Enhancing Trust in the Cryptocurrency Marketplace: A Risk Scoring Approach

Sudip Bhattacharyya[1], Dan Freeman[1], Timothy McWilliams[1],
Craig Hall[2], Pablo Peillard[2]

[1]Southern Methodist University
6425 Boaz Lane
Dallas, TX 75205
{sbhattacharyya, freemand, tmcwilliams}@smu.edu

[2]Caudicum
1920 McKinney Ave., Suite 750
Dallas, TX 75201
{craig, pablo}@caudicum.com

Abstract. In this paper, we introduce a risk scoring mechanism for cryptocurrency users. While the decentralized architecture of blockchain technology on which cryptocurrencies operate has many benefits, the anonymity of users in this space has provided criminal users an alternative to cash for harboring their illicit activities. In order to distinguish law-abiding cryptocurrency users from potential threats in the Bitcoin marketplace, we analyze historical hacks and thefts to profile transactions, classify them into risky and non-risky categories using several machine learning techniques and finally develop a risk score for every unique user based on their past engagement in any unlawful Bitcoin incident. We find that all of our machine learning models have produced an outstanding level of accuracy in classifying transactions. All of the baseline models we develop using Random Forest, *K*-Nearest Neighbors and Support Vector Machine algorithms are over 95% accurate. A successful classification leads to a reliable risk scoring method. The idea of flagging potential risky users acts as a guideline for safe Bitcoin transactions. In a cryptocurrency marketplace where intermediaries are non-existent, a self-attestation mechanism in the form of a risk score offers a viable alternative to enhance trust.

Commented [A1]: You may be able to remove this section of the abstract. The language before and after this is sufficient.

1 Introduction

Recent advances in distributed and parallel networking using internet-ubiquitous ~~accessible~~ services are allowing a re-engineering of previously centralized system design and authority. This new shared access has created the opportunity for nonmediated trust via a combination using the light of transparency with the trust validation of immutability with a historical record by the community rather than authorized mediator(s) declaring what is the declared trusted base record.

While this disintermediation and a more radical de-mediation of third party authorities using blockchains and smart contracts is in its early development for many new services and applications, its use in digital assets like cryptocurrency has seen significant activity.

In this radical ~~borderless~~ self-sovereign asset cybercommunity, without a central authority declaring the value of digital assets and marketplaces, ~~has come~~ a new ~~the~~ challenge of avoiding anarchy ~~and self-policing norms that were~~ once handled ~~with~~ by sanctions and settlements has come to the forefront. ~~by designated central authorities (governments and central banks).~~

Transparency has a long history as a tool for oversight in anti-corruption, anti-trust and anti-money laundering movements. Transparency in money and banking have frequently focused around a few sentences written by U.S. Supreme Court Justice Louis Brandeis (1916–1939), who wrote over one hundred years ago shortly before his time on the court in his book *Other People's Money and How the Bankers Use It* that: "Publicity is justly commended as a remedy for social and industrial diseases. Sunlight is said to be the best of disinfectants; electric light the most efficient policeman. And publicity has already played an important part in the struggle against the Money Trust" [1].

Transparency is not new, but when combined with shared, distributed blockchain records and ledgers it becomes a powerful tool for the disinfecting light of knowledge and reputation among participants in a society. The parallels between Brandeis' world in 1914 and the world today are striking. His writings that followed laid out the details of suspect transactions and his proposal to require additional information. "[T]he disclosure must be real . . . To be effective, knowledge of the facts must be actually brought home to the investor" [1].

In spite of its secure environment, the cryptocurrency marketplace has experienced numerous unlawful activities since its inception, which has caused asset losses to users. ~~suffer from losing assets.~~ This issue motivates us to develop a trust mechanism in the form of a risk score for all users operating in a blockchain network. The proposed risk scoring mechanism aims to enable~~s~~ users to identify and avoid risky parties before~~to deal with while~~ making transactions.

In this paper, we lay out the details to extend and embellish the principles of transparency and open transactions with the use of risk scoring knowledge for self-policing in the cryptocurrency marketplace. The remainder of this paper is structured as described here. Section 2 contains an overview of blockchain and the current cryptocurrency marketplace. Section 3 describes how to protect cryptocurrency from external attacks. In Section 4, we outline the creation of the blacklist and our risk scoring approach in detail. We describe the dataset used in this study in Section 5. Section 6 explains the machine learning algorithms we use and the corresponding measurements of the performance of the models. In Section 7, we present the results of our analyses. We examine the ethical implications of profiling and scoring cryptocurrency users in Section 8. In Section 9, we present conclusions from our analyses. Finally, Section 10 concludes this paper by highlighting future possible improvements on our work.

Commented [A2]: Wordy. Reduce or split into two sentences.

2 Blockchain and Emergence of Cryptocurrency

Put simply, blockchain is a network system and cryptocurrencies are products that operate in a blockchain network. Cryptocurrency is a medium for exchanging assets between individuals. It is defined as a digital currency or asset for which transactions are made in an environment secured by a cryptographic encryption-decryption mechanism. The world's first cryptocurrency, known as Bitcoin, was conceptualized by Satoshi Nakamoto, a pseudonym of a person or a group of people, in 2008. The platform for cryptocurrency transactions is known as blockchain which was described by Satoshi as "a peer to peer electronic cash system" [2].

Blockchain is a distributed ledger where transactions are periodically recorded in a chain of blocks and stored across multiple computers or servers, known as nodes, distributed over the network. Unlike ledgers in our conventional banking system, blockchain has no single point of control. Blockchain is ~~developed~~ based on certain characteristics of a distributed ledger system: cryptography, replication of ledger and immutability.

Transactions in a blockchain network are made secure with the cryptographic encryption. A public key-private key infrastructure ensures authenticity and integrity of a transaction. When a sender approves a transaction consisting of the amount, a fee and the recipient's public key, it is broadcast to the entire network on behalf of the sender's private key. Network nodes then validate the authenticity of the transaction while verifying sender's private key. Once authenticated, the sender's confirmation gets encrypted with a cryptographic hash algorithm. The recipient receives the asset from the transaction with their private key that helps to decrypt the encrypted transaction. ~~After~~ Upon completion, the transaction gets appended to other transactions in a block or from the creation of ~~creates~~ a new block in the distributed ledger system. The ~~and all the~~ participants receive a replication of the ledger updated with digitized information of all historical transactions. With the concept of this replicated ledger, blockchain maintains transparency, eliminates the chance of dispute and thereby makes the intermediaries unnecessary. Moreover, the identity of any user is masked with a hashed communication which ensures anonymity in a blockchain network. A typical cryptocurrency transaction on a blockchain network is illustrated in Figure 2.1.

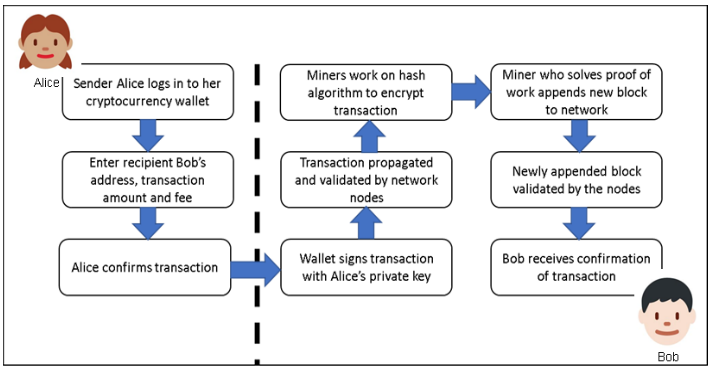

Alice
Sender Alice logs in to her cryptocurrency wallet
Enter recipient Bob's address, transaction amount and fee
Alice confirms transaction
Wallet signs transaction with Alice's private key
Transaction propagated and validated by network nodes
Miners work on hash algorithm to encrypt transaction
Miner who solves proof of work appends new block to network
Newly appended block validated by the nodes
Bob receives confirmation of transaction
Bob

Figure 2.1. Blockchain transaction cycle [9]

Another key feature of the blockchain is immutability. After a transaction is added to the network, it is impossible to remove~~erase it,~~. O~~o~~nly a reverse transaction can~~ be added to~~ nullify the effect of the erroneous or unintentional transaction. This immutable nature of the blockchain makes the length of the chain ever-growing. The advantages of a blockchain network are summarized in Table 2.1.

