Do Connections Pay Off in the Bitcoin Market?

Zichao Yang*

Department of Economics, Virginia Tech

October 17, 2020

Abstract

I study the trading behavior of investors in this relatively new and unregulated bitcoin market. By parsing transaction data from the bitcoin blockchain, I search for addresses that are connected based on their trading behavior and identify the bitcoin investor network. I find that, from June 2016 to May 2019, addresses that are connected in the network earn 20.75% higher return than their unconnected peers on average. Furthermore, the return difference also exists among these connected addresses. By dividing the connected addresses into ten groups I find the addresses in the top two groups earn higher returns than the rest connected addresses. Among the addresses inside the top two groups, I find that, compared with degree centrality, higher eigenvector centrality is a more related indicator to higher returns.

JEL Classification: G11, G14

Keywords: Bitcoin, Network, Centrality, Asset Return

*E-mail: yzc@vt.edu. Address: 880 West Campus Drive, Blacksburg, VA 24061. I would like to thank my committee members for comments. SQL and R codes to replicate the results in this paper are available at www.yzc.me

1 Introduction

This paper asks two questions: (1) Comparing with investors who are unconnected in the bitcoin investor network, do connected investors earn a higher return? (2) Do more connected investors earn a higher return than less connected investors? To answer the first question, I construct a bitcoin investor network using raw transaction data from the bitcoin blockchain. Based on the network, all the qualified transaction addresses are clustered into a connected group and an unconnected group. I find that, on average, over the period from June 2016 to May 2019, the return of the connected group is 20.75% higher than that of the unconnected group. To answer the second question, I divide the addresses into eleven groups: the unconnected addresses serve as the base group, and the connected addresses are divided into ten groups based on either their degree centrality or their eigenvector centrality. I find that there is another return difference between these connected addresses. The addresses in the top two groups consistently show higher returns than all the rest connected addresses, no matter the groups are divided based on degree centrality or eigenvector centrality. Furthermore, I test the importance of degree centrality and eigenvector centrality to the addresses returns in the top two groups, and I find that eigenvector centrality is a better measure to predict these most central addresses' investment returns.

I also conduct a series of robustness checks. I choose three different thresholds of establishing the network connection between addresses, and 2-week and weekly returns are used to replace the monthly returns used in the main results. It turns out that the main results of this paper are not sensitive to these choices. To see how the results change as the bitcoin market is becoming bigger and more mature, I also break down the whole sample into three periods. All these results consistently show that investment return difference exists not only between the connected and unconnected group but also exists among those connected addresses.

A number of studies focus on bitcoin investors' characteristics or behaviors. Xi et al. (2020) conduct a web-based revealed preference survey to profile Initial Coin Offering (ICO) investors' characteristics in China and Australia, and they find the investing motivation factors are different in these two countries. After analyzing the bitcoin trading data from Mt. Gox, Glaser et al. (2014) conclude that new bitcoin users from 2011 to 2013 tend to treat bitcoin as an alternative investment option rather than a new currency. This result aligns with the survey result conducted by Mahomed et al. (2018) in South Africa. Using Mt. Gox exchange trading data, Gandal et al. (2018) and Chen et al. (2019) demonstrate that there are serious market manipulations inside Mt. Gox exchange, which causes unprecedented spikes in the bitcoin price. Exchange data is the ideal data source for us to study bitcoin investors' behavior. However, exchange data are usually tightly secured by bitcoin exchanges as their own asset. The leaked Mt. Gox exchange trading data has been widely used in multiple studies, but the drawback is that this dataset only covers the date period from April 2011 to November 2013. It does not shine a light on more recent development in the bitcoin market. In this study, I am using the data obtained directly from the bitcoin blockchain. Bitcoin blockchain raw data is freely accessible to anyone on the internet, which allows me to choose the periods that can better capture the dynamic change of the bitcoin market in the past few years. However, bitcoin blockchain raw data also brings new challenges to the analysis: the transaction data are connected to bitcoin addresses, not individual investors. In this paper, I proposed a simplified parsing procedure to mitigate this drawback.

Most studies on the investors network are done only on stock markets. By exploiting a dataset that includes all account-level trading information on the Istanbul Stock Exchange in 2005, Ozsoylev et al. (2014) find that investors with higher centrality can earn higher returns. Rossi et al. (2018) show that investment managers who are better connected tend to have a better portfolio performance. Ahern (2017) analyzes an illegal insider trading network and finds that people in this network earn a 35% return over 21 days, and more central traders in this network can earn even greater returns. Walden (2019) introduces a dynamic noisy rational expectations model and finds out that an agent's profitability is determined by Katz centrality. To my best knowledge, most information network studies are conducted using trading data from the stock market, which is usually a highly regulated and mature market. However, here, I look at the bitcoin market as a new market with fewer government regulations. Liu and Tsyvinski (2020) find that coin market returns are positively related to cryptocurrency network growth rates, and current cryptocurrency prices include information about the expected network growth. As a complement to their study, this study focuses on how returns differ among investors inside the bitcoin network using address-level transaction data. This paper will closely follow the approach in Ozsoylev et al. (2014), and see if the widely observed positive relationship between centrality and profitability also applies to the bitcoin market.

The remainder of the paper is structured as follows: section 2 introduces the basic technical background about bitcoin. Section 3 explains data sources and the blockchain data parsing procedure. Section 4 constructs the bitcoin investor network. Section 5 shows the empirical results about the effect of connectedness and centrality on bitcoin investment returns. Section 6 concludes the paper.

2 Background

When bitcoin first debuted in 2008, Nakamoto (2008) promoted it as a replacement of our current payment system. In Satoshi's mind, bitcoin is a decentralized payment system that does not need a centralized clearinghouse but can still keep transaction records secure and immutable. Bitcoin the token itself is, more or less, a byproduct of this revolutionary payment system. Twelve years later, China government is now testing its government-back cryptocurrency in some cities. Facebook proposed a new cryptocurrency Libra, and the project is set to launch in 2020. Cryptocurrency this concept has gained tremendous popularity among regulation institutions and academia. In the following sections, I will lay out some necessary technical details about bitcoin. I hope this can help readers better understand how to identify the bitcoin investor network from bitcoin transactions.

2.1 Structure of a bitcoin block

Blocks are defined as storage units that contain confirmed transaction data. These blocks comprise what is called a blockchain, which is a sequence of blocks. Each block in the blockchain contains a piece of information about the location of its previous block.[1] The size of each block has been set to be less than 1 MB. The block size limit ensures that each block can only contain a certain number of transaction records. Meanwhile, every block includes a block header (BH), which includes six fields: version, previous block hash, merkle root, timestamp, target and nonce. The complete structure of a bitcoin block is well summarized by Antonopoulos (2017) as in Table 1.[2] [3] In this study, the most relevant data comes from the Transactions section in every bitcoin block.

2.2 Bitcoin transactions

Every transaction in the bitcoin system consists of two parts: transaction input and transaction output. Except for coinbase transactions, a transaction input usually should come from an unspent transaction output (UTXO) generated from a previous transaction. In Table 2, I list the most important components in a bitcoin transaction.

When we say a user has received bitcoin from others, what we mean is that this user's

[1]Technically speaking, blockchain is not just a chain, it has many branches due to the occurrence of forks. But this is irrelevant to this study. So I can simply think it is just a chain constructed by thousands of blocks.

[2]Hash function can transfer data of arbitrary size into data of a fixed size. And the output can be called hash. Hash function is pre-image resistance, which means as long as the input is the same, you can always get the same output, but you are not able to deduce the input from the output.

[3]Merkle root is the root of merkle trees, which are data structures that helps nodes quickly verify transactions and reduce data transaction.

bitcoin wallet has detected a new UTXO that can be spent by one of the private keys controlled by this wallet. One feature of UTXO is that UTXO is an indivisible chunk of bitcoin, just like a dollar bill. Users cannot spend a part of a UTXO in one transaction just like they cannot use a part of a dollar bill in one transaction. However, the good news is, unlike in our daily life that we have to worry about how many 5 dollar bills and 20 dollar bills we need to take out of our wallets to fulfill a $107 transaction, bitcoin wallets will take care of these tedious calculations. The user only needs to tell the bitcoin wallet what the desired transaction amount and transaction fee are, and then the bitcoin wallet can automatically select from all the UTXOs that are controlled by the wallet to compose an amount equal or greater than the desired transaction amount.

To verify the ownership of a UTXO, bitcoin relies on the public-key cryptography. A digital wallet can generate, in theory, unlimited pairs of public/private keys. A public key is included in every UTXO. Public keys tell the whole bitcoin network who are the receivers of these UTXOs. Private keys function as certifications telling miners you are the real owner of these UTXOs, because one public key can only be deciphered by its corresponding private key. To simplify the analysis, here we can think of each UTXO as a dollar bill with a face value equal to the amount of bitcoin embedded in this UTXO. For example, in person A's digital wallet, there are 2 pairs of public/private keys (public key P1/private key1, public key P2/private key2). P1 link to UTXO1 (0.3 bitcoin) and UTXO2 (0.7 bitcoins) respectively. And P2 links to UTXO3 (2 bitcoin). Now person A wants to send 2.5 bitcoins to person B's public key (public key P3). As I mentioned above, bitcoin works like a cash system. Person A cannot send 1 bitcoin from P1 and 1.5 bitcoins from P2 to public key P3. What person A can do is send out all 3 bitcoins: 2.5 bitcoins to public key P3 and 0.49 bitcoin to his/her own public key. The 0.01 bitcoin difference serves as the transaction fee collected by the miner who confirmed the transaction. A simplified transaction framework is illustrated in Figure 1.[4]

In Figure 1, person A sends back the rest 0.49 bitcoin change to public address P1, that is how we can figure out the 0.49 bitcoin is not a real transaction. In the bitcoin community, people usually called these self-transactions as change transactions. Change transactions are easy to be detected if people keep sending back the changes to any input addresses. However, directly sending back changes to one of the already used addresses may give other people the ability to trace back all the transactions related to person A. To protect his/her privacy, A can ask the bitcoin wallet to generate a new pair of public key X/private key X and send the rest 0.49 bitcoin to public key X (see Figure 2). By doing so, the outsider can never tell if

[4]Here I use public keys instead of bitcoin addresses (publish key hashes) to illustrate a simplified version of transaction process. Nowadays, majority transactions actually use bitcoin addresses on the output side.

the 0.49 bitcoin transaction is a real transaction or just a change transaction. This practice is usually recommended to anyone who wants to use bitcoin and also wants to keep their privacy. However, this practice also brings out two big challenges to academic researches: (1) it is difficult to cluster the transaction data into individual investor level because investors can generate new addresses for every new transaction to cover their trading traces; (2) it is quite challenging to filter out all the change transactions from the transaction dataset, given investors can always generate new addresses to receive changes. There is no perfect way to address the above two difficulties when we are dealing with bitcoin data. In the following section, I will propose some solutions to mitigate the concern.

