A Bayesian Approach to Identify Bitcoin Users

Péter L. Juhász[1,2] peter.laszlo.juhasz@ericsson.com

József Stéger[1] steger@complex.elte.hu

Dániel Kondor[1,3] dkondor@mit.edu

Gábor Vattay[1] vattay@complex.elte.hu

1 Department of Physics of Complex Systems, Eötvös Loránd University, Budapest, Hungary

2 Business Unit IT & Cloud Products, Ericsson Telecommunications Hungary, Budapest, Hungary

3 SENSEable City Laboratory, Massachusetts Institute of Technology, Cambridge MA, USA

Abstract

Bitcoin is a digital currency and electronic payment system operating over a peer-to-peer network on the Internet. One of its most important properties is the high level of anonymity it provides for its users. The users are identified by their Bitcoin addresses, which are random strings in the public records of transactions, the *blockchain*. When a user initiates a Bitcoin-transaction, his Bitcoin client program relays messages to other clients through the Bitcoin network. Monitoring the propagation of these messages and analyzing them carefully reveal hidden relations. In this paper, we develop a mathematical model using a probabilistic approach to link Bitcoin addresses and transactions to the originator IP address. To utilize our model, we carried out experiments by installing more than a hundred modified Bitcoin clients distributed in the network to observe as many messages as possible. During a two month observation period we were able to identify several thousand Bitcoin clients and bind their transactions to geographical locations.

1 Introduction

Bitcoin is the first widely used digital currency, developed by Satoshi Nakamoto after the beginning of the financial crisis in 2009 [1]. A distinctive feature of Bitcoin is that there is no central authority overseeing transactions, users are connected via a peer-to-peer network where they announce any transaction they wish to make. Transactions can then be validated by anyone using the publicly available list of transactions, the *blockchain*, which is in turn generated in a proof-of-work system. Cheating (e.g. including invalid transactions in the blockchain) thus would require one entity to control more than 50% of the computing power that users dedicate to generating the blockchain. In accordance with the decentralized nature of the system, the specifications of the network protocol is publicly available, while several open-source client programs implementing the protocol exist [2].

One of the key characteristics of Bitcoin is the high amount of anonymity it provides for its users [3]. Although one can learn the details of the transactions via the blockchain, it is still unknown who the users initiating those transactions are. This is possible since as there is no authority overseeing the functioning of the system, users do not need to provide any form of identification to join; anyone with an Internet connection can download a client program, which then allows them to generate any number of Bitcoin addresses that they can use in the transactions to send or receive Bitcoins. This results in that the identity of Bitcoin users is only revealed if they publish their Bitcoin address or this information is intercepted in some way outside the Bitcoin system. While anonymity is not among the main design goals of the Bitcoin system [3], Bitcoin is widely considered a highly anonymous way of performing financial transactions and is often utilized for illegal uncontrolled payments [4], along legal uses where the involved parties do not wish to disclose their identities to controlling entities in the traditional financial system, e.g. banks or governments.

In the paper, we present a probabilistic model based on the information propagating over the Bitcoin network, which gives the possibility of identifying the users initiating the transactions. In this case, identification means binding the transactions to the IP addresses where they were created.

The basic idea consists of two main steps. First, the probability is determined for each transaction that a specific client (identified by its IP address) created it. Assuming that the creator of the transaction controls the Bitcoin addresses from which money is sent in it, this step then results in possible IP address – Bitcoin address pairings. Next the most likely Bitcoin address – client pairings are identified by combining the probabilities in the list of pairings compiled in the previous step. This is further elaborated by grouping Bitcoin addresses that belong to the same user with high probability based on the transaction network. Finally, the geographical localization of the IP addresses opens the door for a large scale analysis of the distribution and flow of Bitcoin.

The following sections of the paper are structured as follows. Section 2 discusses the relevant characteristics of Bitcoin and provides the necessary background for the further sections. In section 3, the mathematical model used for the deanonymization is explained. The data collection is described in section 4. Section 5 presents the results of the application of the model. Finally the method described in this study is compared to the related works of the topic in section 6.

2 The Main Characteristics of Bitcoin

In order to use Bitcoin one has to connect to the Bitcoin network using an open-source *client program* [2]. In this work, we concentrate on the Bitcoin Core client [2], whose source code we inspected and modified for the purpose of data collection. By default, this client establishes eight connections to other clients. If there is a link between two clients, they are *connected*. Clients exchange information of different types, e.g. the transactions they know about, their state, cryptographic signatures and others through the network. This is necessary for the validation of the transactions as it is done by the entire network.

In case of Bitcoin transactions, *Bitcoin addresses* play similar role as the bank account numbers in regular currency transactions. However, there are two major differences:

- each user may have as many Bitcoin addresses as they would like to
- and multiple source and destination Bitcoin addresses can be involved in a single transaction.

In case of the Bitcoin Core client program, when a user initiates a transaction, the client program (the *originator*) immediately broadcasts this fact to the connected clients with 1/4 probability. With a probability of 3/4, the originator sends the message to a randomly chosen connected client in every 100 ms time interval until all connected clients are informed. This method is referred to as *trickling*, and its goal is to hide the source of the transaction. We expect that other types of clients employ similar mechanisms to protect the privacy of the users.

The connected clients use the same algorithm to relay the message further, until finally it spreads throughout the entire network. As a consequence an arbitrarily chosen client is not necessarily directly informed about the transaction by the originator. (Figure 1)

As no state, bank, institute or organization controls or ensures the validity of Bitcoin transactions, cryptographic methods are used by the whole Bitcoin community for this purpose. The security of Bitcoin is based on the blockchain. In this study the source Bitcoin addresses, the destination Bitcoin addresses, the timestamps and the transferred volume of Bitcoin is extracted from the blockchain for each transaction.

If the owners of the Bitcoin addresses were known, the blockchain would reveal all of the transactions of each Bitcoin user. As the anonymity is among the most important requirements of a currency, the inventors of Bitcoin did their best to preserve it.

3 A Bayesian Method for the Identification of Bitcoin Users

In this section probabilities are assigned to the distinct IP address and user pairings, which consists of three

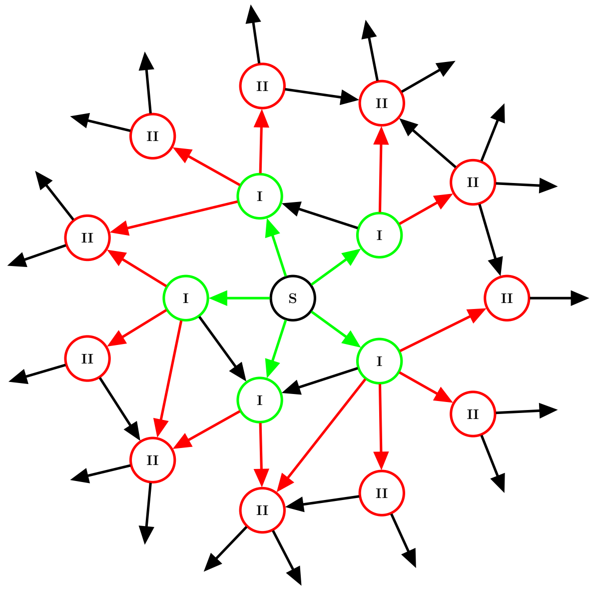


Figure 1: A new transaction is initiated by the client "S". At first it informs the clients denoted by "I" (they are informed directly from the originator). Then, these clients relay the transaction further – among possibly other I type clients – to the ones denoted by "II".

main steps.