Table 2.1. Blockchain advantages.

Advantages	Description
Decentralization	In contrast to a centralized system, a blockchain database is distributed across nodes connected in a distributed network. The network operates peer-to-peer, with the nodes together managing the database.
Durability and robustness	Since it is built on the Internet, blockchain automatically inherits the durability of the Internet. Moreover, since it cannot be controlled by any single entity or node and there is no single point of failure, blockchain is expected to produce a more robust result.
Transparency and incorruptibility	Blockchain works in a consensus. A self-auditing system reconciles transactions in regular intervals. Any block of transactions is visible to all the participants and data cannot be altered, once validated and entered.
Enhanced security	First, with a distributed database architecture, a threat of attack on any centralized point is eliminated. Moreover, the proof-of-work mechanism and the use of hash functions and public-private keys make this blockchain network very secure from attack.

As part of the Bitcoin concept, Nakamoto developed the first-ever blockchain database where the genesis block (the first block) has a timestamp of 18:15:05 GMT on 3 January 2009 [2]. Since the introduction of Bitcoin in 2009, the world economy has experienced an exponential growth both in terms of market capitalization as well as in number of users in the cryptocurrency market. In the cryptocurrency market, users' identities are masked and are known by their respective public keys. In the past few years, this market has experienced an exponential growth in the number of active users, i.e., in the number of unique public keys in the network. The current number of users is reported to be more than 23 million and is expected to ~~cross~~ reach 200 million by 2024 at the present growth rate [3]. The cryptocurrency market is extremely volatile and sensitive to market demand. In a short span of 9 years, cryptocurrencies have become a multi-billion-dollar marketplace worldwide. Even though the market was conservative in nature for first few years, it started to gain momentum in 2016. By the end of 2017, the total market value skyrocketed to a record high of over 600 billion USD due to a ~~huge~~ surge in demand, mostly from China and Japan. As of Q1 2018, the market was valued at slightly more than 400 billion USD [4]. The trend in the global market capitalization of cryptocurrencies is shown in Figure 2.2.

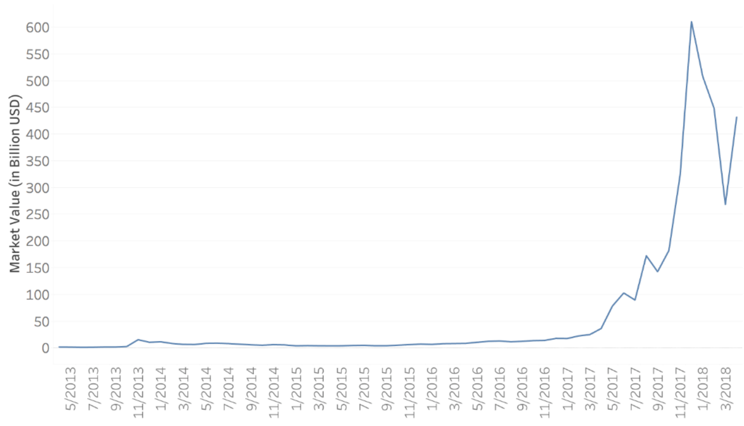


Figure 2.2. Total market capitalization of cryptocurrencies

Following Bitcoin, numerous other cryptocurrencies have emerged to market such as Litecoin, Ethereum, Ripple, Dash, Monero, among others. Being the debutant in the industry, Bitcoin had enjoyed almost a monopoly with over 70% share of global cryptocurrency market until early 2017. Slowly, other players entered the market, and, to date, there are more than 1,500 types of cryptocurrencies in operation. Since its inception in 2013, Ethereum has emerged as the main competitor to Bitcoin based on market share. As of April 2018, Bitcoin captures almost 37% of the cryptocurrency market, followed by Ethereum (16%), Ripple (8%), and Bitcoin Cash (6%) (Figure 2.3) [4].

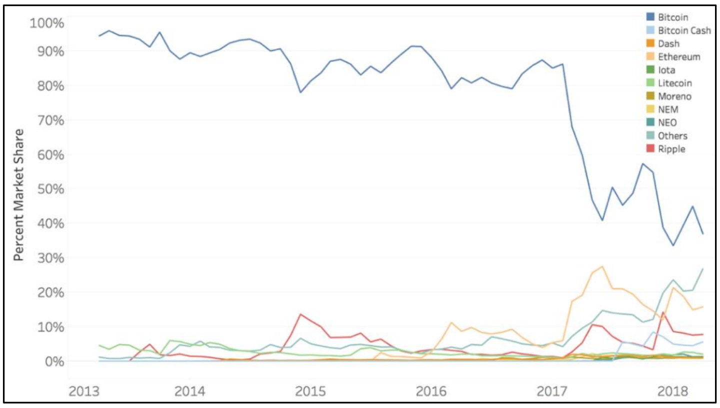


Figure 2.3. Global market share of major cryptocurrencies

Exchanges are an essential component in cryptocurrency transactions. Exchanges are digital trading platforms which offer a marketplace for trading digital assets. Users can buy or sell cryptocurrencies in exchange for conventional currencies or other cryptocurrencies. The cryptocurrencies bought through an exchange are stored in customers' cryptocurrency wallets. A wallet generally stores the public and private keys required to make a successful transaction of crypto assets and monitors customers' funds. The first exchange started to operate in 2010 [5]. We see numerous exchanges across the world operating cryptocurrency trades. Bitfinex, Bitstamp, MtGox, BTCChina, Huboi, OKChain are some of the popular exchanges in present market. Table 2.2 shows that three exchanges – OKChain, Huboi and BTCChina collectively cover almost 84% of total bitcoin trades whereas the oldest exchange, MtGox, has only 3.3% of share in Bitcoin total trade volume. A graphical representation of market share for popular exchanges are shown in Figure 2.4.

Table 2.2. Trade volumes and market share for major exchanges [6]

Exchange	BTC Volume	Market Share
OKChain	586,565,939.90	35.76%
Huboi	544,852,241.80	33.21%
BTCChina	258,007,950.80	15.73%
Mt.Gox	53,861,912.21	3.28%
Bitfinex	45,461,276.87	2.77%
Bitstamp	25,910,946.75	1.58%
Lakebtc	20,767,853.48	1.27%
Bitflyer	15,022,795.60	0.92%
Coinbase	14,915,059.72	0.91%
Others	75,147,603.32	4.58%

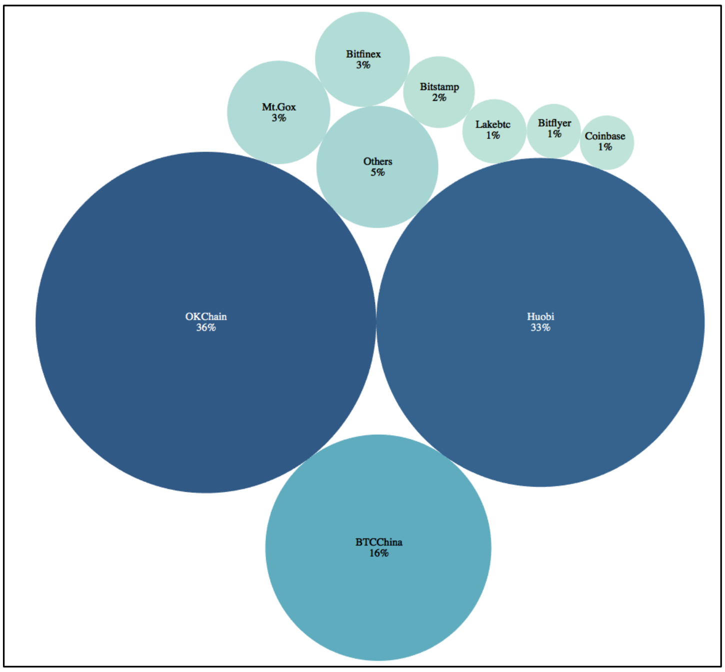


Figure 2.4. Percent of market share for major exchanges [6]

3 Protecting Assets

Blockchain, the operating platform of the cryptocurrencies~~ like Bitcoin~~, appears to be a secure environment. Use of cryptography as a security mechanism enables blockchain to protect digital assets and their transactions from any external attack. The two most important steps to maintain security are: 1~~i~~) protect user's private key and 2~~ii~~) secure a user's wallet.

