 к 2 в исходящих транзакциях
ПК48	Число с плавающей точкой	Процент транзакций вида 1 к n в исходящих транзакциях
ПК49	Число с плавающей точкой	Процент транзакций вида n к 1 в исходящих транзакциях
ПК50	Число с плавающей точкой	Процент транзакций вида n к 2 в исходящих транзакциях
ПК51	Число с плавающей точкой	Процент транзакций вида n к n в исходящих транзакциях
ПК52	Категориальный	Какая из категорий входящих транзакций самая многочисленная: 1к1в, 1к2в, 1кnв, nк1в, nк2в, nкnв
ПК53	Категориальный	Какая из категорий исходящих транзакций самая многочисленная: 1к1и, 1к2и, 1кnи, nк1и, nк2и, nкnи.

Количество транзакций	22.182478775960476
Количество исходящих транзакций	10.80118950549187
Количество входящих транзакций	11.381289270468608
Среднее количество транзакций в день	0.008139250389723057
Средний размер транзакции	1849902835.4203086

Медиана размера транзакций	1807736189.7875679
Дисперсия размера транзакций	1.4202734846776867e+19
Средний размер исходящей транзакции	1882746895.5444891
Средний размер входящей транзакции	1847731798.8238063
Медиана размера исходящей транзакций	1830266922.8556166
Медиана размера входящей транзакций	1806987378.2882392
Дисперсия размера исходящей транзакции	3.2088310541193163e+19
Дисперсия размера входящей транзакции	1.384475715826054e+19
Среднее количество выходов в транзакции	8.081085307284921
Среднее количество входов в транзакции	14.566894987781923
Медиана количества выходов транзакций	6.270036932226965
Медиана количества входов транзакций	11.291524773370426
Дисперсия количества выходов транзакций	4571.6406336007285
Дисперсия количества входов транзакций	1287.5838249290462
Средний размер выходов в транзакциях	3297444014.463111
Средний размер входов в транзакциях	2523040844.6766396
Медиана размера выходов в транзакциях	1580427843.1305819
Медиана размера входов в транзакциях	2164663459.1524653
Дисперсия размера выходов в транзакциях	1.1409921673417399e+22
Дисперсия размера входов в транзакциях	7.460948314081068e+20
Всего пришло	11142722388.631445
Всего ушло	11136846338.193558
Текущий баланс адреса	5876050.437886709
Процент транзакций 1к1 от общего количества транзакций	0.1296966608158774
Процент транзакций 1к2 от общего количества транзакций	0.3895537526547267
Процент транзакций 1кn от общего количества транзакций	0.06094333250887102
Процент транзакций nк1 от общего количества транзакций	0.04395548824553067
Процент транзакций nк2 от общего количества транзакций	0.2843886088695863
Процент транзакций nкn от общего количества транзакций	0.09146215690540795
Процент входящих транзакций	0.5071025900266724
Всего ушло в процентах от прихода	0.9971371010022559
Процент транзакций вида 1 к 1 (1 вход 1 выход) во входящих транзакциях	0.02204091516666529
Процент транзакций вида 1 к 2 во входящих транзакциях	0.552395876147164
Процент транзакций вида 1 к n во входящих транзакциях	0.10077641400062447
Процент транзакций вида n к 1 во входящих транзакциях	0.015264663142277958
Процент транзакций вида n к 2 во входящих транзакциях	0.20468302062377555
Процент транзакций вида n к n во входящих транзакциях	0.10483911091948928
Процент транзакций вида 1 к 1 в исходящих транзакциях	0.2368923957560337
Процент транзакций вида 1 к 2 в исходящих транзакциях	0.21869017234638402
Процент транзакций вида 1 к n в исходящих транзакциях	0.01803797508019239
Процент транзакций вида n к 1 в исходящих транзакциях	0.07646239849603213
Процент транзакций вида n к 2 в исходящих транзакциях	0.37087856970691463
Процент транзакций вида n к n в исходящих транзакциях	0.07656834616156667

Приложение 3

Название гиперпараметра	**Значения**
Логистическая регрессия	
penalty	l1, **l2**, elasticnet, none
C	**0.1**, 0.5, 1.0, 5.0
solver	lbfgs, liblinear, **sag**
max_iter	**100**, 500, 1000
class_weight	**None**, balanced

tol	**0.0001**, 0.001, 0.01
l1_ratio	**0.1**, 0.5, 0.9
К-ближайших соседей	
n_neighbors	3, 5, 7, 9, **15**
weights	uniform, **distance**
algorithm	**auto**, ball_tree, kd_tree, brute
Дерево решений	
criterion	gini, **entropy**, log_loss
splitter	**best**, random
max_depth	**None**, 50, 100, 500
Случайный лес	
n_estimators	50, **100**, 150
criterion	gini, **entropy**, log_loss
max_depth	None, 50, 100, **500**
Градиентный бустинг	
learning_rate	0.1, 0.01, 0.2, **0.5**
n_estimators	50, 100, **150**
criterion	**friedman_mse**, squared_error
max_depth	1, 3, 5, **10**, 15

Литература

1. Федеральный закон «О противодействии легализации (отмыванию) доходов, полученных преступным путем, и финансированию терроризма» от 07.08.2001 № 115-ФЗ.