Let us review the main steps needed to obtain the result. The process is illustrated in Figure 2.

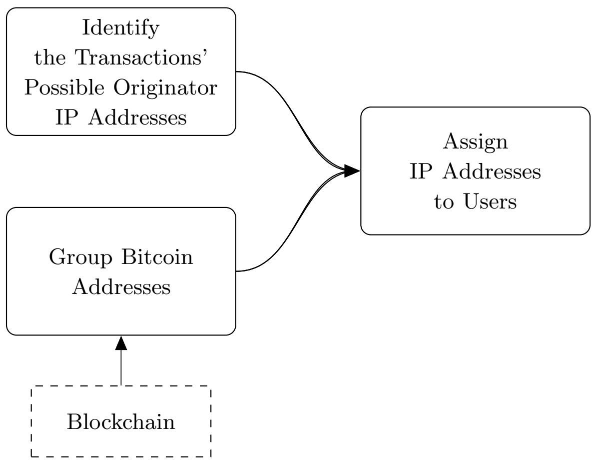


Figure 2: Main Steps

First, the propagating messages are observed and recorded by several *monitoring clients* in order to cover as great part of the network as possible. We have chosen those clients from the senders, which relay information about a specific transaction in the first time segment. They are the possible originators of the transaction. After some theoretical considerations, probabilities are assigned to the clients that show the probability of the clients being the originator.

Next, the blockchain is used to group the Bitcoin addresses owned by the same user. blockchain also enables to calculate the balances of the users for further analysis.

Last, by having possibly several transactions of the same Bitcoin address and the grouping of Bitcoin addresses by user allows us to combine measurements from multiple transactions to identify users with higher confidence. By combining the probabilities from the first step, the users (and their balances) are paired with the clients that are most likely the originators of their transactions. The clients can be geographically localized through their IP addresses, which allows the determination of the geographical distribution and flow of Bitcoin.

Step 1: Individual Probabilities

Let us consider a single transaction observed by one *monitoring client.*

A monitoring client connected to the originator does not necessarily receive the message from the originator first, because in some cases it can be relayed faster through a mediator client (Figure 3).

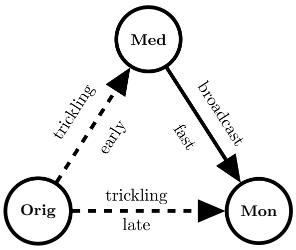


Figure 3: The message can be routed faster from the originator (Orig) to the monitoring client (Mon) through a mediator (Med) in the shown scenario.

The question is how long the monitoring client has to wait after receiving the message first until receiving it directly from the originator. If the delay of the network is negligible, then the traversal of the broadcast messages is also negligible. With this approximation, in the worst case the monitoring client can receive the message through a mediator immediately, right after the transaction is created. Clients are trickled every 100 ms, so if c clients are connected to the originator one of them has to wait $c/10$ seconds from first hearing the message until it surely receives it from the originator. We call this time interval the *first time segment* of the transaction and denote it by t_1.

If the originator has no more than 20 connections, it informs every connected client in at most two seconds from the monitoring client first receiving the message, i.e. $t_1 = 2$s. Assuming that the network is random[1], and the total number of active clients is approximately 50 000, the vast majority (more than 92%) of clients have less than 20 connections. Considering the neglected propagation delay of the network as well, which causes more latency for indirect propagation, we conclude that even in the case of a fast indirect transmission (i.e. the case illustrated in Figure 3), the client is very likely to receive the message from the originator at most $t_1 = 2$s after first receiving the message from a relay. See Appendix A for more detailed calculations. We then proceed with the assumption that connected clients that do not belong to this first time segment are not the creators of the transaction.

The previous reasoning was assuming that the monitoring client is connected to the originator, but in reality this is not known.

From the perspective of a monitoring client, the other Bitcoin clients can be classified to sets based on each transaction according to Figure 4. Some of the Bitcoin clients relay the message to it in the first time segment. This constitutes a subset of the Bitcoin clients to which the monitoring client is connected to at the time of the transaction. Only active Bitcoin clients are connected to the network, but not all of the clients are working at the examined moment.

Before the transaction, no information is known, thus the best estimate we can make is that each Bitcoin client has equal probability of being the originator of the transaction, resulting in a uniform probability distribution among the active clients (left side of Figure 4). After the transaction, each Bitcoin client in the first time segment can be either the real originator of the transaction or a client relaying it (via several hops). Furthermore, the real originator can also be among the rest of the network, not connected to our monitoring client. On the other hand, based on the previous arguments, we presume that clients that did not relay the transaction in the first time segment are certainly not the originators of the transaction. Thus, the probability of the first time segment clients increases while the connected clients not belonging to the first time segment will have zero probability (right side of Figure 4). Still nothing is known about the clients not connected to the monitoring client, therefore their probabilities will not change. Also, clients belonging to the same subsets can not be distinguished.

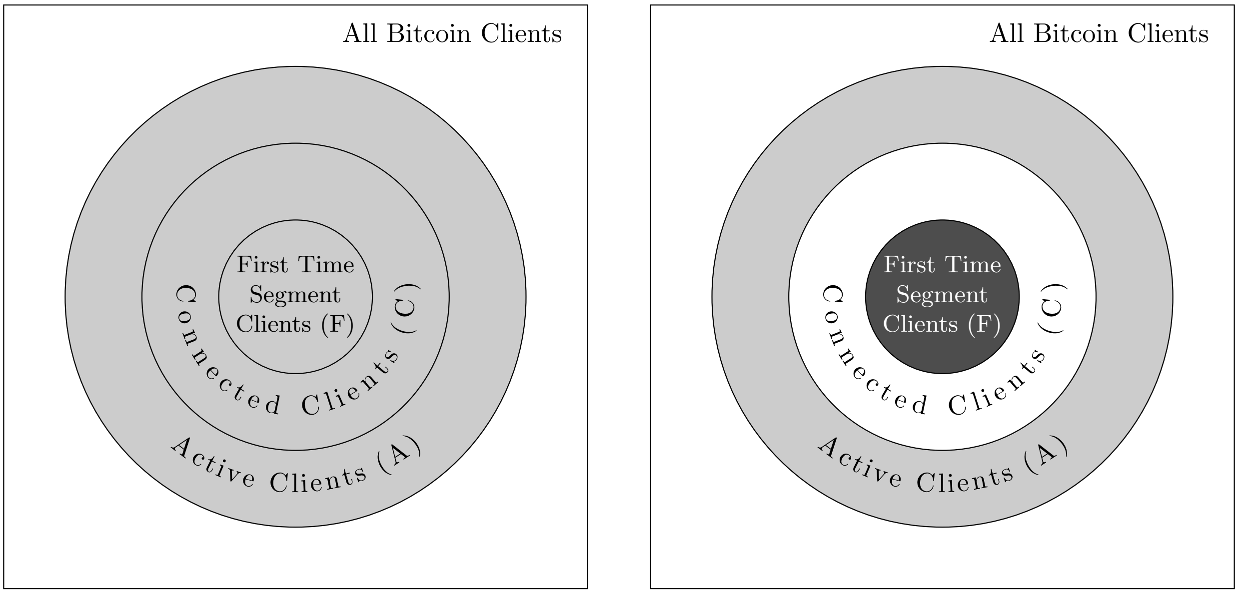
Let us calculate the probabilities of being the originator for clients in each set. The Roman font type notations of Figure 4 are used for the sets. The number of elements in the sets is denoted by $|\cdot|$. Furthermore, calligraphic notations are used for the following events: $\mathcal{C}$ denotes that the monitoring client is connected to the originator of the transaction, $\mathcal{O}$ denotes that the originator relays the message in the first time segment to the monitoring client and $\mathcal{F}$ means that a randomly chosen client from the first time segment is actually the originator of the transaction.

$$\mathbb{P}(\mathcal{C}) = \frac{|C|}{|A|}$$

as inactive clients can not be the originator of the transaction. If the monitoring client is connected to the originator, then it is going to inform the monitoring client in the first time segment. At this time all of the first time segment clients have the same probability of being the originator.

$$\mathbb{P}(\mathcal{O}|\mathcal{C}) = 1 \qquad \mathbb{P}(\mathcal{F}|\mathcal{C}) = \frac{1}{|F|}$$

[1]This is an approximation as there are clients that do not allow incoming connections. These clients have only 8 outgoing connections established when they entered the network.