3.1 Protecting User's Private Key

~~The ~~U~~u~~sers need assurance~~to ensure~~ that the private key is securely stored ~~in a secure way ~~and protected from any unauthorized access. The device that stores the private key needs to be physically secure~~d~~ and ~~strongly~~ password-protected. The loss of a private key could assist ~~will~~ ~~help~~ attackers masquerade as the user~~'s identity~~ and take control of the user's assets and transactions [7].

3.2 Securing User's Wallet

Protecting digital wallets is necessary as it stores digital assets. Measures such as encrypting the wallet and maintaining proper backup of the wallet are always important. Encryption ensures protection from any unauthorized access to the wallet while a backup becomes useful to retrieve the wallet in case the device itself is damaged or stolen. An offline wallet, also known as cold storage, can be maintained to store the majority of a user's assets, and~~. As it is not connected to a network, it~~ is ~~in no way~~ not vulnerable to network security breaches. A multi-signature feature which requires majority of the wallet holders' signature to approve a transaction can be applied to provide the highest level of security of a digital wallet [8].

Commented [A3]: "Majority" definition is somewhat vague here. Explain in greater detail.

3.3 Secured but Vulnerable

No system ~~turns out to be~~ is currently 100% secure ~~full-proof and completely secured in the real world~~ and blockchain is no exception. With the ever-growing popularity of cryptocurrency trading, ~~there comes a~~ the risk of security breaches and attacks is real. The crypto world has experienced several attacks and thefts which has caused considerable loss of funds. The largest bitcoin exchange, Mt. Gox, was hacked in early 2014 where bitcoins valued at approximately €460 million were stolen by hackers [9]. This incident forced them to file for bankruptcy. In January 2018, it was reported that NEM tokens worth $533 million were stolen from Tokyo-based exchange Coinchek [10]. Some of the major Bitcoin hacks are listed in Table 3.1[11]

Table 3.1. Major Bitcoin heists [11].

Hack	Time	Severity (BTC)
Bitcoin Savings and Trust	2011–2012	~ 263,024
Silk Road Seizure	Oct-13	171,955.09
MyBitcoin Theft	Jul-11	78,739.58
Linode Hacks	Mar-12	46,653.46
July 2012 Bitcoinica Theft	Jul-12	40,000
May 2012 Bitcoinica Hack	May-12	18,547.66
Allinvain Theft	Jun-11	25,000.01
Tony Silk Road Scam	Apr-12	~ 30,000
Bitfloor Theft	Sep-12	24,086.17
Bitomat.pl Loss	Aug-11	~ 17,000

Commented [A4]: For entire column, round to nearest dollar and make clear what the denomination is. Also, maintain consistency with year(s) in Time column, e.g. 2012 and 12.

Bitcoin7 Hack	Oct-11	~ 15,000
Cdecker Theft	Sep-12	9,222.21
Stefan Thomas Loss	Jun-11	~ 7,000
BTC-E Hack	Jul-12	~ 4,500
Inputs.io Hack	Oct-13	~ 4,100
Mass MyBitcoin Thefts	Jun-11	4,019.42
Mooncoin Theft	Sep-11	~ 4,000
Kronos Hack	Unknown	~ 4,000
Bitcoin Rain	2011–2013	~ 4,000
2012 Trojan	Sep – Nov 2012	~ 3,457
Betcoin Theft	Apr-12	3,171.5
June 2011 Mt. Gox Incident	Jun-11	2,643.27
October 2011 Mt. Gox Loss	Oct-11	2,609.36
Andrew Nollan Scam	Feb-12	2,211.07
Bit LC Theft	Feb-13	~ 2,000
Bitcoin Syndicate Theft	Jul-12	1,852.61
ZigGap	2012	1,708.65
Just Dice Incident	Jul-13	1,300
BTCGuild Incident	Mar-13	1,254
2012 50BTC Theft	Oct-12	1,173.51
Ubitex Scam	2011	1,138.98
Bitscalper Scam	2012	~ 1,000

We build a statistically rigorous mechanism which empowers the cryptocurrency users with a self-guided trust. This mechanism will lay the foundation for strategic decision making of users while making transactions in a blockchain network.

Commented [A5]: Make the transition smoother. You went from talking about historical bitcoin heists and straight to your methodology.

4 Transaction Profiling and a Risk Scoring Approach

Risk is inherent in any transaction. The potential for fraudulent motives on the part of participants in a transaction always exists. While certain measures can be taken to reduce such risk, invariably, each party places trust in the other to fulfill ~~do good on~~ their end of the deal. In order to mitigate ~~the~~ risk, we propose a two-stage solution. First, we create a blacklist of addresses that are known to be associated with one or more of the major hacks on Bitcoin exchanges. Second, we build a scoring model to identify transactions as risky or safe.

4.1 The Blacklist

Blacklisting is one of the primary techniques that organizations use to protect the public from financial scams, malicious web pages and many other mischiefs on the Internet [12]. For example, SpamCop, an email spam reporting service, uses this approach for listing the IP addresses of entities who send spam. The SafeBrowsing API, a Google service, uses this approach for listing URLs that lead to malicious web pages. Invaluement, an anti-spam service, employs blacklisting on IP addresses or domain names that are involved in scams [12]. These are just a few examples of how organizations use blacklisting. In our context, creating a blacklist is the action of grouping entities with whom law-abiding citizens should distrust and avoid making contact. These entities should be known in order to properly educate and protect users.

The blacklist is continuously updated with known entities so that users can check before engaging in potentially harmful interactions**.** Once the entity is found, the scoring mechanism can generate a warning to let the user know that this entity is on the blacklist and cannot be trusted. As such, a blacklist provides the benefit of lookup efficiency [12]. One key limitation of the blacklist is that it operates in a reactive fashion. The list is updated only after an entity has been found to engage in illegal activity. ~~However, until then,~~ This leaves ~~other~~ entities ~~are~~ vulnerable to ~~the~~ malicious attacks. ~~intentions of the criminal entity.~~

For Bitcoin transactions, a baseline is required to determine which addresses are associated with illegal activities. Certain transactions are associated with criminal actions, hence the user that made the transaction should not be trusted. Events such as the Mt. Gox hacks, May 2012 Bitcoinica Hack, BTC-E Hack and the Allinvain Theft are all instances in which certain addresses were used to commit illegal activities. An address associated with any of these hacks is placed on a blacklist. If a user transacts with one of the addresses on this blacklist, then that user becomes riskier for others to engage in transactions. For example, if User A is associated with the Mt. Gox hacks and User B transacts with User A, then the riskiness associated with doing business with User B increases. The blacklist makes this information publicly available. Therefore, if User C wants to send Bitcoins to User B, they could see that User B has done transactions with the blacklisted User A in the past; ~~this~~ which may make User C hesitatante~~e~~ to transact with User B.

Commented [A6]: Explain why the user who created the transaction cannot be trusted.

4.2 Risk Scoring Approach

A risk score is a means by which a user can gauge the ~~relative~~ potential risk for ~~any~~ future transactions. The observed transactions are profiled based on the known circumstances and categorized accordingly. In this study, transactions are categorized two ways: a hack or a non-hack. A hack is a transaction that has been involved with any type of scam, theft or heist. The address ID's that are categorized as a hack come from the blacklist. ~~The n~~Non-hack instances are labeled as 0 and the hack instances are labeled as 1.