3 Data

3.1 Data source

In the bitcoin transaction raw data, the input side does not include the real transaction amount. Instead, the input side includes the UTXOs that are supposed to be used to fulfill the transaction. It is the miners' responsibility to verify if these UTXOs contain enough bitcoins to fulfill this transaction. The way how raw transaction data is stored in blockchain makes it not user-friendly to our human eyes. Luckily, Google has already pre-parsed the raw data and made the dataset more human-friendly. In the following transaction analysis, I will use Google's bitcoin transaction data as my data source. Considering the huge amount of data that needs to be processed, I choose to use Google's BigQuery service to pre-process the whole dataset and use R for the final stage of data analysis.

I also use the minute-level bitcoin price data from Bitstamp and Coinbase to construct a block-level bitcoin price data. If the price data is available on both Bitstamp and Coinbase, I use the average value to represent the block-level bitcoin price data. If the price data is only available on either Bitstamp or Coinbase, I use the available data to represent the block-level bitcoin price data. If the price data is not available from both sources, the value will be imputed based on the neighboring available values.

3.2 Data parsing

As mentioned in the background section, in theory, users can generate a new public/private key pair for each transaction to hide their identities. One challenge when dealing with bitcoin data is that we can never be certain if two addresses are owned by the same person or owned by two different people. Meiklejohn et al. (2013) propose two heuristic rules to parse bitcoin addresses data: (1) if two public keys are used as inputs in the same transaction, then these two public keys are considered to be owned by the same person; (2) the change address used as output in a transaction should be recognized as it is also controlled by the person who initiates the transaction.[5] Even though these two heuristic rules look plausible, they are not practically perfect in some situations. Kalodner et al. (2020) point out that rule (1) does not apply to CoinJoin transactions.[6] Meanwhile, it is very difficult to distinguish change transactions from real transactions. One way proposed to detect change address is if this address has only been used once, then it is a change address and should be owned by the same person who owns the input addresses. However, this method may mislabel some long-term investors and create false super-clusters. Especially, in this study, more than 50% of addresses during the examined period only showed up once in the dataset.

In this study, I am mostly interested in what happened from June 2017 to May 2018. In this period, the bitcoin price increased from 2,000 dollars to almost 20,000 dollars. Bitcoin and related cryptocurrencies dominated worldwide business news coverage. Meanwhile, I also know that bitcoin had the biggest price drop from November 2018 to February 2019. To test the dynamic change in the bitcoin market, I examine three periods in this study: period 1 from 2016-06-01 to 2017-05-31, period 2 from 2017-06-01 to 2018-05-31, and period 3 from 2018-06-01 to 2019-05-31. Period 1 represents the bitcoin market when it is not popularized by mass media outlets. Period 2 represents the period when bitcoin became a mainstream investment option and experienced an astonishing price jump. Period 3 represents the period when the public hype about bitcoin was dying out, and bitcoin had the largest price drop in its history.

Table 3 shows how many unique transaction addresses are recorded into the bitcoin blockchain in each period. The table also shows how many addresses showed up multiple times in each period. More than 100 million unique transaction addresses are recorded into the bitcoin blockchain in each period, which makes it infeasible for me to include all the addresses into the dataset. Furthermore, more than 90% of addresses showed up less than 3 times in each period. In this study, I use monthly returns, 2-week returns, and weekly returns to measure investors' performance on a short-term horizon. If the address does not trade frequently, I can safely assume this investor does not care that much about the short-term horizon. Also, considering computing pressure, only the addresses that appeared at least 10 times are included in the sample set, which means that around 2 million unique transaction addresses will be examined in each period.

Given the large amount of data I am dealing with, I choose to use a set of simplified rules to parse the bitcoin transaction data:

[5] Change address refers to the output address(es) in a change transaction.

[6] CoinJoin transctions combine transctions from different spenders into one transaction, in this way outsiders cannot tell which spender paid which recipient(s).

Step 1: Subsample the original transaction dataset to only include the addresses that showed up at least 10 times during a given period.[7] In this step, I filter out all the privacy-centered users who only use addresses once for any transaction, along with the users who do not care about short-term profitability. This step can help me significantly reduce computing pressure. Meanwhile, those privacy-centered bitcoin users may have other motivations besides the financial one when they choose to use bitcoin, so remove them can also reduce noise in the data sample.

Step 2: In each transaction, I remove all the output addresses that also show up on the input side because these output addresses are used as change addresses for sure. In this step, I only remove these most obvious change addresses from the dataset. I do not want to adopt more aggressive methods to parse out change addresses because there is no perfect method to detect all the change addresses and I may end up mislabeling some addresses.

Step 3: In each transaction, I use one input address A to represent all the input addresses. It is almost certain that the same person owns all the input addresses in the same transaction. As I mentioned at the beginning of this subsection, CoinJoin transactions can be exceptions. However, comparing with joining in a CoinJoin transaction to protect privacy, a much easier way is to generate different addresses for different transactions. The addresses in this dataset have already been used more than 10 times, I do not think their owners will be interested in participating in CoinJion transactions to protect their identities.

My method will certainly miss some connections between addresses and treat some addresses owned by the same person as they have different owners. However, this simplified method can significantly reduce computing power and exclude false super-clusters. In this next section, I will use the parsed dataset to construct the bitcoin investor network.

4 The Bitcoin Investor Network

4.1 Identifying the bitcoin investor network

I define the bitcoin investor network in the same fashion as the empirical investor network proposed by Ozsoylev et al. (2014). However, there are a few differences between the bitcoin market and the stock market: (1) In the bitcoin market, there is only bitcoin this one asset to trade. (2) Bitcoin transactions are confirmed in batches. When a new block is verified by the bitcoin network, all the transactions inside this block are confirmed at the same time. Hence, I define the bitcoin investor network as:

[7] In this subsample, each transaction record may no longer preserve all the involved addresses. However, my goal is calculate address level return, so transaction level incompleteness will not affect my final result.

Definition The bitcoin investor network , $\varepsilon^{\Delta b,M}$, is defined such that for each pair of investors, i, $j \neq i$, $\varepsilon^{\Delta b,M} = 1$ if and only if agents i and j traded in the same direction within Δb blocks at least M times over the period.

I calculate the bitcoin investor network using the $\Delta b = 1$ and picking $M = 10, 30, 50$. The first reason why I keep the time window Δb short is that I want to separate the information-driven trading, which is usually referred as "fast" trading and happens in a short time window, from other types of trading. The second reason for keeping Δb short is it can reduce computing pressure.

This paper constructs bitcoin investor networks separately for three different periods (see subsection 3.2). The reason behind this decision is that I am concerned about the change in the composition of bitcoin investors given the landscape of the bitcoin market has changed a lot during these three periods. For example, some addresses may be active in period 1 but not active in period 2. By estimating the network using the whole sample, I may under-sample the addresses that are only active in a certain period. Meanwhile, if there is no dramatic investor change, estimating the network by periods should give us similar results as using the whole data sample. If there is significant investor change, then estimating the network by periods can better capture the dynamic change in the bitcoin investor network. Table 4 provides the summary statistics of this bitcoin investor network given $M = 10, 30, 50$ during different periods. We can see that, overall, this network is very sparse, and the statistics are mostly similar across these three different periods.

4.2 Node centrality and trading return

After identifying the bitcoin investor network, I can now measure the centrality of every node (address) in this network. Many methods have been proposed to measures network centrality.[8] In this paper, I use both degree centrality and eigenvector centrality, same as the centrality measures used in Ozsoylev et al. (2014). Degree centrality measures how important a node is in the network by counting how many other nodes are connected with this node. Eigenvector centrality measures a node's importance by considering the importance of all the neighbors connected with the node. Degree centrality and eigenvalue centrality describe different characteristics of the nodes in this bitcoin investor network. The degree centrality measures how widely connected a node is in this network. The eigenvector centrality tells us how important a node's position is in this network. If a node is the only one that connects two

[8] A thorough discussion about network centrality, please see Jackson (2010).

large communities in the network, this node may not have a high degree centrality but can have a high eigenvector centrality. Due to the computing limitation of calculating eigenvector centrality, in this paper, I only calculate nodes' centrality values when $M = 50$.

To measure trading returns, I follow the same approach as in Barber et al. (2009) and Ozsoylev et al. (2014). I define a window length to be $\Delta B = 4320$ blocks, which is roughly a month given bitcoin blocks are generated, on average, every 10 minutes. Defining the window in terms of blocks can help us precisely match the trading returns with the transaction data, which is also generated by blocks. In the Appendix, for the robustness check, I also change the window length to 2160 blocks and 1008 blocks, which correspond to two weeks and one week. For each trade, z, the potential return is:

$$\mu_{i,z} = sign * \left(\frac{P^{b+\Delta B} - P^b}{P^b} - r_{tbill} \right)$$

where $P^{b+\Delta B}$ is the bitcoin price B blocks later after the trade happened in block b. When we set $B = 4320$, this roughly tells us the monthly return of each trade. r_{tbill} comes from the daily U.S. Treasury yield curve rates, I impute the rate value for weekends based on the neighboring available values.[9] Both data sources have been annualized to make sure they are comparable.[10] The symbol $sign$ indicts the trade direction, and it should be negative for input addresses and positive for output addresses. To further reduce the computing pressure and lower the computing memory requirement, I group the dataset by addresses and months, then calculate the average as an address's return for each month.

Due to the difficulty of linking all the addresses that are owned by the same investor, it is not possible to accurately calculate the trade amount of every investor. This drawback makes the value-weighted average return proposed in Ozsoylev et al. (2014) highly inaccurate in this study. For example, if there are three addresses: a, b, c that are owned by investor A. Suppose c serves as a safe vault, and A usually transfer a large amount of bitcoin into c and move a small amount of bitcoin out of c during the period when bitcoin price is increasing rapidly. If I failed to realize that these addresses are owned by the same person and all the transactions among them are actually change transactions, which should be excluded from the dataset, I will amplify these abnormal returns caused by these change addresses in regressions by using weighted return. To mitigate the effect of these undetected change transactions, I think the potential return is a better candidate to measure investors' performance. Potential return

[9] The daily U.S. Treasury yield curve rate can be found on the U.S. department of the Treasury websitte.