(a) Before Transaction

(b) After Transaction

Figure 4: Relation of the different sets of clients. The darker the subset is, the higher the probability of a client in that subset is the originator of the transaction.

Let us apply the law of total probability for $\mathbb{P}(\mathcal{F})$.

$$\mathbb{P}(\mathcal{F}) = \mathbb{P}(\mathcal{F}|\mathcal{C}) \cdot \mathbb{P}(\mathcal{C}) + \mathbb{P}\left(\mathcal{F}|\overline{\mathcal{C}}\right) \cdot \mathbb{P}\left(\overline{\mathcal{C}}\right) = \frac{|C|}{|A|\,|F|}$$

where it was exploited that a client can not send any messages in the first time segment if it is not at all connected to the monitoring client: $\mathbb{P}\left(\mathcal{F}|\overline{\mathcal{C}}\right) = 0$.

The above formula gives the probability assigned to the first time segment clients. The connected clients not belonging to the first time segment have zero probability. The rest of the active clients has the same $1/|A|$ probability. We note that these probabilities still sum up to 1.

So far only one monitoring client was considered. If there are more monitoring clients the above mentioned sets are defined separately for each of them, and then the union of the corresponding sets is determined, i.e. $F(tx_i) = \cup_{j=1}^{N} F_j(tx_i)$ and $C(tx_i) = \cup_{j=1}^{N} C_j(tx_i)$ for N monitoring clients, where the subscripts denote the corresponding sets as observed for transaction tx_i by the jth monitoring client. Using this method, monitoring Bitcoin clients do not need to be synchronized in time. If time synchronization among monitoring clients was achieved, we could further limit the F set of first time segment clients to those that broadcast the transaction in t_1 time after *any* of our monitoring clients first received the transaction. In our experiments, achieving reliable time synchronization was not possible, so the union of sets was used as described. We note that the set of active clients at a given time (A) is not straightforward to estimate even with a large number of monitoring clients. To do that, we would need to perform an active network discovery over the peer-to-peer network of Bitcoin clients. Instead of implementing this functionality ourselves, we relied on the Bitnodes.io database [13], which provides the estimated number of active Bitcoin clients as a function of time (i.e. $|A|$). The actual set is not required for the calculations, only the size of the set at the time of the transactions is considered.

Step 2: Grouping the Transactions Belonging to the Same User

The next task is to group the Bitcoin addresses according to the users they are owned by. After this, every transaction can be assigned to the users by looking at the source Bitcoin addresses of the transaction.

It is known, that Bitcoin addresses appearing on the input side of the same transaction belong to the same user [6, 7, 8, 9]. This can be used for grouping individual Bitcoin addresses. The process is demonstrated in Figure 5. The left side of the figure shows the transactions and the input Bitcoin addresses where the Bitcoins are sent from. These Bitcoin addresses belong to the same user. When a Bitcoin address appears in different transactions (marked red and bold), all Bitcoin addresses can be merged and assigned to the same user.

The transactions belong to the user that owns its input Bitcoin address(es).

Step 3: Combining Probabilities – Naive Bayes Classification

From the message propagation it can be determined how likely the clients are the originators of the transactions. So far we considered the transactions independently from each other.

According to our assumptions, the transactions belonging to a single user were created by a few originator clients. This means that these transactions provide probabilities for the same set of originator clients. The originator clients can be identified more efficiently by

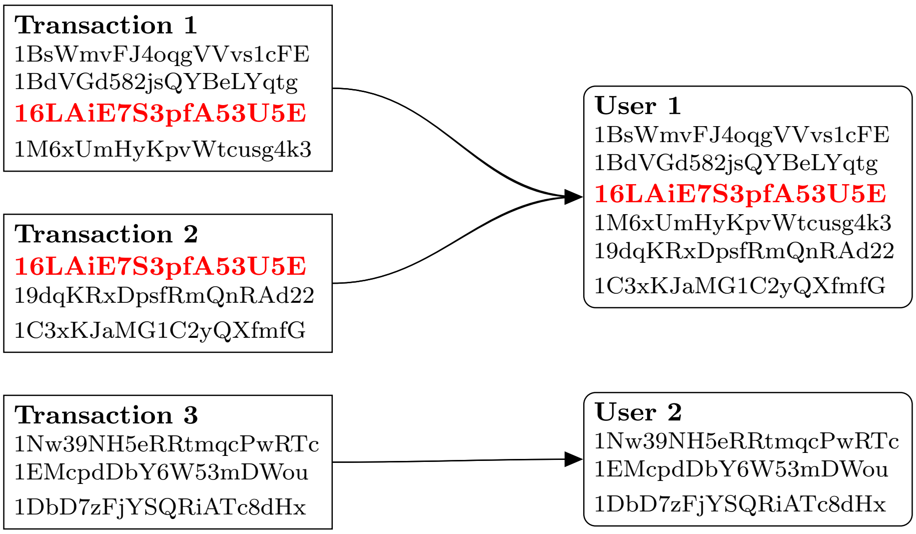


Figure 5: Grouping of Bitcoin addresses: the left side shows three transactions and the input Bitcoin addresses of these transactions, while the right side indicates how these Bitcoin addresses are grouped.

combining the probabilities belonging to these transactions, thus obtaining a more decisive result. This can be calculated by the naive Bayes classifier method [5]. Table 1 shows the transactions (denoted by tx) created by a single user. The transactions assign probabilities to the clients (IP addresses), which indicates the likelihood that the client is the originator of the transaction.

Table 1: The transactions of a single user (tx) assign probabilities to the clients (IP addresses), which shows the likelihood that the client is the originator of the transaction. $\mathbb{P}(\mathrm{IP}_i|\mathrm{tx}_j)$ denotes the probability that IP_i address created the tx_k transaction

	IP_1	$\cdots$	IP_i	$\cdots$	IP_n
tx_1	$\mathbb{P}(\mathrm{IP}_1|\mathrm{tx}_1)$		$\mathbb{P}(\mathrm{IP}_i|\mathrm{tx}_1)$		$\mathbb{P}(\mathrm{IP}_n|\mathrm{tx}_1)$
tx_2	$\mathbb{P}(\mathrm{IP}_1|\mathrm{tx}_2)$		$\mathbb{P}(\mathrm{IP}_i|\mathrm{tx}_2)$		$\mathbb{P}(\mathrm{IP}_n|\mathrm{tx}_2)$
$\cdots$					
tx_j	$\mathbb{P}(\mathrm{IP}_1|\mathrm{tx}_j)$		$\mathbb{P}(\mathrm{IP}_i|\mathrm{tx}_j)$		$\mathbb{P}(\mathrm{IP}_n|\mathrm{tx}_j)$
$\cdots$					
tx_m	$\mathbb{P}(\mathrm{IP}_1|\mathrm{tx}_m)$		$\mathbb{P}(\mathrm{IP}_i|\mathrm{tx}_m)$		$\mathbb{P}(\mathrm{IP}_n|\mathrm{tx}_m)$

If the ratio of the connected clients is small, the individual probabilities in the table are also low. The probabilities of an IP address related to the different transactions can be combined by the naive Bayes classification, resulting a row of combined probabilities. This shows how likely the IP addresses belong to the examined user.