Every address ID has numerous transactions associated with it. Each transaction, for every address ID, is classified into a hack or a non-hack. A risk score is then developed and assigned against every address ID. The risk score, a number between 0 and 1, is defined as:

$$\frac{h}{t}\,, \tag{1}$$

where t represents the total number of transactions per address ID and h represents the total number of transactions classified as a hack per address ID.

Addresses with higher scores are to be avoided because of their higher engagement in suspicious cryptocurrency dealings. Addresses with lower scores are considered to be less risky with which to make transactions. If a risk score is 0, then this particular address ID has never been associated with any type of malicious activities. A pictorial explanation of our risk scoring mechanism is displayed in Figure 4.1. In this instance, we can clearly see Bob and Alice are safer entities to transact with than Trudy.

Commented [A7]: Maybe make it simpler by stating that increased risk is associated with higher scores.

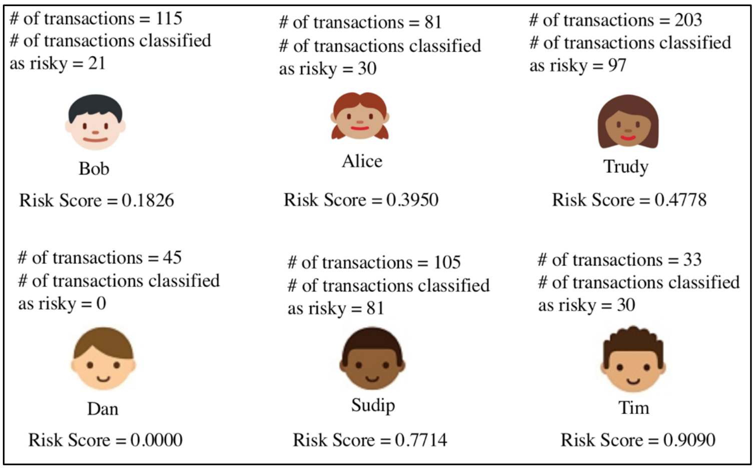


Figure 4.1. Risk scoring examples

In addition to the risk scoring metric, we introduce a flagging criterion for all the unique address ID’s to indicate their risk level. Figure 4.2 displays the different risk levels: green (low), yellow (medium) and red (high).

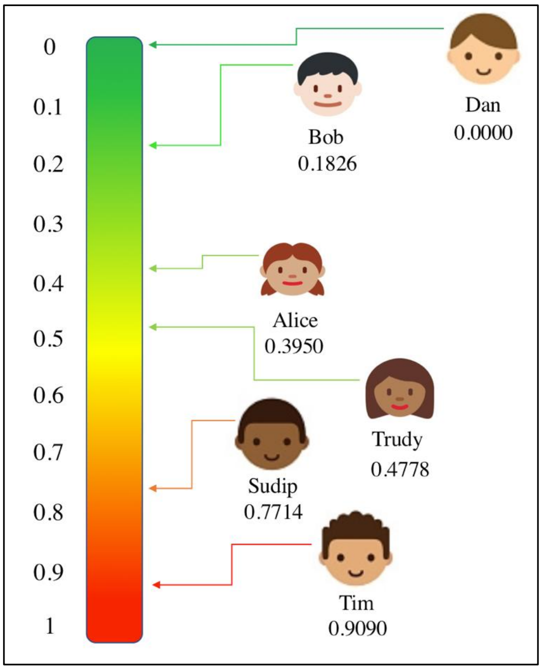


Figure 4.2. Levels of the risk score.

5 Blockchain Data

To build a scoring model, transactional data for bitcoin with details on sender, receiver, transaction time and amount was our minimum requirement. For this purpose, we leverage a database on historical Bitcoin transactions from the ELTE Bitcoin Project[1].This database provides block- and transaction-level details for every Bitcoin transactions recorded until February 9, 2018. The database contains 508,241 blocks and the total number of unique transactions is 303,641,057. The database contains 9 tables, each of them capturing information on a specific aspect of transactions provided by a specific set of attributes.

~~All the tables are connected to each other either directly or through another table.~~ A primary key, which is unique with respect to any particular table, enables the mapping between tables. For example, *blockID* is the primary key for table bh and is also part of

[1] Data source, [9 - http://www.vo.elte.hu/bitcoin/default.htm].

the table *tx*, hence *bh* and *tx* can be mapped with *blockID*. Similarly, *txID* is used to map the table *txin* to the table *tx*. The schema for our database is presented in Figure 5.1. All tables are joined together to create a single dataset containing all information without any duplication of attributes.

Table 5.1. Table view.

Table	Fields/Attributes
bh	blockID
	hash
	block_timestamp
	n_txs
txh	txID
	hash
tx	txID
	blockID
	n_inputs
	n_outputs
txin	txID
	input_seq
	prev_txID
	prev_output_seq
	addrID
	Sum
txout	txID
	output_seq
	addrID
	sum
multiple	txID
	output_seq
	addrID
nonstandard	txID
	output_seq
addresses	addrID
	address
addr_sccs	addrID
	userID

Table 5.2. Attribute descriptions.

Fields/Attributes	Description
blockID	Unique identifier for each block in bitcoin transaction history
hash	An encrypted hex value for block address
block_timestamp	Current time in seconds calculated based on 1970-01-01-00:00:00 as starting time

n_txs	Total number of transactions registered in a block
txID	Unique identifier for each transaction
n_inputs	Number of inputs in one transaction
n_outputs	Number of outputs in one transaction
input_seq	Sequence for inputs in a specific transaction
prev_txID	Unique identifier for the previous transaction
prev_output_seq	Previous output sequence for the output prior to the current output
addrID	Unique integer value for each address
sum	The total input amount in a transaction which is defined in Satoshis (1e-9 BTC)
output_seq	Sequence for outputs in a specific transaction
address	Unique identifier of 26-35 alphanumeric characters, beginning with the number 1 or 3, that represents a possible destination for payments
userID	Unique identifier for a user

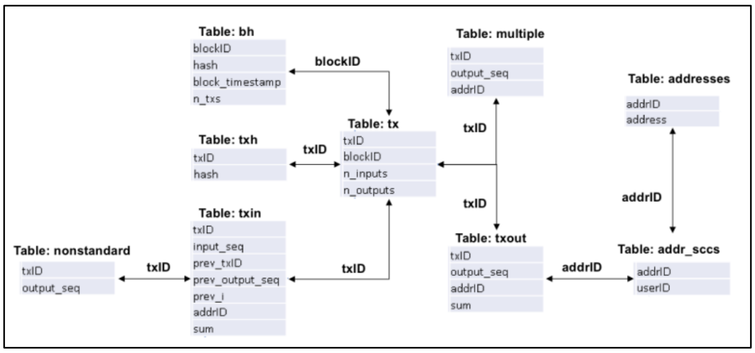


Figure 5.1. Database schema.

We assign a risk factor, binary in nature, on this dataset. The transactions related to any historical Bitcoin heist are assigned a risk level of 1 whereas all other transactions are assigned a risk factor of 0. Finally, a sampled set of transactions are selected from the complete and consolidated list of Bitcoin transactions and were considered for analysis and model development. The sampling is designed in such a way that all the historical hacks are selected in the sample and the non-hacks are selected randomly from the transactions without any hacking history. This sampling design results in a dataset with a total of 8,109 data points for analysis with 2,172 transactions with risk level 1 and 5,937 transactions with risk level 0.