[10] To annualized the monthly return, I time the results with 365/30.5. Following the same logic, I use 365/14 for two-week return data and 365/7 for weekly return data. I did not use the formula $(1 + r)^t - 1$ because bitcoin returns are very high in some months, $(1 + r)^t - 1$ may make the performance of an address in a high return month dominates its whole year performance.

measures how often an address can show up at the right side of a trade at the right time and ignores the trading amount due to incomplete trading information on the individual investor level. By using potential returns, we can focus on the effect of centrality on the profitability of addresses without worrying about if these addresses are owned by the same investor or not.

4.3 Stability of the bitcoin investor network over time

In subsection 4.1 I have already stated the benefits of estimating the bitcoin investor network individually for three different periods. Here I try to examine if the bitcoin investor network is stable over these periods. Following Ozsoylev et al. (2014) approach, I try to answer whether the bitcoin investor networks are more similar in these three periods than if they were randomly generated in these periods.

There are two different methods to test the stability of the bitcoin investor network. The first intuitive way is assuming the network links are generated randomly. Suppose a network contains N nodes and k_1 links, then the possibility of any two different nodes that are linked is k_1/K, where $K = N(N-1)/2$ is the total possible links in this network. Suppose the network in two different periods has k_1 and k_2 links respectively, and $k_1 << K, k_2 << K$, then the expected number of overlap links ($E_{random}[y]$) in these two periods should be approximately:[11]

$$E_{random}[y] \approx \frac{k_1 k_2}{K}$$

where k_1 is the number of links in the first period, k_2 is the number of links in the second period, K is all the possible links among all the addresses in the network.

Meanwhile, considering the bitcoin investor network is not truly random generated and the degree distribution has heavy tails, I also adopt the degree-adjusted method proposed in the Internet Appendix of Ozsoylev et al. (2014). By taking degree into consideration, the degree-adjusted expected number of overlap links ($E_{\text{Degree-adjusted}}$) between period 1 and period 2 can be written as:

$$E_{\text{Degree-adjusted}} = \frac{k_2}{2k_1 N} \sum_{N}^{i=1} (D_i - 1)^2$$

where N is the number of investors in the network, D is the degree distribution of the network in period 1.

[11]Condition $k_1 << K, k_2 << K$ is easily satisfied in this study given the bitcoin investor network is very sparse.

Table 5 shows the results of the above two methods when $M = 50$.[12] We can see that the actual number of overlap links between any two periods is significantly higher than the expected number of overlap links when we assume the network links are randomly generated. When I am using the degree-adjusted method, the actual number of overlap links is still significantly higher than the theory predicted. Using both methods, we should have high confidence to reject the null hypothesis that the network is not stable over time. I also used $M = 10, 30$ for robustness checks. The results for $M = 10, 30$ are included in the appendix (Table A.1, Table A.2). We still can see that, under the random generation assumption, the real overlap link data still reject the null hypothesis that the network is not stable over time. However, the picture is not so clear when we compare the results using the degree-adjusted method when $M = 10$. These results may indicate that high-connected, more central nodes stay active in the bitcoin investor network longer than the rest nodes. Hence, when I set $M = 50$ and rule out more less-connected nodes, we can observe a much more stable network over time.

Given we are almost certain that the network is stable over time when $M = 50$, I will report the results using $M = 50$ as the main results and use $M = 10, 30$ for robustness check. Also, I will report the results for the whole data sample along with the results for each period to provide more detailed information.

5 Results

5.1 Do connected addresses have better profitability?

I divide the addresses into two groups based on their degree values: the unconnected and connected groups. The unconnected group includes all the addresses that are not connected with any other addresses in my sample. The connected group includes the addresses which are connected with other addresses. To avoid influence by outliers and maintain consistency with the following regression part, I truncate the addresses in the top two percentiles of connectedness in the dataset. I calculate the median monthly return of all the addresses in each group for every month in the sample period under $M = 10, 30, 50$, and the results are shown in Figure 3, Figure 4, and Figure 5.

When $M = 10$, we can see that the median return of the connected group is much higher than that of the unconnected group when the bitcoin market return, in general, is increasing. The most obvious example is in November 2017, the median annualized monthly

[12] The number of links in Table 5 is slighted different from the number in Table 4. This is because Ozsoylev et al. (2014) counted self-links in their method. To maintain consistency, I also include self-links in this table.

return in the connected group is around 917% and 556% for the unconnected group. Also, the connected group is doing worse during the period when the bitcoin market return, in general, is dropping. The patterns are consistent when I increase the threshold to $M = 30$ or 50 even though the median return difference between these two groups shrinks. This is not surprising because more addresses have been categorized into the unconnected group when we raise the threshold M to 30 or 50.

To further examine if the return difference between the connected and the unconnected group is consistent, in addition to using monthly return data, I also plot out the median 2-week return and weekly return of these two groups under $M = 10, 30, 50$ (see Figure A.1-Figure A.6 in appendix). The results consistently show that the connected group gains higher returns than the unconnected group when the bitcoin market return, in general, is increasing. And sometimes, when the bitcoin market return is dropping, the connected group may suffer more than the unconnected group. No matter which format of returns or threshold M I choose, the pattern is consistent and clear.

In the next step, I want to make sure the connected group's higher profitability is not driven by a small group of addresses in the group. Hence, I further divide the addresses in the connected group into 10 subgroups based on their degree values. I then calculate the median returns of each group over the full time period and compare them with the median return of the unconnected group. Figure 6 shows the results under different threshold $M = 10, 30, 50$. No matter which M I pick, it is clear that the median return of every decile in the connected group is higher than the median return of the unconnected group. This conclusion still holds when I replace monthly returns with 2-week returns or weekly returns (see Figure A.7 and Figure A.8).

To better understand the effect of being connected, I do multivariate regressions on address level to see how connectedness can affect the return rate. Meanwhile, I also control the monthly market trade volume (Q), the monthly market number of trades (N), the monthly individual trade volume (q), the monthly individual number of trades (n) and the monthly bitcoin price volatility (V), which is calculated as the standard deviation of bitcoin prices in each month. Table 6 shows the summary statistics of these variables over the whole sample period and breaks it down into period 1,2, and 3, representing the different bitcoin market periods described in subsection 3.2. All of these variables are in log-from.

I regress the returns of addresses on these variables mentioned in Table 6 using the whole data sample. Then I further break it down to the three different periods. Table 7 shows the results when $M = 50$. The difference between the connected and unconnected groups is relatively small in period 1, during which the bitcoin was only popular in some small circles. The difference becomes very obvious during period 2, when bitcoin finally entered the public eye and gained its worldwide popularity. During period 2, the addresses in the connected group, on average, have almost 45% higher return rate than their peers in the unconnected group. Hence, it is obvious to say connected addresses have better profitability. I also conduct the same regression using $M = 10, 30$, and replace monthly returns with 2-week returns or weekly returns for robustness check, and the results consistently show that the connected addresses have significantly higher profitability than the unconnected ones. The difference is most obvious in period 2, during which the bitcoin price had a rapid increase and reached its historically high. Detailed regression results can be found in the appendix from Table A.3 to Table A.10.

5.2 Do more central addresses earn higher returns?

I have shown that addresses being connected in the network have higher returns than their unconnected peers, and naturally, the next question to ask is whether it matters to be more central than your peers in the network.

There are multiple ways to measure network centrality, and I choose the most commonly used measures: degree centrality and eigenvector centrality. Degree centrality measures how connected a node is by counting the number of nodes connected to this node. Usually, we expect a more connected node to be playing a more important role in spreading information in the network as an information hub. Being a hub may give a node the advantage of accessing more exclusive information about the market, which may help the node earn higher return. However, it is also worth mentioning that some addresses may not have a high degree value, but they may become important information hubs by occupying strategical positions in the network and being connected to some influential nodes. That is why I also consider the eigenvector centrality in this analysis. Eigenvector centrality measures how important a node is by looking at how important its neighbors are. Eigenvector centrality is calculated using iGraph package in R, and the value has been normalized to be between 0 and 1. Due to computing limitations, here, I only report the results when $M = 50$.[13]

Figure 7shows how addresses with different degree centrality values perform in the full sample. I divide the addresses in the connected group into 10 subgroups based on their degree centrality values. Then I calculate the median monthly returns of each group over the whole period and compare them with the median return of the unconnected group. This figure shows that connected subgroups always have higher returns compared with the unconnected group. Also, we can observe another return jump around the top two groups. These results

[13]In this study, Eigenvector centrality is calculated on Virgina Tech ARC servers using 20 cores (128 GB shared memory). Given the size of the network, more computing resources are needed to calculate the Eigenvector centrality when M = 30 or 10.

are consistent no matter which return data I use. Using the same approach, I also show the performance of the addresses with different eigenvector centrality values in the full sample in Figure 8. And the results are very similar.

To further test the higher return associated with the top two groups is not driven by a specific period of data, I bread down the full sample into the three different periods mentioned in subsection 3.2. The figures showing the relationship between return and degree centrality/ eigenvector centrality among different groups in different periods can be found in Figure A.9 to Figure A.14. In these figures, we can see that no matter which centrality measure we choose to use, addresses in the top two most central groups are earning higher returns than their other connected peers in the first two periods. A slight difference in these two periods is that the return jump in the top two groups in the first period is not so dramatic as in the second period. However, in the third period, the figures show that addresses in the top two groups are doing even worse than the addresses in the unconnected group. Directly drawing conclusions from the figures from different periods may be misleading. The different shapes of the relationships between return and centrality may serve as a piece of evidence reconfirming these three periods have captured some dynamic changes in the bitcoin investment environment. As we mentioned in subsection 3.2, the first period represents a period when this token is still out of the public eye; the second period represents a period when ordinary investors flocked into the market and bitcoin price reached its historically high; and the third period represents a period when bitcoin had the largest price drop in its history.

To carefully examine the relationship between returns and groups with different centrality, I run regressions for returns on groups with different centrality. The base group consists of addresses without connections. The 1-10 groups in regression results represents the least central group to the most central group. The groups are divided using either degree centrality or eigenvector centrality. In the regressions, I also control bitcoin price volatility, market number of trades, market trade volume, individual trade volume and individual number of trades.The summary statistics of these variables are in Table 6. Table 8 shows the relationship between monthly returns and groups with different centrality. From Table 8, we can reach two conclusions: (1) most time, connected groups have higher returns than the unconnected base group. There are only two exceptions in the first period, during which addresses in group 6 and group 7 may have lower returns than their unconnected peers. We do not see this kind of exceptions in more recent periods. (2) the top two groups are having higher returns than the other groups. This conclusion is consistent no matter these groups are identified based on degree centrality or eigenvector centrality.