2. URL: https://www.rbc.ru/crypto/news/6188cfe09a79472892042eba (Дата обращения: 06.04.2023)

3. URL: https://www.investopedia.com/terms/b/blockchain.asp (Дата обращения: 06.04.2023)

4. URL: https://www.bestchange.ru/ (Дата обращения: 06.04.2023)

5. URL: https://accounts.binance.com/ru/register?ref=78176083 (Дата обращения: 06.04.2023)

6. URL: https://payeer.com/ru/ (Дата обращения: 06.04.2023)

7. URL: https://advcash.com/ (Дата обращения: 06.04.2023)

8. URL: https://www.blockchain.com/ru/ (Дата обращения: 06.04.2023)

9. URL: https://www.reuters.com/technology/crypto-money-laundering-rises-30-2021-chainalysis-2022-01-26/ (Дата обращения: 13.04.2023)

10. URL: https://archive.ics.uci.edu/ml/datasets/BitcoinHeistRansomwareAddressDataset (Дата обращения: 06.04.2023)

11. URL: https://www.kaggle.com/datasets/ellipticco/elliptic-data-set (Дата обращения 13.04.2023)

12. D. R. Cox. The Regression Analysis of Binary Sequences, Journal of the Royal Statistical Society. Series B (Methodological), Vol. 20, No. 2 (1958), P. 215–242.

13. Fix Evelyn, Hodges Joseph L. Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties. USAF School of Aviation Medicine, Randolph Field, Texas (1951).

14. L. Breiman, J. Friedman, R. Olshen, and C. Stone. Classification and Regression Trees. Wadsworth, Belmont, CA, 1984.

15. Breiman, L. «Random forests» Machine Learning (45), 5–32 (2001).

16. Jerome H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. The Annals of Statistics. Vol. 29, No. 5 (Oct., 2001), P. 1189–1232 (44 pages).

17. URL: https://scikit-learn.org/stable/ (Дата обращения: 10.04.2023)

18. URL: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html (Дата обращения: 10.04.2023)

References

1. Federal'nyj zakon «O protivodejstvii legalizacii (otmyvaniju) dohodov, poluchennyh prestupnym putem, i finansirovaniju terrorizma» ot 07.08.2001 № 115-FZ

2. URL: https://www.rbc.ru/crypto/news/6188cfe09a79472892042eba (Data obrashhenija: 06.04.2023)

3. URL: https://www.investopedia.com/terms/b/blockchain.asp (Data obrashhenija: 06.04.2023)

4. URL: https://www.bestchange.ru/ (Data obrashhenija: 06.04.2023)

5. URL: https://accounts.binance.com/ru/register?ref=78176083 (Data obrashhenija: 06.04.2023)

6. URL: https://payeer.com/ru/ (Data obrashhenija: 06.04.2023)

7. URL: https://advcash.com/ (Data obrashhenija: 06.04.2023)

8. URL: https://www.blockchain.com/ru/ (Data obrashhenija: 06.04.2023)

9. URL: https://www.reuters.com/technology/crypto-money-laundering-rises-30-2021-chainalysis-2022-01-26/ (Data obrashhenija: 13.04.2023)

10. URL: https://archive.ics.uci.edu/ml/datasets/BitcoinHeistRansomwareAddressDataset (Data obrashhenija: 06.04.2023)

11. URL: https://www.kaggle.com/datasets/ellipticco/elliptic-data-set (Data obrashhenija 13.04.2023)

12. D. R. Cox. The Regression Analysis of Binary Sequences, Journal of the Royal Statistical Society. Series B (Methodological), Vol. 20, No. 2 (1958), P. 215–242

13. Fix Evelyn, Hodges Joseph L. Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties. USAF School of Aviation Medicine, Randolph Field, Texas (1951)

14. L. Breiman, J. Friedman, R. Olshen, and C. Stone. Classification and Regression Trees. Wadsworth, Belmont, CA, 1984.

15. Breiman, L. «Random forests» Machine Learning (45), 5–32 (2001).

16. Jerome H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. The Annals of Statistics. Vol. 29, No. 5 (Oct., 2001), P. 1189–1232 (44 pages)

17. URL: https://scikit-learn.org/stable/ (Data obrashhenija: 10.04.2023)

18. URL: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html (Data obrashhenija: 10.04.2023)

АНДРЕЕВ Александр Михайлович, аспирант, ассистент Учебно-научного центра «Информационная безопасность» федерального государственного автономного образовательного учреждения высшего образования «Уральский федеральный университет им. первого Президента России Б.Н. Ельцина». Россия, 620002, г. Екатеринбург, ул. Мира, 32. E-mail: a.m.andreev@urfu.ru

ПОРШНЕВ Сергей Владимирович, доктор технических наук, профессор, директор Учебно-научного центра «Информационная безопасность» федерального государственного автономного образовательного учреждения высшего образования «Уральский федеральный университет им. первого Президента России Б.Н. Ельцина». Россия, 620002, г. Екатеринбург, ул. Мира, 32. E-mail: s.v.porshnev@urfu.ru

ANDREEV Alexander Mikhailovich, postgraduate student, assistant of the Educational and Scientific Center «Information Security» of the Federal State Autonomous Educational Institution of Higher Education «Ural Federal University named after the first President of Russia B.N. Yeltsin». Russia, 620002, Yekaterinburg, st. Mira, 32. E-mail: a.m.andreev@urfu.ru

PORSHNEV Sergey Vladimirovich, Doctor of Technical Sciences, Professor, Director of the Educational and Scientific Center «Information Security» of the Federal State Autonomous Educational Institution of Higher Education «Ural Federal University named after the first President of Russia B.N. Yeltsin». Russia, 620002, Yekaterinburg, st. Mira, 32. E-mail: s.v.porshnev@urfu.ru