6 Machine Learning Approach

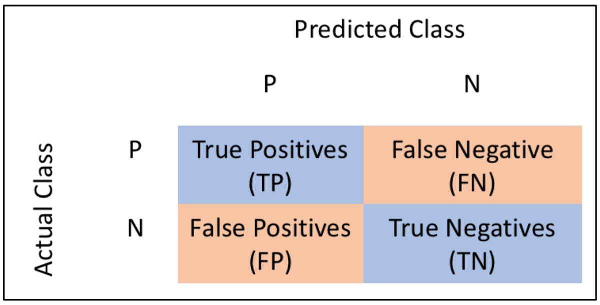
This section describes several different machine learning approaches that are implemented in order to classify whether a transaction is a hack or not. Three classification algorithms are trained and tested, as follows: Random Forest, *K*-Nearest Neighbor and Support-Vector Machine. The data are split two different ways. We follow two different splitting criteria: 80% train, 20% test and 70% train, 30% test.

The data are split using a Stratified ShuffleSplit cross-validator from Python's scikit-learn library. This method returns stratified randomized folds. The number of folds we cho~~o~~se is 10. These folds are made by preserving the percentage of samples for each class. Then, baseline models are created with all 10 variables included. After the baseline models are created, feature engineering is performed in order to fine-tune the models.

6.1 Model Evaluation

An essential binary classification model evaluation measure is a confusion matrix. In order to evaluate each model, certain metrics are calculated. The confusion matrix is an *N*-by-*N* matrix, where *N* is the number of classes being classified [13]. The columns and rows of the matrix list the number of instances as actual class versus predicted class, shown in Figure 6.1. The overall accuracy can be calculated as,

$$\frac{TP + TN}{FP + FN + TP + TN} = 1 - Error\,. \tag{2}$$



		Predicted Class	
		P	N
Actual Class	P	True Positives (TP)	False Negative (FN)
	N	False Positives (FP)	True Negatives (TN)

Figure 6.1. Confusion Matrix.

True positives are data points labeled as positive that are actually positive whereas false positives are data points labeled as positive that are actually negative. True negatives are data points labeled as negative that are actually negative whereas false negatives are data points labeled as negative that are actually positive.

Metrics such as precision, recall, F1-score and support are also used to evaluate how well a model preformed. Precision is the proportion of data points the model assigns as true positives (negatives) actually true positives (negatives). It can be calculated as:

$$\frac{TP}{TP+FP}\ \left(\frac{TN}{TN+FN}\right). \quad (3)$$

Recall is the model's ability to find every data point of interest. It can be calculated as:

$$\frac{TP}{TP+FN}\ . \quad (4)$$

It is important to note that when recall is increased, precision is decreased, and vice versa. When combining precision and recall the result is an F1 score. An F1 score is the harmonic mean of precision and recall which can be given by:

$$2*\frac{p*r}{p+r}\ , \quad (5)$$

where *p* is precision and *r* is recall. The harmonic mean is used because it strictly deals with extreme values. To create a balanced classification model with the optimal balance of recall and precision, the F1 score should be utilized [13]. Support is the total number of occurrences of each class.

6.2 Random Forest

Random Forest is an ensemble method for classification or regression. ~~In this paper, we use this method for classification.~~ The Random Forest randomly selects *k* features from *m* total features, where *k* < *m*. Among the *k* features, it calculates the node *d* using the best split point. Then, it splits the node into child nodes using the best split. This is repeated until *l* number of nodes has been reached. Finally, the forest is built by repeating the previous steps *n* number of times to create *n* number of trees. The results are obtained by the modal value of categories obtained by individual trees.

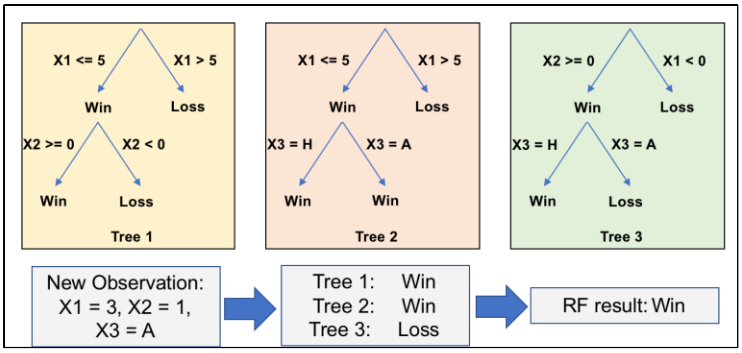


Figure 6.2. Random Forest tree.

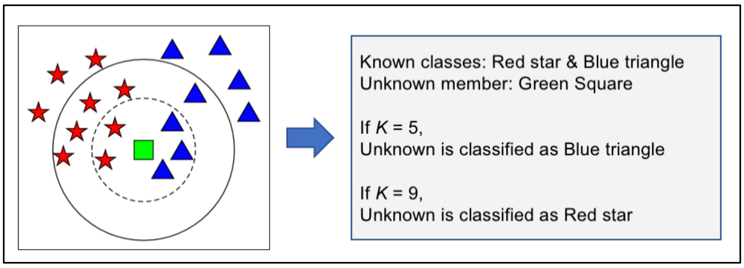
6.3 *K*-Nearest Neighbor

K-Nearest Neighbor (*K*NN) is a non-parametric method for classification or regression. ~~In this paper, we use this method for classification~~. In classification, the unknown is classified as the class with the majority of the members. Results of the unknown observation is determined by the average value of the *K* nearest neighbors of the unknown. *K* represents the number of data points considered for the classification.

The *K*NN method implements two search types: neighbor search and radius search. When given a set *x* of *n* points and a distance function, the search finds the *K* closest points in *x* to a single point or set of points *y*. The distance function used in this paper is the Minikowski distance,

$$d(x,y) = \left(\sum_{i=0}^{n-1} |x_i - y_i|^p\right)^{1/p}. \quad \textbf{(6)}$$

If *p* is equal to 1, (6) is equivalent to the Manhattan distance; if *p* is equal to 2, (6) is the Euclidean distance.



Known classes: Red star & Blue triangle
Unknown member: Green Square

If *K* = 5,
Unknown is classified as Blue triangle

If *K* = 9,
Unknown is classified as Red star

Figure 6.3. *K*NN classification.

6.4 Support-Vector Machine

Support vector machine (SVM) creates a hyperplane for high dimensional or set of hyperplanes for classification. SVM uses "one vs. one" approach for multiclass classification. In a 2-dimensional space, the hyperplane becomes a straight or curved line. The line with the maximum margin forms the support-vector. This approach trains

$$n \times (n - 1) \times 2 = b\ , \tag{7}$$

where n is the number of classifiers and b are the separate binary classifiers. Each binary classifier is trained using examples of two variables that are being classified. The test sample, *t,* is applied to each of the separate binary classifiers. Then, each binary classifier votes for the variable it is trying to classify. Finally, the test sample receives the label *c,* where *c* is the *y* variable receiving the highest number of votes.

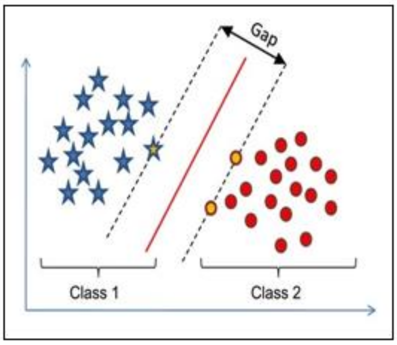


Figure 6.4. SVM classification.

7 Results

In this section, we summarize the results and visualization of our analyses. Accuracy of each model as well as a comparative study between models and split methods are presented. Model performance metrics are displayed to assess how well each method performs.

7.1 Random Forest Results

We train a baseline model on all 10 variables with an 80/20 split. Table 7.1 displays the Random Forest metrics for the baseline model. This model has a classification accuracy of 100%, as shown in Tables 7.1 and 7.2.

Table 7.1. Random Forest baseline metrics.

	Precision	Recall	F1-Sore	Support
0	1.00	1.00	1.00	1,193
1	1.00	1.00	1.00	429
Average	1.00	1.00	1.00	1,622
Accuracy		100.00%		

Table 7.2. Random Forest baseline confusion matrix.