The IP addresses will be divided into two classes, to the "originator" and the "non-originator" classes. For each transaction, there can be at most one IP address in the originator class. On the other hand, as a user can use multiple IP addresses to create Bitcoin transactions, after combining multiple transactions, more than one IP address can be in the originator class in the final result.

It is assumed that the Bitcoin users can be identified by a limited number of IP addresses they use when connected to the Bitcoin network. This involves that the users do not use TOR ("The Onion Router"), proxy servers or other similar systems hiding their IP addresses. If this does not hold, the probabilities would be distributed among the IP addresses. Note, that the invalidity of this assumption does not result in false IP address, user pairings: only those users will be identified whom the assumption holds for.

The naive Bayes classification can only be applied if the transactions provide conditionally independent probabilities. Otherwise the dependencies between the transactions should be determined [10].

By the application of the naive Bayes classifier (see Appendix B for the detailed derivation), the combined probability of an IP address (IP_i) belonging to the C_o originator class is given by

$$\mathbb{P}(\mathrm{IP}_i \in C_o|\mathbf{tx}) = \frac{1}{1+\exp\left[(1-m)\ln\left(\overline{|A|}-1\right)+\sum_{k=1}^{m}\ln\left(\frac{1}{\mathbb{P}(\mathrm{IP}_i\in C_o|\mathrm{tx}_k)}-1\right)\right]}$$

where $\mathbf{tx}$ denotes the vector of all considered transactions, $\overline{|A|}$ is the average of the total number of active clients through the transactions[2] and m is the number of transactions.

[2]The $|A|$ number of active clients varies through the transactions as they occur in different times. Thus, the $\overline{|A|}$ average of the different $|A|$ values is used as it is suggested in [11].

4 Data Collection

During the data collection campaign, modified Bitcoin clients were connected to the network. As the program code is open-source, it was straightforward to implement a monitoring client.

The tx type messages are part of the Bitcoin communication protocol and they contain information regarding a transaction. These messages identified by a 128-bit hash code are observed and recorded.

The Bitcoin addresses, the amount of Bitcoin sent and other information of interest can be looked up in the blockchain according to the hash code. When receiving a tx message, the monitoring client recorded the IP address of the sender client, the time of reception and the transaction's hash identifier.

In order to monitor as large part of the Bitcoin network as possible, the modified Bitcoin clients were installed simultaneously to 169 computers located at different parts of the world, and all of these were recording the observed traffic during the campaign. Bitcoin clients behind firewalls usually do not allow incoming connections, i.e. our monitoring clients can not establish connections to them. By using a large number of monitoring clients it is more likely, that the Bitcoin clients behind firewalls initiate connections to some of our monitoring clients when they enter the network. The monitoring clients were installed on computers that are part of PlanetLab, a system maintained for network communication research. [12]

The data collection campaign took slightly more than two months between 10/14/2013 and 12/20/2013. During this period 300 million records were obtained, in which 4 155 387 transactions and 124 498 IP-addresses were identified. The collected data was imported into an SQL database server.

To calculate the probabilities described above, the total number of active clients need to be determined. From the Bitnodes.io database [13] one can look up the number of active IP addresses of the Bitcoin clients as a function of time.

5 Results

When calculating the combined probability of each IP address belonging to the specific user, the question arises when should a pairing be *accepted*? As more than one IP address can be used by each user and one IP address can be used by several users, no restriction is made of this kind. A pairing is accepted, if its probability is higher than 0.5. This means that the IP address of interest has at least 0.5 probability of being used by the user.

As a result, 22 363 users could be identified, and altogether 1 797 IP addresses were assigned to them.

The imbalance is caused by three outstanding IP addresses to which 20 680 users are assigned. These IP addresses probably belong to Bitcoin wallet services, which can be used for creating transactions on a website without using a private computer. Note that the incomplete grouping of Bitcoin addresses can also result in an IP address being associated with several groups of Bitcoin addresses. These groups actually belong to the same user, but they could not be connected in the grouping algorithm.

For the remainder data, 1.07 IP addresses belong to one user on average. This is due to the fact that a user can use multiple IP addresses when connecting to the Bitcoin network. The maximum number of IP addresses identified as belonging to a single user is 11.

Calculating the Balances

Examining the blockchain data alone allows to investigate the time evolution of user balances before, during and after the data collection campaign.

Figure 6 shows the total balance of all identified users versus time. The time interval in which the data collection was taking place is marked by a shaded area. Before data collection, the amount of Bitcoin owned by the identified users is increasing. This is due to the fact that some of the identified Bitcoin addresses were created before the beginning of the measurement campaign. After the measurement, some of the identified Bitcoin addresses were not used anymore, and other new unidentified Bitcoin addresses took over their place. The steep drop during the measurement is probably due to the significant increase of exchange rate in this time interval, which inspired the users to sell their Bitcoins for traditional currencies. We found a significant, -0.91 linear correlation coefficient between the total amount of Bitcoin owned by the identified users and the exchange rate during the measurement period.

The total number of Bitcoins in use is constantly increasing as time goes by. At the time of the measurement $\sim$ 13 500 000 Bitcoins were in circulation. The amount of Bitcoins owned by the identified users reached a maximum of 432 666 on 10/25/2013, which corresponds to $\sim 3.2\%$ of the total amount of Bitcoins. We believe that this ratio is a statistically representative sample, if the data is collected with random sampling. However, systematic differences could have affected the data collection as users in different parts of the world, with different intentions and technical backgrounds were possibly operating differently in the network. The users could be protected by firewalls thus banning incoming connections, and they could also obscure their operation by using VPN, proxy service or TOR.

Geographical Distribution of Bitcoin and the Cash Flow

The location of IP addresses can be determined from publicly available databases such as MaxMind [14], which contains approximate locations of the IP addresses. If the Bitcoin users use additional tools to hide their IP addresses, or if the IP addresses are located at other positions than they are registered to, the database gives false location results. However, these inaccuracies are not relevant in the vast majority of the cases.