Table A.11 and Table A.12 replace the monthly returns with two-week returns and weekly returns for the robustness check. To visualize all these results, I also plot out the coefficients from these regressions. Figure 9 and Figure 10 show the coefficients in the regressions for returns on groups with different centralities using the full dataset. We can clearly see a jump on the coefficient values around the top two groups in these two figures. This pattern also matches the upward trend we observed in Figure 7 and Figure 8. Also, this pattern is consistent no matter which return data we choose to use.

Recall in Figure A.9 to Figure A.14, we find out that the median return in the top two groups is higher than that in other groups during the first two periods, but the conclusion does not hold in the third period. After controlling some individual-level and macro-environment variables, I also plot out the coefficients in these different periods. Figure A.15 to Figure A.18 show that the top two groups consistently have larger coefficient values, no matter whether the groups are identified using degree centrality or eigenvector centrality. In the third period, we can see the top two groups are having higher returns than the unconnected group because the coefficients are positive. Meanwhile, addresses in these two groups are still earning higher returns than their other connected peers. Among all these regressions, the top two groups only seem to perform worse than their other connected peers in the third period when we use weekly returns. Even in this case, the addresses in the $80-90\%$ range still have the highest return among all the groups. It is only the top 10% that does not show high weekly returns.[14]

All these discussed results suggest that the connected group can be further divided into two subgroups: the central group, which includes addresses whose centrality belongs to the top 20%, and the less-central group, which includes all the rest connected addresses. Besides the return difference between the unconnected addresses and connected address, now we observed another return difference among these connected addresses: the addresses in the central group are earning even higher returns than the rest connected addresses.

5.3 Centrality measures in the central group

In the last subsection, we find the addresses that belong to the central group are having higher returns than the rest connected addresses. And the conclusion is true no matter

[14]A plausible explanation for the abnormal performance of the top 10% in the third period is that I accidentally include some exchanges or other institutions in this group. These institutions, like bitcoin exchanges, may have goals other than pursuing high returns in the bitcoin market. In this regression dataset, I truncate the top two percent most connected addresses in the original dataset to avoid influence by outliers, and we get the most connected address in the first period has 7707 connections, in the second period has 5896 connections. However, in the third period, the most connected address still has 16082 connections. If I only include the addresses with less than 7000 degrees into the regression, then the results will be similar to what we see in the first two periods. For the sake of consistency, in this study, I choose to only truncate the top 2% most connected addresses in all three periods.

which centrality measure is used. One explanation for this is maybe degree centrality and eigenvector centrality are both important. Another explanation is that these two measures are correlated, but they may not be equally important. In this subsection, I try to test these two theories. I run regressions for returns on degree centrality and eigenvector centrality and the interaction of these two measures among the addresses which belong to the top 20% of both centrality measures. Same as previous regression, here I also include bitcoin price volatility, market number of trades, market trade volume, individual trade volume, and individual number of trades as control variables.The summary statistics of these variables are in Table 9.

Regression results are reported in Table 10. We can see that, compared with degree centrality, eigenvector centrality seems to be a better measure to predict investment returns of these most central addresses in the bitcoin investor network. The results from the different period samples make the conclusion even more obvious: the coefficient of degree centrality becomes negative when we break down the full sample into three periods. The conclusion also holds no matter which type of returns I use in the regressions.

Even though the regression results consistently show that eigenvector centrality is a better measure to predict the returns of these most central addresses, the reason behind this conclusion still bears further research. One plausible explanation is that the results pick up the exchange addresses effect. Bitcoin exchanges profit from commissions, which are usually charged inside exchanges and off the bitcoin blockchain. These exchange addresses may not care much about on-chain investment returns, but they also tend to have a large number of connections (high degree centrality). If these exchange addresses are still part of the most central addresses in this dataset, then we may find negative coefficients of degree centrality and the interaction term of degree and eigenvector centrality. Another plausible explanation is that adding more connections may not be beneficial to those addresses which are already in the most central position in the network. What these addresses need is connecting with more central addresses like themselves. On some occasions, adding low-quality connections may bring in more noise and affect these central addresses' investment performance.

6 Conclusion

This paper reaches two conclusions: (1) compared with their unconnected peers, connected investors earn higher returns, and (2) the connected investors can be further divided into two subgroups: the less central group describes those who have connections but are not connected to other important investors, the central group describe those who occupy more central positions in the network and earn even higher returns than their peers in the less-central group. I also find that, compared with degree centrality, eigenvector centrality is a better measure to capture the addresses' investment returns in the central group.

In this paper, I assume each address in my dataset is owned by a unique investor. However, due to the data parsing limitation, this approach almost certainly mis-categorize some addresses that are actually controlled by the same person. To mitigate the effect of this mis-categorization issue on results, I use each address's potential return instead of the weighted actual return. With the improvement of data parsing technology, such as BlockSci (Kalodner et al., 2020), we may be able to cluster bitcoin addresses into the individual level with higher accuracy. We can then calculate an individual investor's weighted actual return, which may give us a more precise picture of how connectedness and centrality are related to return in the bitcoin market.

References

Kenneth R Ahern. Information networks: Evidence from illegal insider trading tips. *Journal of Financial Economics*, 125(1):26–47, 2017.

Andreas M Antonopoulos. *Mastering Bitcoin: programming the open blockchain*. O'Reilly Media, Inc., 2017.

Brad M Barber, Yi-Tsung Lee, Yu-Jane Liu, and Terrance Odean. Just how much do individual investors lose by trading? *The Review of Financial Studies*, 22(2):609–632, 2009.

Weili Chen, Jun Wu, Zibin Zheng, Chuan Chen, and Yuren Zhou. Market manipulation of bitcoin: evidence from mining the mt. gox transaction network. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 964–972. IEEE, 2019.

Neil Gandal, JT Hamrick, Tyler Moore, and Tali Oberman. Price manipulation in the bitcoin ecosystem. *Journal of Monetary Economics*, 95:86–96, 2018.

Florian Glaser, Kai Zimmermann, Martin Haferkorn, Moritz Christian Weber, and Michael Siering. Bitcoin-asset or currency? revealing users' hidden intentions. *Revealing Users' Hidden Intentions (April 15, 2014). ECIS*, 2014.

Matthew O Jackson. *Social and economic networks*. Princeton university press, 2010.

Harry Kalodner, Malte Möser, Kevin Lee, Steven Goldfeder, Martin Plattner, Alishah Chator, and Arvind Narayanan. Blocksci: Design and applications of a blockchain analysis platform. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pages 2721–2738, 2020.

Yukun Liu and Aleh Tsyvinski. Risks and Returns of Cryptocurrency. *The Review of Financial Studies*, 09 2020. ISSN 0893-9454. URL https://doi.org/10.1093/rfs/hhaa113.

Nadim Mahomed et al. *Understanding consumer adoption of cryptocurrencies*. PhD thesis, University of Pretoria, 2018.

Sarah Meiklejohn, Marjori Pomarole, Grant Jordan, Kirill Levchenko, Damon McCoy, Geoffrey M Voelker, and Stefan Savage. A fistful of bitcoins: characterizing payments among men with no names. In *Proceedings of the 2013 conference on Internet measurement conference*, pages 127–140, 2013.

Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008. URL https://bitcoin.org/bitcoin.pdf.

Han N Ozsoylev, Johan Walden, M Deniz Yavuz, and Recep Bildik. Investor networks in the stock market. *The Review of Financial Studies*, 27(5):1323–1366, 2014.

Alberto G Rossi, David Blake, Allan Timmermann, Ian Tonks, and Russ Wermers. Network centrality and delegated investment performance. *Journal of Financial Economics*, 128(1): 183–206, 2018.

Johan Walden. Trading, profits, and volatility in a dynamic information network model. *The Review of Economic Studies*, 86(5):2248–2283, 2019.

Dingli Xi, Timothy Ian O'Brien, and Elnaz Irannezhad. Investigating the investment behaviors in cryptocurrency. *The Journal of Alternative Investments*, 2020.

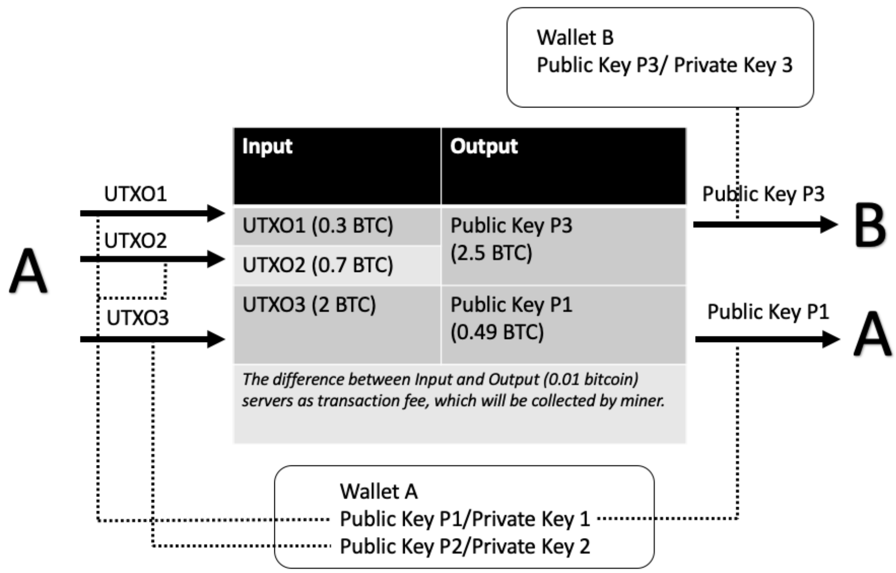

Figure 1: A simplified transaction framework I This figure shows a simplified transaction framework. Private key 1 in wallet A controls UTXO1 and UTXO2, private key 2 controls UTXO3. A extracts 3 BTC from her bitcoin wallet, sends 2.5 BTC to B, and sends back 0.49 BTC to herself. The 0.1 BTC difference is served as transaction fee collected by the miner who confirmed this transaction.