	Predicted: 0	Predicted: 1
Actual: 0	1,193	0
Actual: 1	0	429

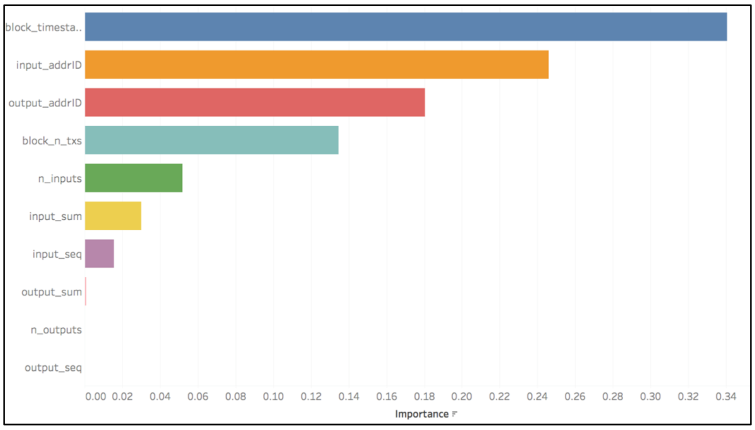


Figure 7.1. Random Forest baseline model feature importance.

Figure 7.1 displays the important features in the model: *block_timestamp*, *input_add ID, output_addrID, block_n_txs, n_inputs, input_sum, output_sum, n_outputs, output_seq* and *input_seq*. We then ~~decide to run~~ re-ran a reduced Random Forest model based on the top 8 features. Variables related to address ID's are dropped from the model as they are only used as identifiers. Table 7.3 displays the Random Forest metrics for the second model. This model has a classification accuracy of 100%, as shown in Tables 7.3 and 7.4.

Table 7.3. Random Forest second model metrics

	Precision	Recall	F1-Sore	Support
0	1.00	1.00	1.00	1,193
1	1.00	1.00	1.00	429
Average	1.00	1.00	1.00	1,622
Accuracy		100.00%		

Table 7.4. Random Forest second model confusion matrix.

	Predicted: 0	Predicted: 1
Actual: 0	1,193	0
Actual: 1	0	429

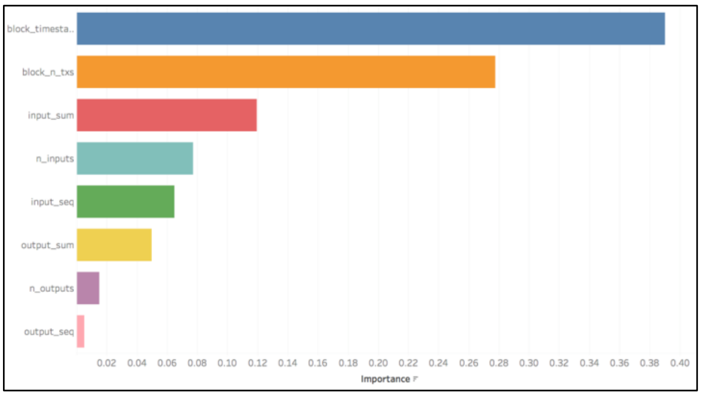


Figure 7.2. Random Forest second model feature importance

The important features (variables) are displayed in Figure 7.2 for the second model. The variables *block_timestamp, block_n_txs, n_inputs, input_sum, output_sum, n_outputs, output_seq* and *input_seq*. Since the model appears ~~continues~~ to be overfitted, more features are removed.

The third model is trained without *block_timestamp*. Table 7.5 displays the Random Forest metrics for the third model. This model has a classification accuracy of 99.81%. This model misclassifies only 3 out of 1,193 instances where no actual hack happened.

Table 7.5. Random Forest third model metrics.

	Precision	Recall	F1-Sore	Support
0	1.00	1.00	1.00	1,193
1	0.99	1.00	0.89	429
Average	1.00	1.00	1.00	1,622
Accuracy		99.81%		

Table 7.6. Random Forest third model confusion matrix.

	Predicted: 0	Predicted: 1
Actual: 0	1,190	3
Actual: 1	0	429

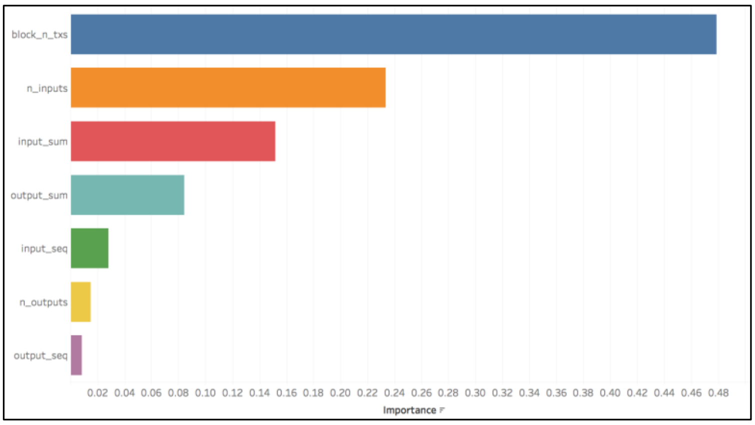


Figure 7.3. Random Forest third model feature importance.

The important features (variables) are displayed in Figure 7.3 for the third model. The variables *block_timestamp, block_n_txs, n_inputs, input_sum, output_sum, n_outputs, output_seq* and *input_seq*.

7.2 *K*-Nearest Neighbor Results

We trained a baseline model on all 10 variables with an 80/20 split. Table 7.7 displays the *K*NN metrics for the baseline model. This model has a classification accuracy of 99.96%. There are 8 non-hacks that were classified as a hack and 10 hacks that were classified as a non-hack, as shown in Table 7.8.

Table 7.7. *K*NN baseline model metrics.

	Precision	Recall	F1-Sore	Support
0	0.99	0.99	0.99	1,193
1	0.98	0.98	0.98	429
Average	0.99	0.99	0.99	1,622
Accuracy		99.96%		

Table 7.8. *K*NN baseline confusion matrix.

	Predicted: 0	Predicted: 1
Actual: 0	1,185	8
Actual: 1	10	419

To determine the optimal number of neighbors, we iterate through different values of *K* ranging from 1 to 50. Figure 7.4 displays the optimal number of neighbors. The optimal *K* is determined by the lowest mean squared error which happens to be 1 in this case.

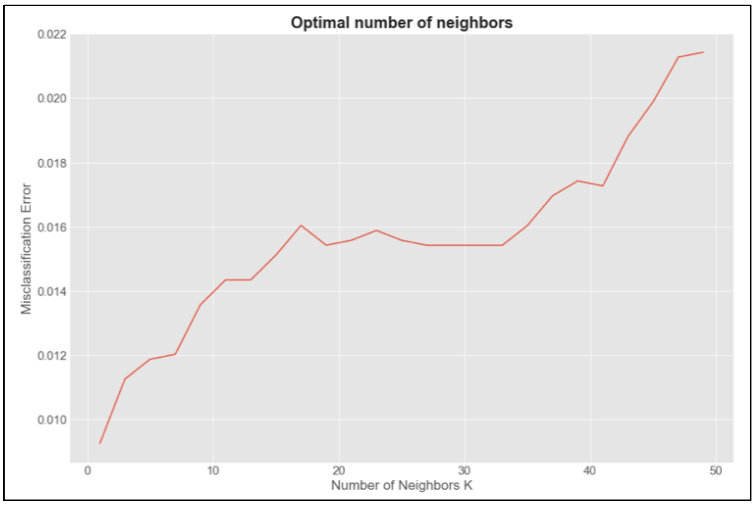


Figure 7.4. Optimal number of neighbors

Once the baseline is established, we perform feature engineering in order to explore how well the *K*NN model classifies with fewer features. The optimal number of neighbors will be used along with feature engineering. The second *K*NN model is trained with only 8 features: *block_timestamp, block_n_txs, n_inputs, input_sum, output_sum, n_outputs, output_seq* and *input_seq*.