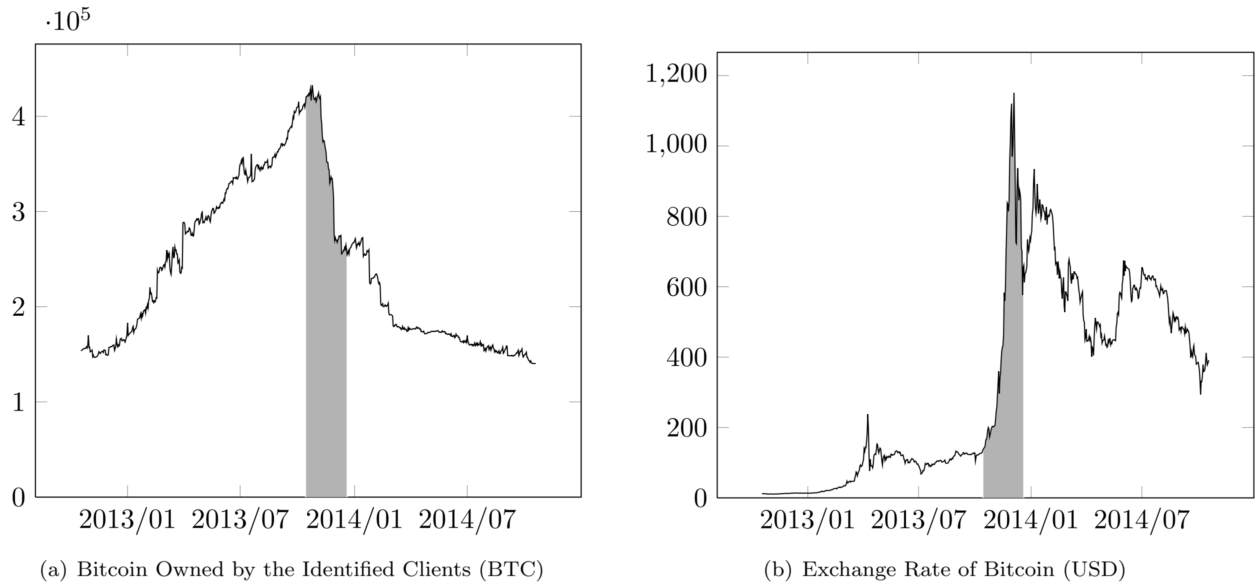


Figure 6: Balances of Bitcoin users identified in our study and the Bitcoin exchange rate. The shaded area corresponds to our data collection period.

Figure 7 shows the distribution of the identified Bitcoin clients. The coloring represents the logarithmic value of the density. The identified Bitcoin clients are mostly located on the more developed regions of the world. Note that in some countries, such as Russia or China, the Internet is regulated, therefore some interference of the connected clients (and their messages) can occur.

By the localization of the IP addresses, the geographical distribution of Bitcoin can also be determined (Figure 8). This figure only shows the distribution of the Bitcoins that are owned by the identified clients; the coloring is logarithmic. The snapshot belongs to the end of the data collection period, 12/20/2013.

The analysis detailed in Section 3 results in a data set of transactions and identified originators. It is worth to examine if some originator addresses can be mapped to receiver Bitcoin addresses as well. There are 44 561 transactions in which both sides could be found, and altogether 190 819 Bitcoins were transferred in the identified transactions. In these transactions 6 266 users appear as senders and 5 118 appear as receivers.

The transactions are visualized on a world map (Figure 9). The thickness, opacity and saturation of the arrows express the amount of Bitcoin transferred in the related transaction. The time course of the transactions is demonstrated on a video (*transactions_video.avi*) that can be found among the supplementary materials.

Let us have a look at the flow of Bitcoin between the different countries, which is illustrated in Figure 10. As the vast majority of the identified Bitcoin transactions belongs to a few countries, only the top ten most significant ones are shown in the figure. 87.5% of the Bitcoins in our dataset were transferred between these countries.

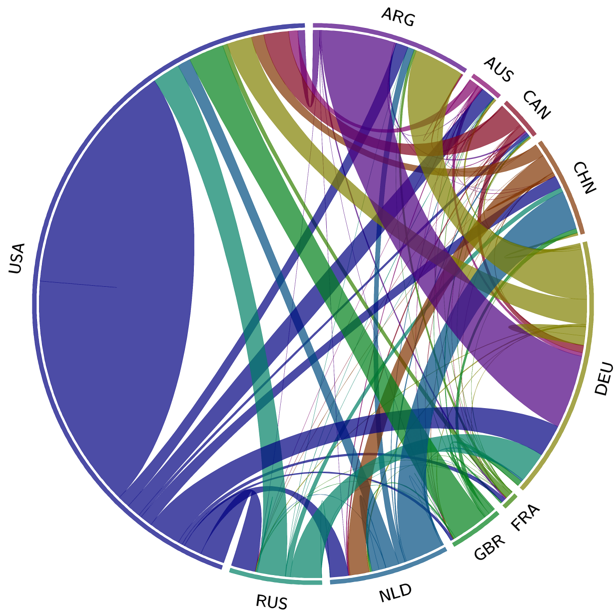


Figure 10: The Flow of Bitcoin Between Countries

The different countries are indicated by arcs on the perimeter of the figure. The colors of the links are identical with the color of the country where the Bitcoins were sent from. Most of the Bitcoins (more than 25.4% of the total amount) are transacted internally in the United States. There are several interesting connections: the second largest flow is between Germany and Argentina (25 508 Bitcoins, 13.4% of the total amount), and there is a significant Bitcoin flow between China and the Netherlands as well.

6 Related Work

There have been several works discussing the anonymity concerns of Bitcoin. All of them show that

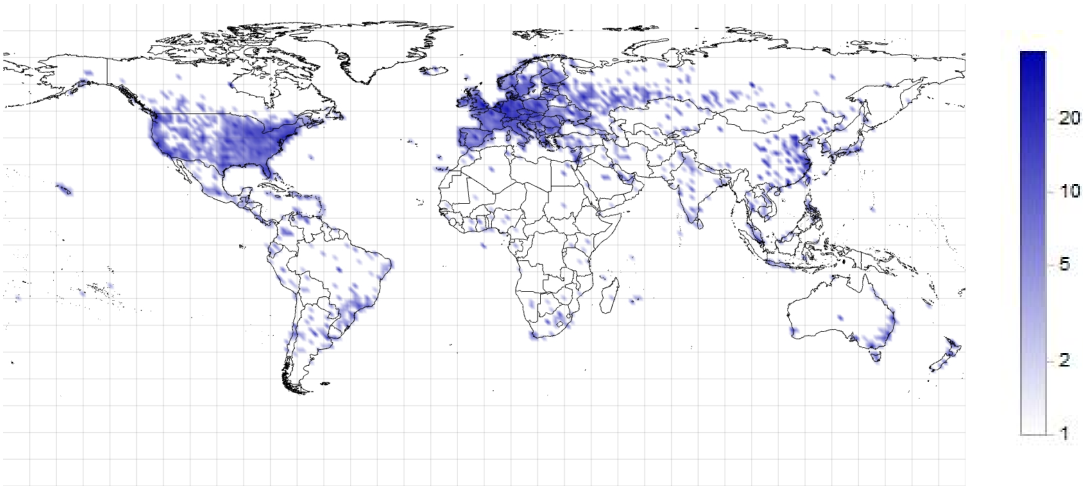


Figure 7: Distribution of the Identified Bitcoin Clients ($1/100\,\mathrm{km}^2$)