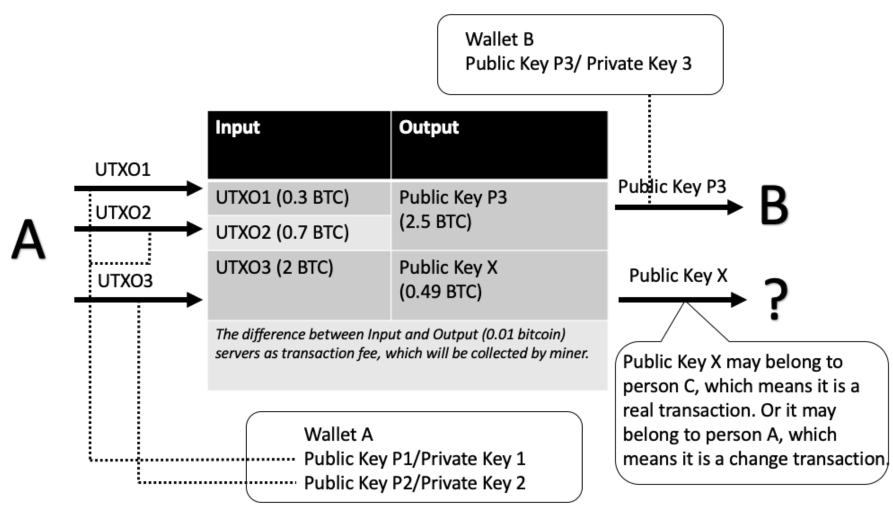

Figure 2: A simplified transaction framework II This figure shows a simplified transaction framework. Private key 1 in wallet A controls UTXO1 and UTXO2, private key 2 controls UTXO3. A extracts 3 BTC from her bitcoin wallet, sends 2.5 BTC to B. In this transaction, I do not know the identity of the person who receive the 0.49 BTC. It could be a new address created by A to receive the change, or it could be owned by a third person. The 0.1 BTC difference is served as transaction fee collected by the miner who confirmed this transaction.

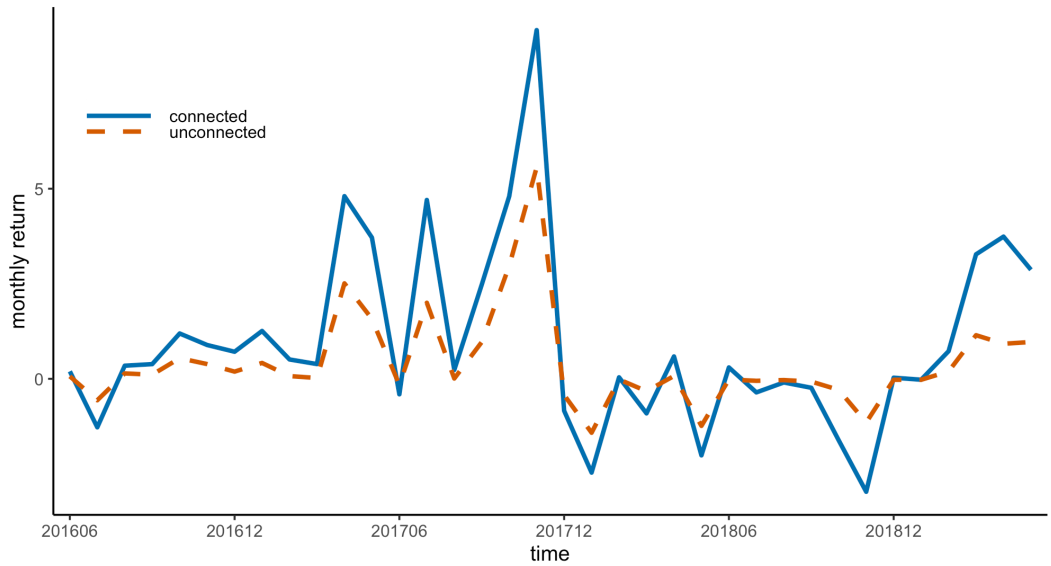

Figure 3: Median monthly return in connected and unconnected groups, M=10 In this figure, addresses are divided into connected group and unconnected group based on threshold $M = 10$. Then I calculate the median monthly return of all the addresses in each group for every month. The connected group has higher profitability than unconnected group when the whole bitcoin market is experiencing positive returns, but also suffers lower profitability when the whole market is experiencing negative returns.

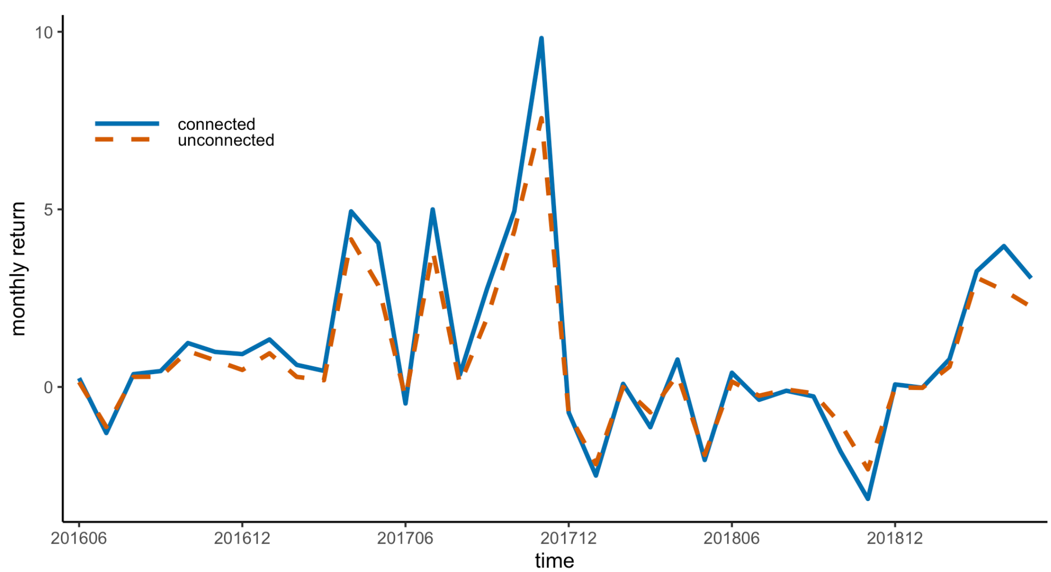

Figure 4: Median monthly return in connected and unconnected groups, M=30 In this figure, addresses are divided into connected group and unconnected group based on threshold $M = 30$. Then I calculate the median monthly return of all the addresses in each group for every month. The connected group has higher profitability than unconnected group when the whole bitcoin market is experiencing positive returns, but also suffers lower profitability when the whole market is experiencing negative returns.

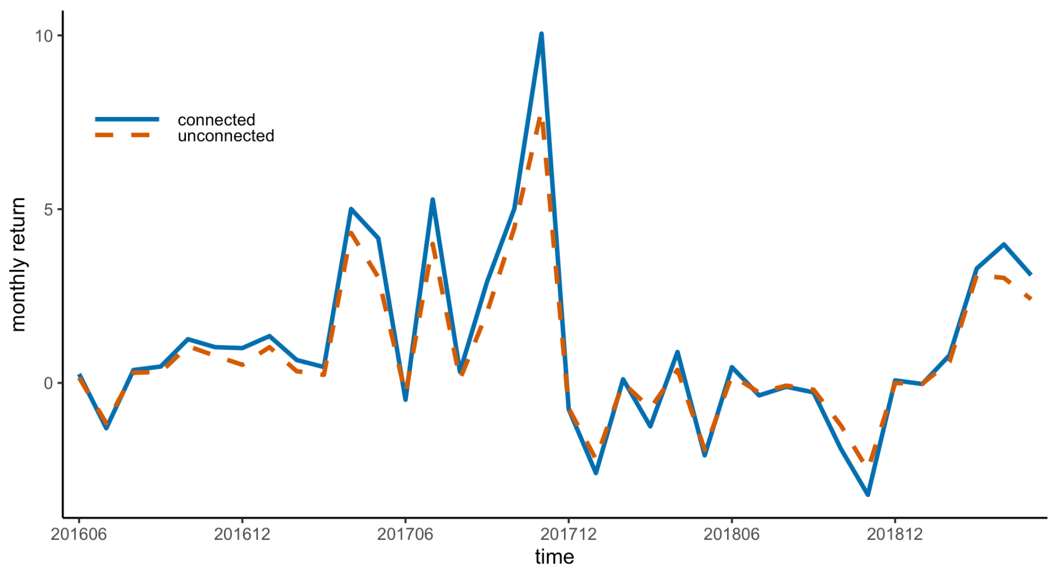

Figure 5: Median monthly return in connected and unconnected groups, M=50 In this figure, addresses are divided into connected group and unconnected group based on threshold $M = 50$. Then I calculate the median monthly return of all the addresses in each group for every month. The connected group has higher profitability than unconnected group when the whole bitcoin market is experiencing positive returns, but also suffers lower profitability when the whole market is experiencing negative returns.

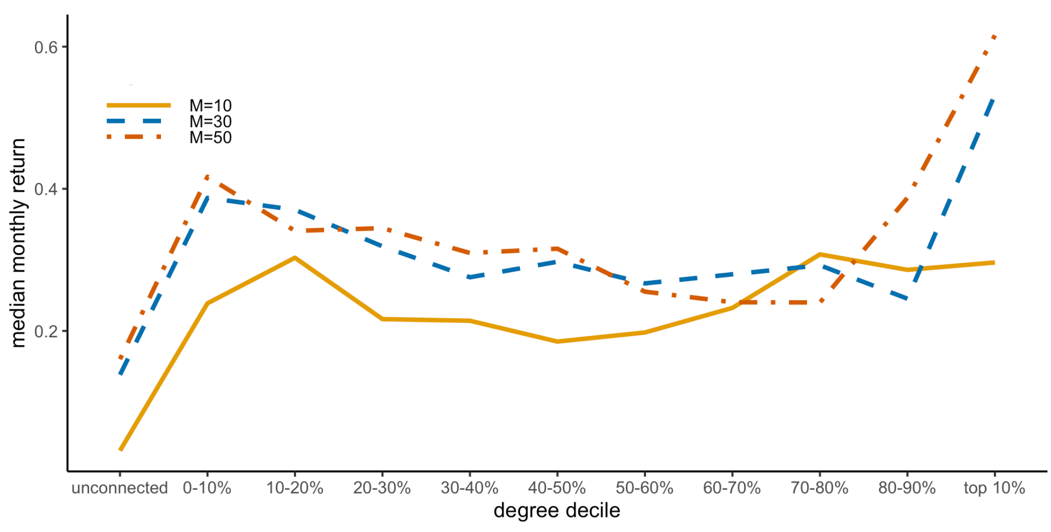

Figure 6: Median monthly return in connected and unconnected groups In this figure, I further divide the addresses in the connected group into 10 subgroups based on their degree values. I calculate the median monthly returns of each group over the full time period and compared them with the median return of the unconnected group. This figure shows connected subgroups always have higher returns no matter which threshold M I choose.