Table 7.9. *K*NN second model metrics with optimal number of neighbors.

	Precision	Recall	F1-Sore	Support
0	0.99	0.99	0.99	1,193
1	0.98	0.98	0.98	429
Average	0.99	0.99	0.99	1,622
Accuracy		98.83%		

Table 7.10. *K*NN second model confusion matrix with optimal number of neighbors.

	Predicted: 0	Predicted: 1
Actual: 0	1,184	9
Actual: 1	10	419

Table 7.9 displays the *K*NN metrics for the second model with the optimal number of neighbors. This model has a classification accuracy of 98.83%. There are 9 non-hacks

that are classified as a hack and 10 hacks that are classified as a non-hack, as shown in Table 7.10.

The third model was trained with only two features: *block_n_txs* and *n_inputs*. Table 7.11 displays the *K*NN metrics for the third model with optimal number of neighbors. This model has a classification accuracy of 98.83%. There are two non-hacks that were classified as a hack and zero hacks that are classified as a non-hack, as shown in Table 7.12.

Table 7.11. *K*NN model #3 metrics with optimal number of neighbors.

	Precision	Recall	F1-Sore	Support
0	1.00	1.00	1.00	1,193
1	0.99	1.00	0.99	429
Average	1.00	1.00	1.00	1,622
Accuracy		99.88%		

Table 7.12. *K*NN third model confusion matrix with optimal number of neighbors.

	Predicted: 0	Predicted: 1
Actual: 0	1,190	2
Actual: 1	0	427

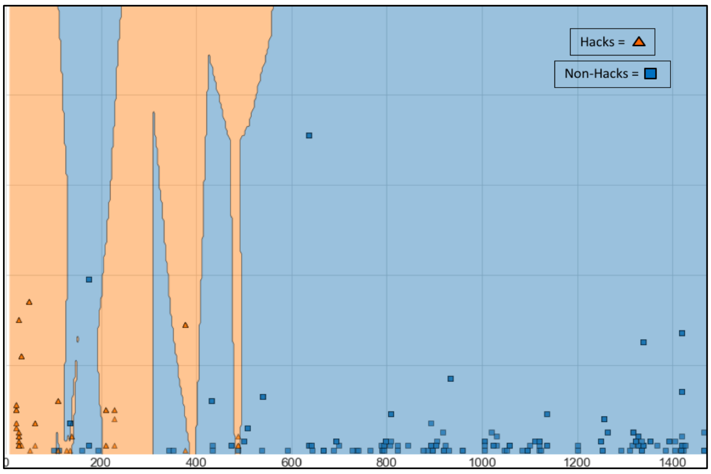


Figure 7.5. *K*NN third model classification results.

Figure 7.5 displays the results of the third model. The orange shaded areas are the hacks and the blue shaded areas are the non-hack. The orange triangles represent the hacks classified values whereas the blue squares represent the non-hack classified values.

7.3 Support-Vector Machine Results

We trained a baseline model on all 10 variables with an 80/20 split. Table 7.13 displays the SVM metrics for the baseline model. This model has a classification accuracy of 99.96%. There are 8 non-hacks that were classified as a hack and 10 hacks that were classified as a non-hack, as shown in Table 7.14.

Table 7.13. SVM baseline metrics.

	Precision	Recall	F1-Sore	Support
0	0.86	1.00	0.92	1,188
1	1.00	0.54	0.70	434
Average	0.89	0.88	0.86	1,622
Average Accuracy	97.76%			

Table 7.14. SVM baseline confusion matrix.

	Predicted: 0	Predicted: 1
Actual: 0	1,186	2
Actual: 1	0	434

Once the baseline is established, we perform feature engineering in order to explore how well the SVM model classifies with fewer features. A second SVM model is trained with only 8variables: *block_timestamp, block_n_txs, n_inputs, input_sum, output_sum, n_outputs, output_seq* and *input_seq*.

Table 7.15. SVM second model metrics.

	Precision	Recall	F1-Sore	Support
0	0.79	1.00	0.89	1,188
1	0.99	0.29	0.45	434
Average	0.85	0.81	0.77	1,622
Average Accuracy	82.66%			

Table 7.16. SVM second model confusion matrix.

	Predicted: 0	Predicted: 1
Actual: 0	1,187	1
Actual: 1	307	127

Table 7.15 displays the SVM metrics for the second model. This model has a classification accuracy of 82.66%. There is 1 non-hack that is classified as a hack and 307 hacks that are classified as a non-hack, as shown in Table 7.16.

A third SVM model is trained and tested with only *block_n_txs* and *n_inputs*. Table 7.17 displays the SVM metrics for the third model. This model has a classification accuracy of 85.96%. There are 189 non-hacks that are classified as a hack and 0 hacks that are classified as a non-hack, as shown in Table 7.18.

Table 7.17. SVM third model metrics.

	Precision	Recall	F1-Sore	Support
0	1.00	0.84	0.91	1,188
1	0.70	1.00	0.82	434
Average	0.92	0.88	0.89	1,622
Average Accuracy		85.96%		

Table 7.18. SVM third model confusion matrix.

	Predicted: 0	Predicted: 1
Actual: 0	999	189
Actual: 1	0	434

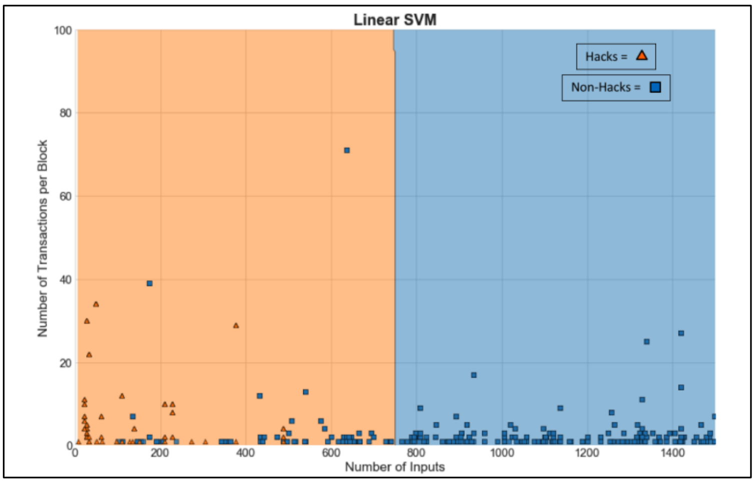


Figure 7.6. SVM third model classification results.

Figure 7.6 displays the results of the third SVM model. The orange shaded areas are the hacks and the blue shaded areas are the non-hacks. The orange triangles represent the hacks classified values whereas the blue squares represent the non-hack classified values.

7.4 A Comparison of all Models

Here, we perform a comparative study of the model performances. We built~~d~~ a Random Forest, *K*NN and SVM model~~s~~ with all 10 features and reduced models with the top 8 and 7 features, respectively. We consider an 80/20 split for all of the aforementioned models. The accuracies for all of these models are presented in Table 7.19 and Figure 7.7.

Commented [A8]: Maybe talk about which model is the ‘preferred’ or ‘chosen’ method for this type of analysis.

Table 7.19. Model comparison for 80/20 split.