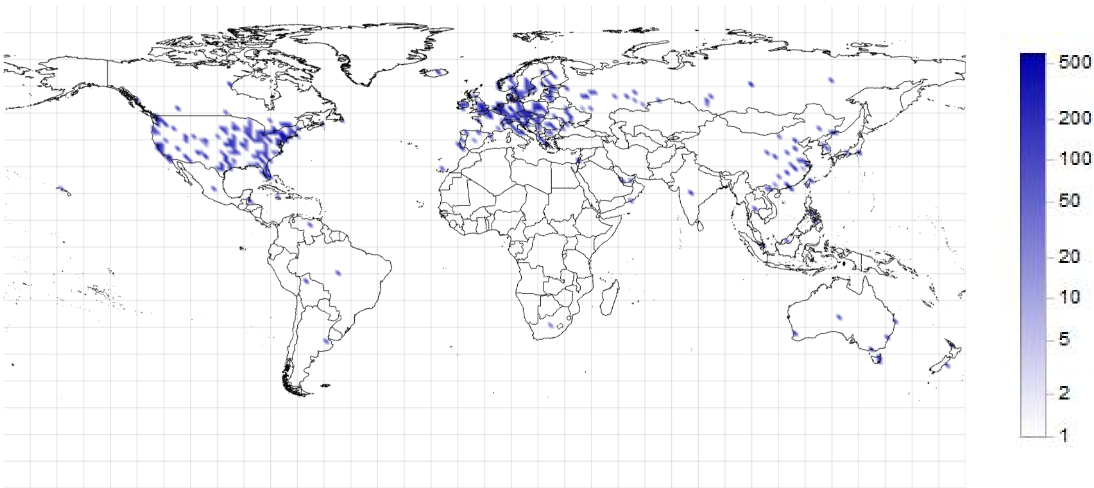


Figure 8: Distribution of Bitcoin Owned by the Identified Clients on 12/20/2013 ($1/100\,\mathrm{km}^2$)

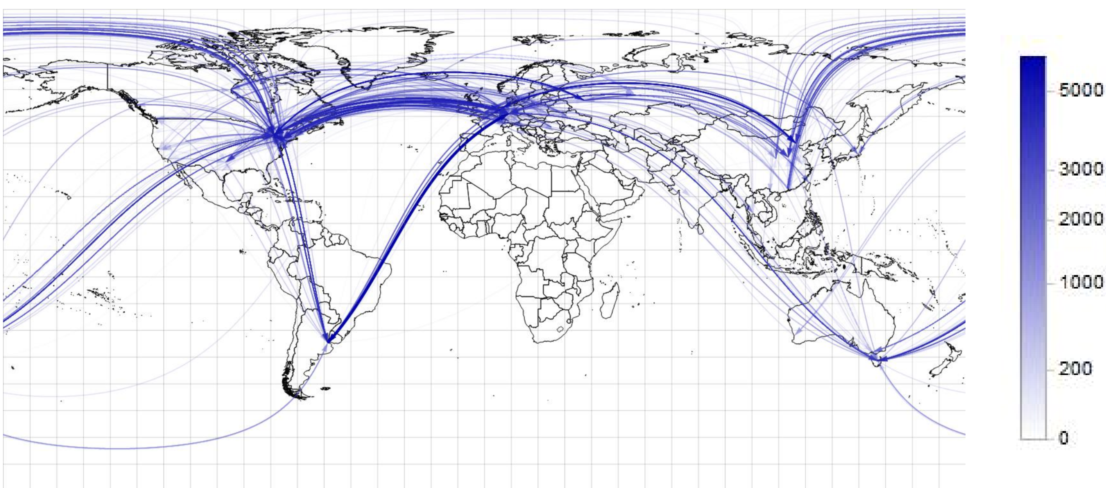


Figure 9: The Flow of Bitcoin

the statistical processing of a huge amount of seemingly insignificant information can take the attacker closer to reveal the identity of people using Bitcoin.

Elli Androulaki, et al. [7] evaluated the privacy of Bitcoin by analyzing the system using a simulator. After grouping Bitcoin addresses they used behavior-based clustering techniques (K-Means and Hierarchical Agglomerative Clustering algorithms) to bind the Bitcoin addresses to real users.

Fergal Reid and Martin Harrigan [8] used mainly offline data processing of the blockchain to analyze the transaction graph. They identified its clusters and components, and analyzed the degree distribution of the user network.

It was also shown that the analysis of publicly available data from social websites and forums can also reveal the Bitcoin addresses of some users. [8, 15]

A. Biryukov, D. Khovratovich and I. Pustogarov provided a method which connects the users to IP addresses [16]. They connected to all publicly available Bitcoin nodes (servers) and listened the messages they were relaying. Targeting clients that do not accept incoming connections, they established connections to as many servers (Bitcoin clients that accept incoming connections) as possible. They used Bitcoin's peer discovery mechanism to link transactions to their originators: the servers that broadcast the newly connected clients' IP addresses were the same set of servers which first relayed their transactions. The difficulty of this method is that a lot of connections have to be established to reach good results as the number of the servers increases. On the other hand it promises results for Bitcoin clients which monitoring nodes cannot directly connect to (e.g. because they are protected by firewalls and connect to a limited). In contrast, our methodology requires direct connections to the originators and we thrive to achieve this by running a lot of Bitcoin clients accepting a large number of incoming connections. While they use a fixed number of message relays to infer the local network of the originator, we use a short initial time span for message broadcasts to infer the actual originator. A further main difference in our methodology is combining information from many transactions and linking addresses based on the blockchain to provide more transactions per Bitcoin user in that step. Our probabilistic approach could be combined with the methodology in [16] in order to identify the "hidden" Bitcoin nodes with higher probability. This gives the possibility of linking Bitcoin address groups belonging to the same user as grouping the Bitcoin addresses was only partly achieved based on the transaction history in the blockchain.

Philip Koshy et al. also monitored the messages about the transactions and they classified the transactions to distinct relay patterns [9]. After applying heuristics to determine the possible owner IP addresses of the transaction, they computed simple aggregate statistics to filter out the correct Bitcoin address - IP address pairings for both in- and output addresses.

Basically the following common methods are used to reveal the identities of Bitcoin users:

1. analyzing transactions with multiple input and group the input Bitcoin addresses of the same transactions;
2. analysis of Bitcoin flow in the transaction graph, usage of clustering techniques;
3. analysis of propagating network-layer information to bind their content to the users,
4. and finally using publicly available information (e.g., in forums) to connect Bitcoin addresses to identities.

The methodology presented in our work belongs to the 1 and 3 types of approaches, mainly based on statistical processing of network propagation properties.

7 Conclusions

In this paper we examined the problem of user identification in the Bitcoin network. While Bitcoin provides a significant level of anonymity as Bitcoin addresses can be generated freely and without providing any form of personal identification, the requirement to announce new transactions on the peer to peer network opens up the possibility of linking Bitcoin addresses to the IP addresses of clients. Our main goal was evaluating the feasibility of this procedure.

A modified Bitcoin client program was installed on over a hundred computers, which recorded the propagating messages on the network that announced new transactions. Based on the information propagation properties of these messages, we developed a mathematical model using naive Bayes classifier method to assign Bitcoin addresses to the clients that most probably control them. As a result Bitcoin address – IP address mappings were identified. Through the IP addresses of the clients we could determine their geographical location, which enabled the spatial analysis of distribution and flow of Bitcoin.

The method is cheap in terms of resources, the used algorithms are relatively easy to implement and can be combined with other Bitcoin-transaction related information.

All monitoring clients behaved as regular Bitcoin clients during the measurement. Although they did not generate any transactions, the source code can be modified to do so if a better concealment is required. Furthermore, the monitoring clients do not need to be connected to other Bitcoin users in any detectable way, making it virtually impossible to reveal their monitoring activity. This raises the question if the Bitcoin network might already be monitored by a similar methodology. It can be implied that Bitcoin users should take further steps to adequately disguise their real IP addresses and preserve their anonymity.

Acknowledgements

This work has been accomplished at the Department of Physics of Complex Systems, Eötvös Loránd University.