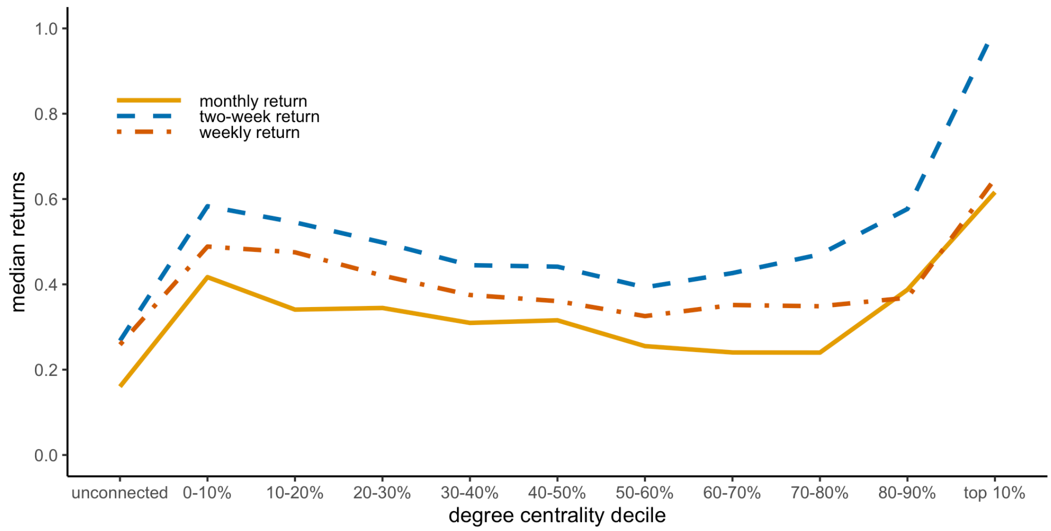

Figure 7: Median returns in groups with different degree centrality In this figure, I divide the addresses in the connected group into 10 subgroups based on their degree centrality values. I calculate the median monthly returns of each group over the full time period and compared them with the median return of the unconnected group. This figure shows connected subgroups always have higher returns compared with the unconnected group. Also there is another return jump around the top 20%. These results are consistent no matter which return data I use.

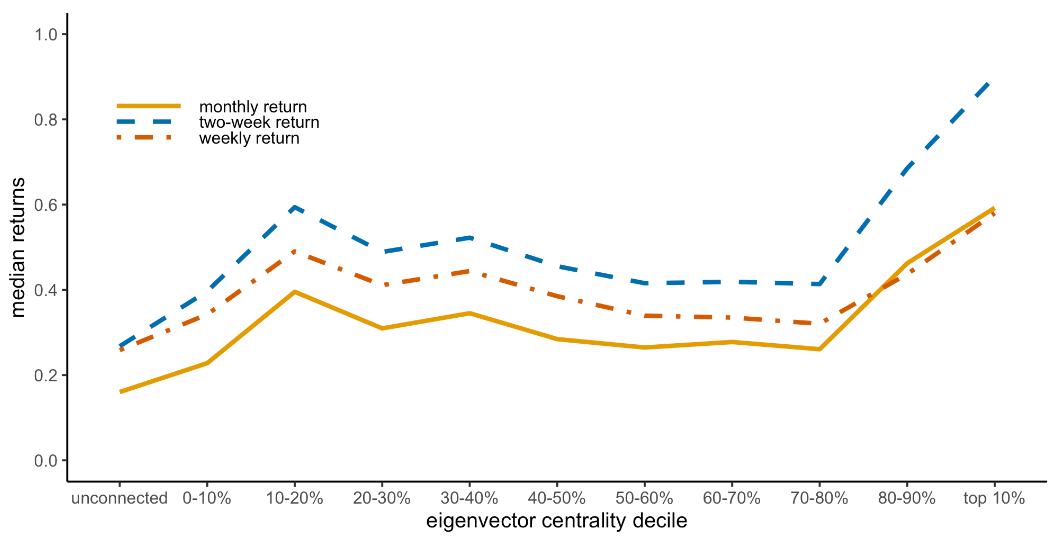

Figure 8: Median returns in groups with different eigenvector centrality In this figure, I divide the addresses in the connected group into 10 subgroups based on their eigenvector centrality values. I calculate the median monthly returns of each group over the full time period and compared them with the median return of the unconnected group. This figure shows connected subgroups always have higher returns compared with the unconnected group. Also there is another return jump around the top 20%. These results are consistent no matter which return data I use.

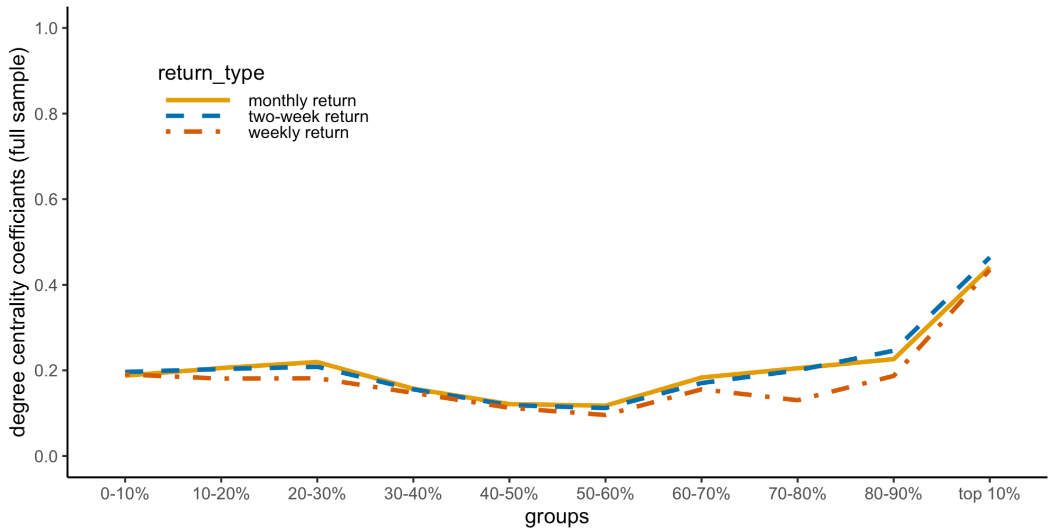

Figure 9: Regression coefficients for returns on groups with different degree centralities This figure shows the coefficients in the regression for returns on groups with different degree centralities. The full sample is used in this regression. Three different returns: monthly returns, two-week returns and weekly returns are used to represent the return data. If the coefficient is not significant, then it will be replaced with 0.

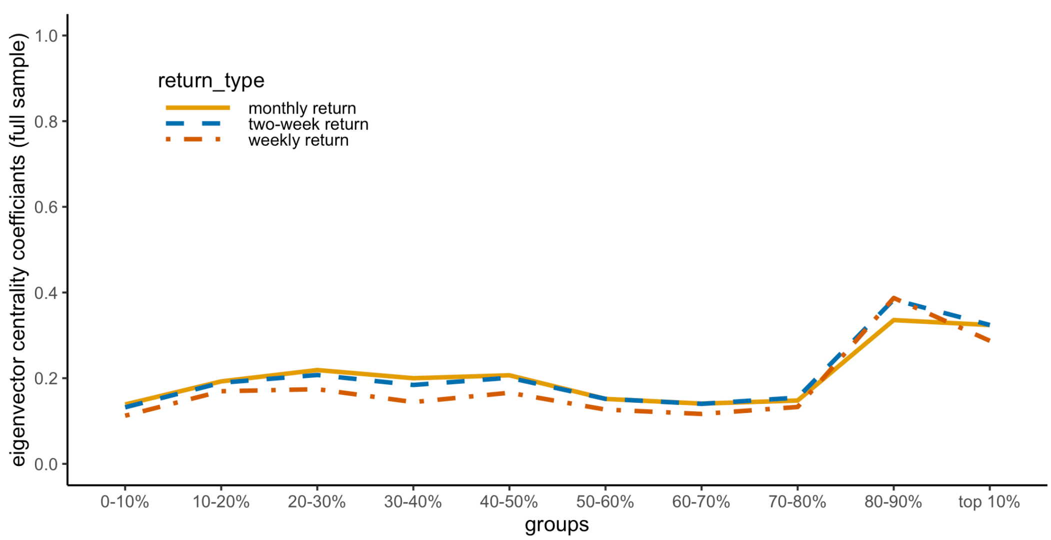

Figure 10: Regression coefficients for returns on groups with different eigenvector centralities This figure shows the coefficients in the regression for returns on groups with different eigenvector centralities. The full sample is used in this regression. Three different returns: monthly returns, two-week returns and weekly returns are used to represent the return data. If the coefficient is not significant, then it will be replaced with 0.

Table 1: The structure of a bitcoin block

Field	Description
Block Size	The size of the block
BH: Version	A version number to track software/protocol upgrades
BH: Previous Block Hash	A reference to the hash of the previous block in the chain
BH: Merkle Root	A hash of the root of the merkle tree of this block's transactions
BH: Timestamp	The approximate creation time of this block
BH: Target	The Proof-of-Work algorithm target for this block
BH: Nonce	A counter used for the Proof-of-Work algorithm
Transaction Counter	The number of transactions included in this block
Transactions	The transactions data

This table is adopted from Table 9-1 and Table 9-2 in Antonopoulos (2017).

Table 2: The structure of a bitcoin transaction

Field	Description
Input: Transaction Hash	Pointer to the previous transaction that contains the UTXO
Input: Output Index	The location of the UTXO in the previous transaction
Input: Unlocking Script	The script to fullfill the condition set by the UTXO locking script
Output: Amount	The amount of bitcoin that should be sent to the output address
Output: Locking Script	The script that set the condition to spend the bitcoin

This table is adopted from Table 6-1 and Table 6-2 in Antonopoulos (2017).