Model	10 Features	8 Features	7 Features
Random Forest	100.00%	100.00%	99.81%
K-Nearest Neighbor	99.96%	98.83%	99.88%
Support-Vector Machine	97.76%	82.66%	85.96%

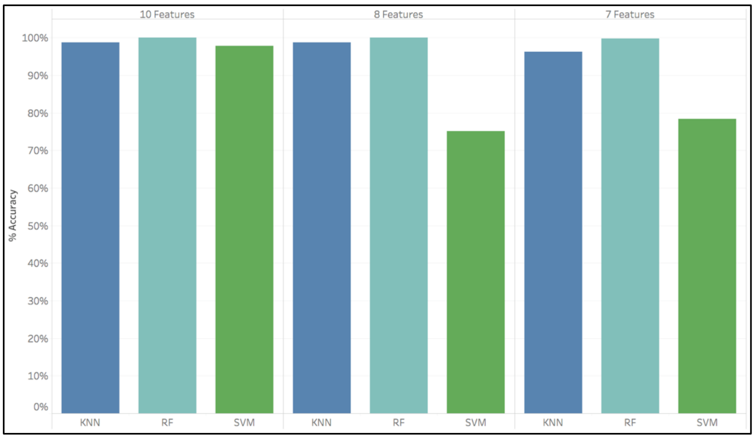


Figure 7.7. Comparison of the 80/20 split models.

8 Ethical Discussion

The Association for Computing Machinery's (ACM) code of ethics contains 24 imperatives by which all professionals should adhere [13]. Among them, "Avoid harm to others" and "Respect the privacy of others" are arguably the most relevant to our work here.

First and foremost, our risk scoring methodology has the potential to do harm to law-abiding cryptocurrency users. No machine learning model is 100% accurate. As a result, our machine learning models are capable of overstating a user's frequency of transactions with addresses associated with a hack or theft of a cryptocurrency exchange. This would make other users less likely to engage in transactions with such a user, despite the user's non-malicious intentions. Consequently, undue harm would be caused to that user by restricting their ability to do business in the marketplace.

Secondly, our approach arguably restricts the privacy of cryptocurrency users. By developing and publicizing a risk score for each cryptocurrency user, we are revealing information about their possible inclination to engage in unlawful activities. While the identity of each user is not explicitly disclosed, a single number that represents one's risk score, on the surface, says much about the kind of individuals and entities – whether good, bad or neutral – with which a user makes transactions. This privacy concern becomes especially salient if a user would prefer not to share the address ID's they have transacted with in the past. After all, it is the right of an individual to conceal most types of personal information from the general public.

The publicization of a risk score such as the one we propose in this paper generates additional questions that may have not come to light previously. For starters, would a cryptocurrency user have the option to "opt out" of their risk score being viewable by all other users? If so, how many users could realistically be expected to remove their user ID from the risk scoring list? If the vast majority of users opt out, that severely reduces the usefulness of the risk scoring concept. If users are not given opt-out ability, that raises the concern of the aforementioned privacy restriction. Moreover, who or what would be in charge of storing and managing our risk score database? If an independent third-party is responsible, would that undermine the de-centralized nature of the cryptocurrency marketplace? These are just some of the many questions that would need to be carefully answered before our risk scoring methodology could be implemented in the marketplace.

9 Conclusions

In our study, we have used key machine learning techniques to classify transactions based on propensity of risk involved in a transaction. We start with a baseline model with all relevant features followed by reduced models trained with key features selected from the baseline model. Both the Random Forest and *K*-Nearest Neighbors methods provide almost a perfect classification for the baseline model as well as the reduced models. ~~None of these models have misclassified a risky transaction as a non-risky one.~~

The method of Support Vector Machine also provides 98% accuracy in classifying transactions based on predicted risk levels. This method misclassified two of the non-risky transactions but classified all risky transactions correctly. The success level of SVM drops significantly to 83% when the top 8 influential features are considered. Moreover, this model fails to capture risky transactions in 70% of cases, 307 times out of 434 in total. The drop in accuracy prompted us to retrain SVM based on the top 2 important predictors – *block_n_txs* (number of transactions in a block) and *n_inputs* (number of inputs in a transaction). This model provides a better accuracy of 86% but the key improvement is in reducing 70% misclassification to no misclassification in case of risky transactions.

A successful classification leads to a successful and reliable risk scoring mechanism. After transactions are classified, all the address IDs are assigned a risk score based on their involvement in risky transactions from their full transaction basket to date. Address IDs with a high risk score are then flagged and blacklisted. The blacklist works as a guideline for making safe Bitcoin transactions.

The success of a financial deal depends largely on trust which can be ensured by the users on either side of a transaction. In a cryptocurrency marketplace where intermediaries are non-existent, a self-attestation mechanism in the form of a risk score offers a viable alternative to enhance trust. It empowers users with the ability to take careful consideration on others before transacting with them.

10 Future Work

In our present solution, we develop a risk scoring mechanism which is predominantly built on characteristics of transactions. We may extend this work to study the impact of additional factors such as whether a transaction is pooled or an exchange in our classification of transactions into risky or non-risky categories. A pooled transaction is of a newly minted coin that goes to the same input address. An exchange is an address in which a user can exchange some amount of Bitcoin to either another type of cryptocurrency or to fiat currency. These types of transactions can be identified by the output location. Typically, an exchange transaction will have an output to some third-party website. The website matbea.net is a popular source from which to collect this information.

Educating participants in cryptocurrency transactions on the advantages of this risk scoring approach is key for blockchain technology to become more trustworthy in the future. While adoption will undoubtedly be slow, pilot testing of our method on real-world cryptocurrency transactions will allow entities to check how a risk score empowers users with a self-guided trust in the crypto world. An advanced analysis to quantify the impact of this proposed risk scoring mechanism in cryptocurrency marketplaces would then reveal further possible improvements with a viable go-to-market solution for the industry.

References

1. Louis D. Brandeis, 1914, Other People’s Money and How the Bankers Use It. Frederick A. Stokes Company: New York. Originally published in Harper’s Weekly. Page 92. [Online]. Available: http://www.law.louisville.edu/library/collections/brandeis/node/196.
2. Satoshi Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System”. Published in 2008 on Bitcoin.org [Online]. Available: https://bitcoin.org/bitcoin.pdf
3. Statista, "Number of Blockchain Wallets 2018" [Online]. Available: https://www.statista.com/statistics/647374/worldwide-blockchain-wallet-users/
4. Coin Market Cap, "Cryptocurrency Market Capitalizations," [Online]. Available: https://coinmarketcap.com/A. Marshall “Combined Crypto Market Capitalization Races Past $800 Bln”Jan 07, 2018. [Online]. Available: https://cointelegraph.com/news/combined-crypto-market-capitalization-races-past-800-bln
5. Dr. G. Hileman, M. Rauchs University of Cambridge – Judge Business School “Global Cryptocurrency Benchmarking Study”. [Online]. Available: https://www.jbs.cam.ac.uk/fileadmin/user_upload/research/centres/alternative-finance/downloads/2017-global-cryptocurrency-benchmarking-study.pdf
6. Data.bitcoinity, "Bitcoin trading volume," [Online]. Available: https://data.bitcoinity.org/markets/volume/30d?c=e&t=b
7. Trustis, "Private Key Protection", 2010, [Online]. Available: http://www.trustis.com/pki/bat/guide/private-key-protection.htm#Contents
8. bitcoin, "Securing your Wallet", [Online]. Available: https://bitcoin.org/en/secure-your-wallet#update
9. A. Norry, "The History of the Mt Gox Hack: Bitcoin's Biggest Heist", [Online]. Available: https://blockonomi.com/mt-gox-hack/
10. Coindesk, "Coincheck Confirms Crypto Hack Loss Larger Than Mt.Gox," [Online]. Available: https://www.coindesk.com/coincheck-confirms-crypto-hack-loss-larger-than-mt-gox/
11. Bitcoin Forum. "List of Major Bitcoin Heists, Thefts, Hacks, Scams, and Losses," [Online]. Available: https://bitcointalk.orgindex.php?topic=83794 M. Felegyhazi, C. Kreibich, V. Paxson, “On the Potential of Proactive Domain Blacklisting”. Published on the International Computer Science Institute. [Online]. Available: https://www.usenix.org/legacy/event/leet10/tech/full_papers/Felegyhazi.pdf
12. D. Lary, "Supervised Classification", 2018, [Online].
13. ACM Council, “ACM Code of Ethics and Professional Conduct”, [Online]. Available: https://www.acm.org/about-acm/acm-code-of-ethics-and-professional-conduct