We thank the PlanetLab community [12] that the servers of their network could be used for the data collection.

We are also grateful to MaxMind [14] for the database that could be used for the localization of the IP addresses.

Additional Information

Author contributions PJ analyzed the data, derived the statistical formulation and drafted the manuscript. JS performed data collection, helped in deriving the statistical formulation and helped writing the manuscript. DK provided software for data collection, helped with the data collection and helped write the manuscript. GV participated in designing the study. All authors have read the final manuscript and approved it for publication.

Data availability All data collected during the study is made publicly available.

Ethics statement All data used in the analysis is made publicly available by the Bitcoin users as it is required by the Bitcoin protocol. Collecting data on the level of network traffic allows linking Bitcoin addresses to the IP addresses of Bitcoin users. No other personally identifiable information beside IP address was collected about users, and no attempt was made to link IP addresses to actual people beside establishing coarse-grained geographic location. In the shared data, IP addresses were replaced with identifiers to prevent connecting the transactions with individuals based on other IP address related information.

Competing interests The authors declare that they have no competing interests.

References

[1] Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." (2008).

[2] Bitcoin Core, GitHub repository, "https://github.com/bitcoin/bitcoin" (2013).

[3] Bitcoin Wiki, "https://en.bitcoin.it/wiki/Anonymity".

[4] Christin, Nicolas. "Traveling the Silk Road: A measurement analysis of a large anonymous online marketplace." *Proceedings of the 22nd international conference on World Wide Web.* ACM, 2013.

[5] Lewis, David D. "Naive (Bayes) at forty: The independence assumption in information retrieval." *European conference on machine learning.* Springer Berlin Heidelberg, 1998.

[6] Ron, Dorit, and Adi Shamir. "Quantitative analysis of the full bitcoin transaction graph." *International Conference on Financial Cryptography and Data Security.* Springer Berlin Heidelberg, 2013.

[7] Androulaki, Elli, et al. "Evaluating user privacy in bitcoin." *International Conference on Financial Cryptography and Data Security.* Springer Berlin Heidelberg, 2013.

[8] Reid, Fergal, and Martin Harrigan. "An analysis of anonymity in the bitcoin system." *Security and privacy in social networks.* Springer New York, 2013. 197-223.

[9] Koshy, Philip, Diana Koshy, and Patrick McDaniel. "An analysis of anonymity in bitcoin using p2p network traffic." *International Conference on Financial Cryptography and Data Security.* Springer Berlin Heidelberg, 2014.

[10] Rish, Irina, Joseph Hellerstein, and Jayram Thathachar. "An analysis of data characteristics that affect naive Bayes performance." *IBM TJ Watson Research Center* 30 (2001).

[11] Figueiredo, Mário AT. "Lecture notes on Bayesian estimation and classification." *Instituto de Telecomunicacoes-Instituto Superior Tecnico* (2004): 60.

[12] Chun, Brent, et al. "Planetlab: an overlay testbed for broad-coverage services." *ACM SIGCOMM Computer Communication Review* 33.3 (2003): 3-12.

[13] Bitnodes.io, "https://getaddr.bitnodes.io"

[14] MaxMind, "http://www.maxmind.com/"

[15] Gross, Ralph, and Alessandro Acquisti. "Information revelation and privacy in online social networks." *Proceedings of the 2005 ACM workshop on Privacy in the electronic society.* ACM, 2005.

[16] Biryukov, Alex, Dmitry Khovratovich, and Ivan Pustogarov. "Deanonymisation of clients in Bitcoin P2P network." *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security.* ACM, 2014.

Appendix A Degree Distribution

Let us determine the degree distribution of the Bitcoin clients $f(c)$. The total number of active clients is considered to be constant N. The transient effects of the network formation are neglected as these effects cancel out with the constant connection and disconnection of the clients to the network.

The distribution of the incoming connections has to be determined. When connecting to the network, each client connects to $c_{\text{out}} = 8$ randomly chosen clients. The probability that a specific client establishes a connection to another one is $c_{\text{out}}/(N-1) \approx c_{\text{out}}/N$. The number of incoming connections has a binomial distribution, which can be approximated by a normal distribution:

$$
\begin{aligned}
\mathbb{P}(c_{\text{in}} = k) &= \binom{N}{k} \left(\frac{c_{\text{out}}}{N}\right)^k \left(1 - \frac{c_{\text{out}}}{N}\right)^{N-k} \approx \\
&\approx \frac{1}{\sqrt{2\pi N \frac{c_{\text{out}}}{N}\left(1 - \frac{c_{\text{out}}}{N}\right)}} \exp\left[-\frac{\left(k - N\frac{c_{\text{out}}}{N}\right)^2}{2N\frac{c_{\text{out}}}{N}\left(1 - \frac{c_{\text{out}}}{N}\right)}\right] \\
f_{c_{\text{in}}}(x) &= \frac{1}{\sqrt{2\pi c_{\text{out}}\left(1 - \frac{c_{\text{out}}}{N}\right)}} \exp\left[-\frac{(x - c_{\text{out}})^2}{2c_{\text{out}}\left(1 - \frac{c_{\text{out}}}{N}\right)}\right]
\end{aligned}
$$

where c_{in} is the number of incoming connections. This expression only describes the distribution of the incoming connections. The c_{out} number of outgoing connections has to be added to this formula to obtain the distribution of the total number of connections, so the density function must be shifted to the right by c_{out}: if $c \geq c_{\text{out}}$,

$$
f_c(x) = \frac{1}{\sqrt{2\pi c_{\text{out}}\left(1 - \frac{c_{\text{out}}}{N}\right)}} \exp\left[-\frac{(x - 2c_{\text{out}})^2}{2c_{\text{out}}\left(1 - \frac{c_{\text{out}}}{N}\right)}\right]
$$

This gives the total degree distribution of the clients, which is illustrated in Figure 11. If $c_{\text{out}} = 8$ and $N = 50\,000$, more than 92% of the clients has no more than 20 connections.

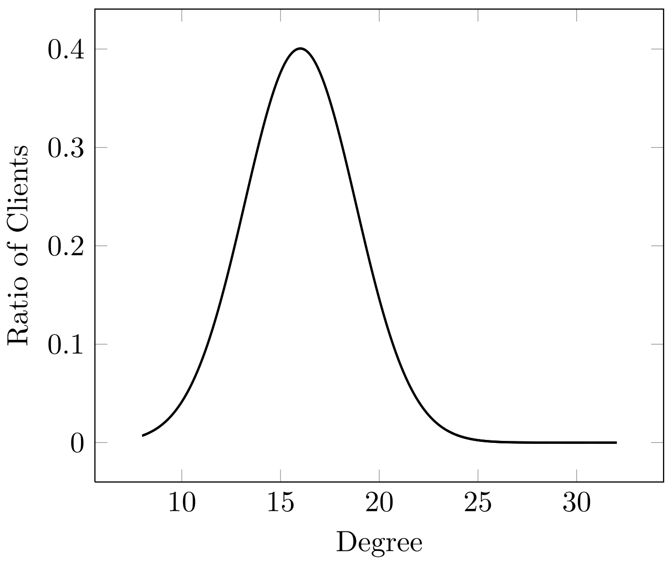


Figure 11: Degree Distribution of Bitcoin clients

Appendix B Derivation of Naive Bayes Classifier Method

The model classifies the clients into the *originator* and *non-originator* classes (C_o and C_n respectively) based on their IP addresses and by considering m transactions. Transactions are denoted by $\mathbf{tx} = \{\mathrm{tx}_i\}$, $(i \in [1;\, m])$.