Table 3: Number of unique addresses in each period under different criteria

Appear Times	Period 1	(%)	Period 2	(%)	Period 3	(%)
$\geq$1	116,796,653	100	139,381,552	100	121,402,453	100
$\geq$2	63,967,311	54.77	74,344,299	53.34	65,959,905	54.33
$\geq$3	7,511,554	6.43	10,481,970	7.52	7,979,337	6.57
$\geq$5	4,135,734	3.54	5,241,400	3.76	4,143,966	3.41
$\geq$10	2,144,685	1.84	2,206,217	1.58	2,025,334	1.67
$\geq$30	733,001	0.63	574,290	0.41	637,315	0.52
$\geq$50	418,306	0.36	308,356	0.22	354,599	0.29

This table shows how many unique transaction addresses are recorded in the bitcoin blockchain in each period. Period 1 represents Jun 2016-May 2017, period 2 represents Jun 2017-May 2018, and period 3 represents Jun 2018-May 2019. Appear times $\geq$ 1 represents the whole data sample, appear times $\geq$ 2 means only includes the addresses that showed up at least 2 time during a given period. (%) shows the percent of addresses that showed up certain times in the dataset.

Table 4: Summary statistics for the bitcoin investor network in different periods

	Statistics	M=10	M=30	M=50
Period 1: Jun 2016 - May 2017	Number of links	696,588,746	203,830,771	108,242,916
	Average number of links	325	95	50
	Fraction of links	0.0152%	0.0044%	0.0023%
	Maximum number of links	1,092,437	351,897	193,247
Period 2: Jun 2017 - May 2018	Number of links	736,537,035	209,150,592	92,857,510
	Average number of links	334	95	42
	Fraction of links	0.0151%	0.0043%	0.0019%
	Maximum number of links	1,234,352	311,727	162,539
Period 3: Jun 2018 - May 2019	Number of links	764,699,333	256,855,523	145,163,621
	Average number of links	378	127	72
	Fraction of links	0.0187%	0.0063%	0.0036%
	Maximum number of links	1,120,383	332,901	168,979

Fraction of links is equal to average number of links divided by the number of all the potential links a node can connect. The number of potential links is 2144684 in period 1, 2206216 in period 2, 2025333 in period 3.

Table 5: Stability of the bitcoin investor network, M=50

Compared periods	period 1-period 2	period 2-period 3	period 1-period 3
Number of links in period 1, k_1	114,014,182		114,014,182
Number of links in period 2, k_2	98,628,776	98,628,776	
Number of links in period 3, k_3		150,934,887	150,934,887
Overlap links, y	10,077,079	12,961,294	5,821,282
$E_{random}[y]$	675	894	1,033
$y/E_{random}[y]$	14923.97	14499.97	5633.55
$E_{degree-adjusted}[y]$	118,705	108,186	181,658
$y/E_{degree-adjusted}[y]$	84.89	119.81	32.05

The table shows the stability of the bitcoin investor network across the three periods. The network is constructed when the connection threshold is set as $M = 50$. Period 1 is from Jun 2016 to May 2017, period 2 is from Jun 2017 to May 2018, and period 3 is from Jun 2018 to May 2019. Two agents are linked if they trade in the same direction for multiple time M. Overlap links, y, shows the number of intersecting links between two different periods. $E_{random}[y]$ is the expected number of intersecting links between two periods if the network is random. $E_{degree-adjusted}[y]$ is another measure of expected number of intersection links between two periods, but it takes the degree distributions of the original networks into consideration.

Table 6: Summary statistics for addresses

		Address Level		Market Level		
		Trade Volume	# of Trades	Trade Volume	# of Trades	Price Volatility
Full sample	Mean	-2.3802	1.5275	18.0377	15.8896	5.5178
	Median	-2.3585	1.3863	18.1413	15.9061	5.6333
	Std.dev.	3.0047	1.1155	0.4609	0.1967	1.4514
First Period :	Mean	-2.1212	1.7005	18.3343	15.8902	3.8420
Jun 2016 - May	Median	-1.7785	1.6094	18.3185	15.9290	4.2279
2017	Std.dev	3.2914	1.1770	0.1941	0.1255	0.9759
Second Period :	Mean	-2.2195	1.4115	18.2454	15.8465	6.7236
Jun 2017 - May	Median	-2.3026	1.3863	18.3838	15.8652	6.8567
2018	Std.dev	2.7703	1.0679	0.4498	0.2287	0.7613
Third Period :	Mean	-2.7835	1.4922	17.5510	15.9343	5.7734
Jun 2018 - May	Median	-2.8328	1.3863	17.5233	15.9192	5.6333
2019	Std.dev	2.9256	1.0878	0.1403	0.2032	0.7860

This table shows the summary statistics of regression variables. These variables include the address-level monthly trade volumes, address-level monthly number of trades, market-level monthly trade volume, market-level monthly number of trades and bitcoin price volatility. All the variables are in log-form.

Table 7: Connectedness and monthly return, M=50

	Full Sample	First Period	Second Period	Third Period
Address Level Variables				
Connectedness	0.2075	0.0606	0.4475	0.1665
	(>20)	(>20)	(>20)	(>20)
Trade Volume	-0.0543	-0.0764	-0.0995	0.0000
	(<-20)	(<-20)	(<-20)	(-0.1342)
# of Trades	-0.0486	0.0812	-0.0654	-0.1074
	(<-20)	(>20)	(<-20)	(<-20)
Market Level Variables				
Price Volatility	0.0520	0.1621	0.8592	-0.4957
	(>20)	(>20)	(>20)	(<-20)
Trade Volume	1.7569	0.3722	7.1016	6.8990
	(>20)	(>20)	(>20)	(>20)
# of Trades	2.2585	4.8400	-8.2764	0.5978
	(>20)	(>20)	(<-20)	(>20)
$\bar{R}^2$	0.1203	0.1404	0.2057	0.2464
obs.	26,242,261	8,046,594	9,314,410	8,881,202

This table display the results from the regression of monthly returns on connectedness, log address-level trade volume, log address-level number of trades, log bitcoin price volatility, log market-level trade volume, log market-level number of trade. I first examine the relationship between return and connectedness using the whole sample, then I further break it down to three different periods. All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table 8: Centrality and monthly returns in different groups, M=50

	Degree Centrality				Eigenvector Centrality			
	Full Sample	First Period	Second Period	Third Period	Full Sample	First Period	Second Period	Third Period
Group 1	0.1876	0.1011	0.4158	0.1462	0.1386	-0.0450	0.4078	0.1160
(0-10%)	(>20)	(>20)	(>20)	(>20)	(>20)	(-8.34)	(>20)	(>20)
Group 2	0.2054	0.0549	0.4071	0.1479	0.1925	0.1200	0.3243	0.1173
(10-20%)	(>20)	(9.36)	(>20)	(>20)	(>20)	(>20)	(>20)	(>20)
Group 3	0.2195	0.0619	0.3565	0.1728	0.2190	0.0269	0.2836	0.1204
(20-30%)	(>20)	(12.33)	(>20)	(>20)	(>20)	(5.21)	(>20)	(>20)
Group 4	0.1570	0.0516	0.3520	0.1661	0.1999	0.1017	0.2870	0.1807
(30-40%)	(>20)	(10.87)	(>20)	(>20)	(>20)	(>20)	(>20)	(>20)
Group 5	0.1212	0.0347	0.3663	0.1518	0.2068	0.0625	0.3672	0.1559
(40-50%)	(>20)	(7.96)	(>20)	(>20)	(>20)	(12.69)	(>20)	(>20)
Group 6	0.1174	-0.1156	0.4063	0.1325	0.1516	-0.0080	0.3900	0.1673
(50-60%)	(>20)	(<-20)	(>20)	(>20)	(>20)	(-1.77)	(>20)	(>20)
Group 7	0.1832	-0.0496	0.3573	0.1109	0.1407	-0.0013	0.4386	0.1383
(60-70%)	(>20)	(-10.36)	(>20)	(>20)	(>20)	(-0.32)	(>20)	(>20)
Group 8	0.2049	0.0410	0.3660	0.1794	0.1479	-0.0106	0.4685	0.1935
(70-80%)	(>20)	(8.19)	(>20)	(>20)	(>20)	(-2.46)	(>20)	(>20)
Group 9	0.2262	0.1139	0.5564	0.2044	0.3357	0.0832	0.8512	0.2632
(80-90%)	(>20)	(>20)	(>20)	(>20)	(>20)	(>20)	(>20)	(>20)
Group 10	0.4414	0.2264	0.9113	0.2424	0.3241	0.2391	0.6802	0.2175
(Top 10%)	(>20)	(>20)	(>20)	(>20)	(>20)	(>20)	(>20)	(>20)
$\bar{R}^2$	0.1203	0.1403	0.2057	0.2461	0.1203	0.1402	0.2058	0.2461
obs.	26,242,261	8,046,594	9,314,410	8,881,202	26,242,261	8,046,594	9,314,410	8,881,202

This table display the results from the regression of monthly returns on groups with different centrality, controlling log address-level trade volume, log address-level number of trades, log bitcoin price volatility, log market-level trade volume, log market-level number of trade. Centrality is measured using both degree centrality and eigenvector centrality. I first examine the relationship between return and centrality using the whole sample, then I further break it down to three different periods. All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table 9: Summary statistics for addresses in the central group

		Address Level				Market Level		
		Degree	Eigenvector	Trade Volume	# of Trades	Trade Volume	# of Trades	Price Volatility
Full sample	Mean	8.2901	-2.9096	-1.6762	2.9977	18.1530	15.8780	5.2001
	Median	8.3328	-3.2675	-1.8104	3.3322	18.2352	15.9061	5.3863
	Std.dev.	0.5060	1.5626	2.6589	1.0936	0.4340	0.1792	1.5694
Period 1:	Mean	8.2078	-3.1112	-1.9406	3.1050	18.3342	15.8841	3.7850
Jun 2016 - May	Median	8.3114	-3.3372	-2.0497	3.4012	18.3185	15.9290	4.2279
2017	Std.dev	0.5300	1.2572	3.0089	1.1582	0.1948	0.1249	0.9717
Period 2:	Mean	8.1577	-3.2153	-1.4318	2.9199	18.3116	15.8674	6.6600
Jun 2017 - May	Median	8.1798	-3.9291	-1.7204	3.3322	18.5099	15.8652	6.6652
2018	Std.dev	0.3097	1.4783	2.2195	1.06625	0.4251	0.2169	0.7607
Period 3:	Mean	8.4277	-2.6631	-1.5186	2.9805	17.5368	15.9090	5.7955
Jun 2018 - May	Median	8.4921	-2.3872	-1.9080	3.1781	17.4930	15.9017	5.7219
2019	Std.dev	0.7937	2.0028	2.6021	1.1228	0.1351	0.2009	0.7752

This table shows the summary statistics of regression variables. These variables include the degree centrality, eigenvector centrality, address-level monthly trade volumes, address-level monthly number of trades, market-level monthly trade volume, market-level monthly number of trades and bitcoin price volatility. All the variables are in log-form.