Consider a single IP address, and let us examine the probabilities that the different transactions assign to it. Using the Bayes theorem the probability of belonging to the originator class is

$$\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathbf{tx}\right) = \frac{\mathbb{P}\left(\mathrm{IP}_i \in C_o\right)}{\mathbb{P}\left(\mathbf{tx}\right)}\,\mathbb{P}\left(\mathbf{tx}|\mathrm{IP}_i \in C_o\right)$$

where $\mathbb{P}\left(\mathrm{IP}_i \in C_o\right)$ is the frequency of C_o class (a priori probability). By assuming that the probabilities $\mathbb{P}\left(\mathbf{tx}|\mathrm{IP}_i \in C_o\right)$ are conditionally independent, the expression can be simplified.

$$\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathbf{tx}\right) = \frac{\mathbb{P}\left(\mathrm{IP}_i \in C_o\right)}{\mathbb{P}\left(\mathbf{tx}\right)}\prod_{i=1}^{m}\mathbb{P}\left(\mathrm{tx}_i|\mathrm{IP}_i \in C_o\right)$$

Bayes theorem can be applied again to the factors in the product.

$$\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathbf{tx}\right) = \frac{\mathbb{P}\left(\mathrm{IP}_i \in C_o\right)}{\mathbb{P}\left(\mathbf{tx}\right)}\prod_{i=1}^{m}\frac{\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathrm{tx}_i\right)\mathbb{P}\left(\mathrm{tx}_i\right)}{\mathbb{P}\left(\mathrm{IP}_i \in C_o\right)} = \underbrace{\frac{\prod_{i=1}^{m}\mathbb{P}\left(\mathrm{tx}_i\right)}{\mathbb{P}\left(\mathbf{tx}\right)}}_{\text{const}}\cdot\frac{\prod_{i=1}^{m}\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathrm{tx}_i\right)}{\mathbb{P}\left(\mathrm{IP}_i \in C_o\right)^{m-1}}$$

The first factor is constant (depends only on the data), and it can be eliminated by the normalization of the probabilities. As $\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathbf{tx}\right) + \mathbb{P}\left(\mathrm{IP}_i \in C_n|\mathbf{tx}\right) = 1$ is valid,

$$\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathbf{tx}\right) = \underbrace{\frac{1}{\frac{\prod_{i=1}^{m}\mathbb{P}(\mathrm{IP}_i\in C_o|\mathrm{tx}_i)}{\mathbb{P}(\mathrm{IP}_i\in C_o)^{m-1}} + \frac{\prod_{i=1}^{m}\mathbb{P}(\mathrm{IP}_i\in C_n|\mathrm{tx}_i)}{\mathbb{P}(\mathrm{IP}_i\in C_n)^{m-1}}}}_{\text{const}}\cdot\frac{\prod_{i=1}^{m}\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathrm{tx}_i\right)}{\mathbb{P}\left(\mathrm{IP}_i \in C_o\right)^{m-1}} =$$

$$= \frac{1}{\frac{\prod_{i=1}^{m}\mathbb{P}(\mathrm{IP}_i\in C_o|\mathrm{tx}_i)}{\mathbb{P}(\mathrm{IP}_i\in C_o)^{m-1}} + \frac{\prod_{i=1}^{m}(1-\mathbb{P}(\mathrm{IP}_i\in C_o|\mathrm{tx}_i))}{(1-\mathbb{P}(\mathrm{IP}_i\in C_o))^{m-1}}}\cdot\frac{\prod_{i=1}^{m}\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathrm{tx}_i\right)}{\mathbb{P}\left(\mathrm{IP}_i \in C_o\right)^{m-1}}$$

The expression can be simplified further.

$$\mathbb{P}\left(\mathrm{IP}_i \in C_o|\mathbf{tx}\right) = \frac{\frac{\prod_{k=1}^{m}\mathbb{P}(\mathrm{IP}_i\in C_o|\mathrm{tx}_k)}{\mathbb{P}(\mathrm{IP}_i\in C_o)^{m-1}}}{\frac{\prod_{k=1}^{m}\mathbb{P}(\mathrm{IP}_i\in C_o|\mathrm{tx}_k)}{\mathbb{P}(\mathrm{IP}_i\in C_o)^{m-1}} + \frac{\prod_{k=1}^{m}(1-\mathbb{P}(\mathrm{IP}_i\in C_o|\mathrm{tx}_k))}{(1-\mathbb{P}(\mathrm{IP}_i\in C_o))^{m-1}}} =$$

$$= \frac{1}{1 + \frac{\mathbb{P}(\mathrm{IP}_i\in C_o)^{m-1}}{(1-\mathbb{P}(\mathrm{IP}_i\in C_o))^{m-1}}\cdot\frac{\prod_{k=1}^{m}(1-\mathbb{P}(\mathrm{IP}_i\in C_o|\mathrm{tx}_k))}{\prod_{k=1}^{m}\mathbb{P}(\mathrm{IP}_i\in C_o|\mathrm{tx}_k)}}$$

$\mathbb{P}\left(\mathrm{IP}_i \in C_o\right)$ is the initial frequency of occurrence of the clients in the C_o class, which is $1/\left|A\right|$. Although a Bitcoin client can use multiple IP addresses in the network, it is assumed that the $1/\left|A\right|$ value is a good approximation for the initial frequency in the vast majority of the cases. The total number of active clients varies with time in the scale of all considered transactions.

Thus, the $\overline{|A|}$ average of the different $|A|$ values is used as suggested in [11].

$$\mathbb{P}\left(\mathrm{IP}_i \in C_o | \mathbf{tx}\right) = \frac{1}{1 + \left(\frac{\frac{1}{\overline{|A|}}}{1 - \frac{1}{\overline{|A|}}}\right)^{m-1} \cdot \prod_{k=1}^{m} \left(\frac{1}{\mathbb{P}(\mathrm{IP}_i \in C_o | \mathrm{tx}_k)} - 1\right)} =$$

$$= \frac{1}{1 + \left(\overline{|A|} - 1\right)^{1-m} \cdot \prod_{k=1}^{m} \left(\frac{1}{\mathbb{P}(\mathrm{IP}_i \in C_o | \mathrm{tx}_k)} - 1\right)},$$

This formula brings in a technical problem. Huge numbers are multiplied together in the product, which become significantly biased in regular number representations by rounding and may result in overflow. To relax this problem, the second term of the denominator is written in an exponential form.

$$\mathbb{P}\left(\mathrm{IP}_i \in C_o | \mathbf{tx}\right) = \frac{1}{1 + \mathrm{e}^{\xi}},$$

$$\xi = (1 - m) \ln \left(\overline{|A|} - 1\right) + \sum_{k=1}^{m} \ln \left(\frac{1}{\mathbb{P}\left(\mathrm{IP}_i \in C_o | \mathrm{tx}_k\right)} - 1\right).$$

This results in the following practical formula.

$$\mathbb{P}\left(\mathrm{IP}_i \in C_o | \mathbf{tx}\right) = \frac{1}{1 + \exp\left[(1 - m) \ln \left(\overline{|A|} - 1\right) + \sum_{k=1}^{m} \ln \left(\frac{1}{\mathbb{P}(\mathrm{IP}_i \in C_o | \mathrm{tx}_k)} - 1\right)\right]}$$

This formula enables us to combine the probabilities assigned to the IP addresses by the transactions.