Table 10: Returns and centrality in the central group

	Full Sample	First Period	Second Period	Third Period
Monthly Return				
Degree Centrality	-0.0224	-0.1679	-0.5186	-0.0895
	(-1.27)	(-4.77)	(-7.42)	(-9.34)
Eigenvector Centrality	0.5251	0.4887	1.8362	0.1810
	(12.91)	(6.47)	(11.76)	(7.75)
Degree × Eigen	-0.0673	-0.0529	-0.2320	-0.0196
	(-13.96)	(-5.85)	(-12.11)	(-7.23)
$\bar{R}^2$	0.2336	0.2834	0.3369	0.3545
Two-week Return				
Degree Centrality	-0.0737	-0.2836	-0.2964	-0.2027
	(-3.41)	(-7.14)	(-3.70)	(-14.79)
Eigenvector Centrality	0.6624	0.8449	2.1017	0.3679
	(13.43)	(9.81)	(11.82)	(11.13)
Degree × Eigen	-0.0878	-0.1010	-0.2761	-0.0402
	(-15.02)	(-9.77)	(-12.68)	(-10.49)
$\bar{R}^2$	0.2184	0.2703	0.3221	0.2406
Weekly Return				
Degree Centrality	-0.0387	-0.2021	0.0116	-0.1422
	(-1.76)	(-4.54)	(0.14)	(-9.42)
Eigenvector Centrality	0.5314	0.5855	1.7357	0.2266
	(10.54)	(6.04)	(9.30)	(6.33)
Degree × Eigen	-0.0742	-0.0665	-0.2378	-0.0244
	(-12.41)	(-5.71)	(-10.41)	(-5.91)
$\bar{R}^2$	0.2038	0.2152	0.2394	0.1523
obs.	650,076	281,287	192,202	264,949

This table display the results from the regression of returns on degree centrality, eigenvector centrality and the interaction term, controlling address-level trade volume, address-level number of trades, bitcoin price volatility, market-level trade volume, market-level number of trade. I first examine the relationship between return and centrality in the central group using the whole sample, then I further break it down to three different periods. All the variables are in log form. All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

A Appendix

The results in the appendix section serve as a series of robustness checks of my main results reported in the paper. All these robustness checks can be summarized as two approaches: (1) in the first approach, instead of using $M = 50$ to establish addresses connections in the bitcoin investor network, I also choose $M = 10, 30$ to verify if the results will be consistent with the main results. (2) in the second approach, instead of using monthly returns, I use 2-week returns and weekly returns to verify if the conclusions in the main results still hold. In the following part, I iterate all the possible situations listed in these two approaches, and I find that all the results are consistent with the main results reported in the paper.

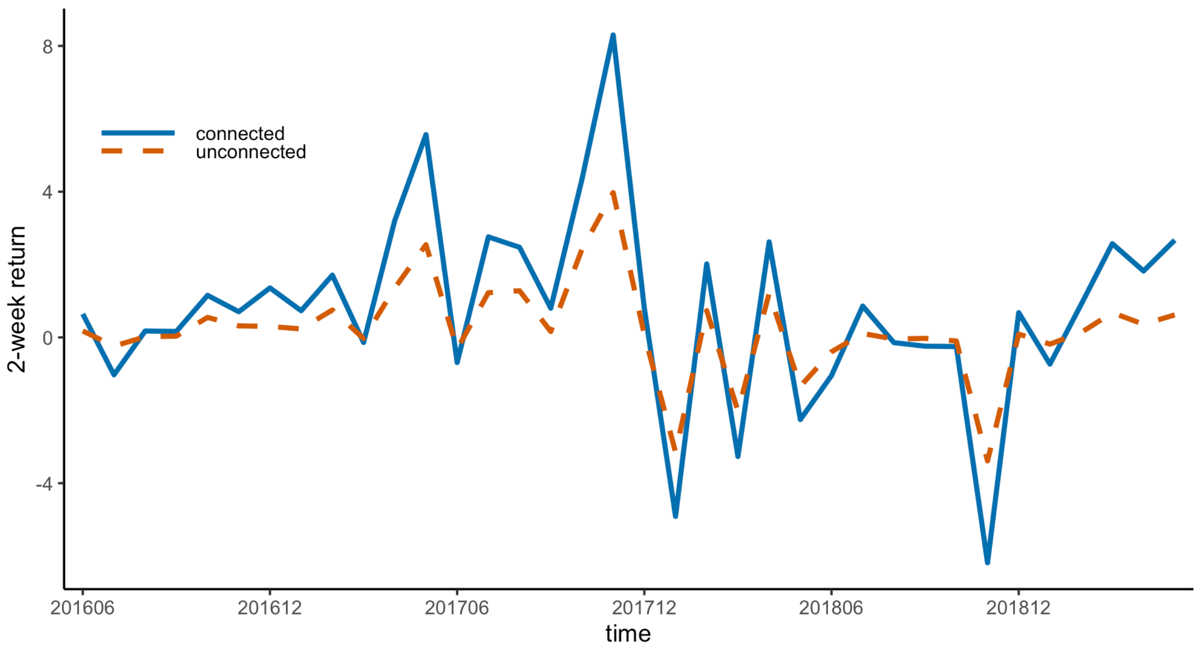

Figure A.1: Median 2-week return in connected and unconnected groups, M=10
In this figure, addresses are divided into connected group and unconnected group based on threshold $M = 10$. Then I calculate the median 2-week return of all the addresses in each group for every month. The connected group has higher profitability than unconnected group when the whole bitcoin market is experiencing positive returns, but also suffers lower profitability when the whole market is experiencing negative returns.

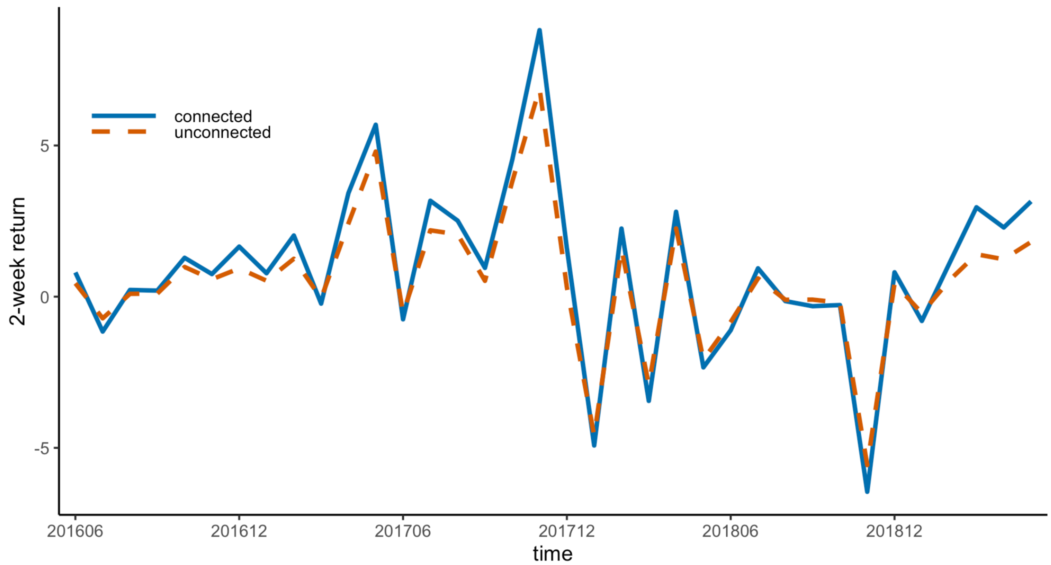

Figure A.2: Median 2-week return in connected and unconnected groups, M=30
In this figure, addresses are divided into connected group and unconnected group based on threshold $M = 30$. Then I calculate the median 2-week return of all the addresses in each group for every month. The connected group has higher profitability than unconnected group when the whole bitcoin market is experiencing positive returns, but also has slightly lower profitability when the whole market is experiencing negative returns.

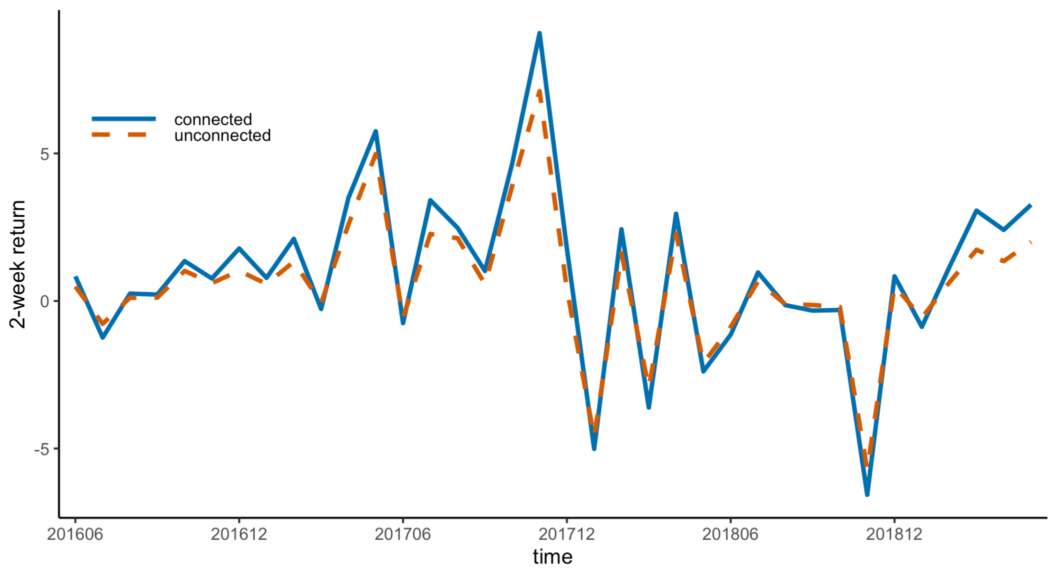

Figure A.3: Median 2-week return in connected and unconnected groups, M=50
In this figure, addresses are divided into connected group and unconnected group based on threshold $M = 50$. Then I calculate the median 2-week return of all the addresses in each group for every month. The connected group has higher profitability than unconnected group when the whole bitcoin market is experiencing positive returns, but also has slightly lower profitability when the whole market is experiencing negative returns.

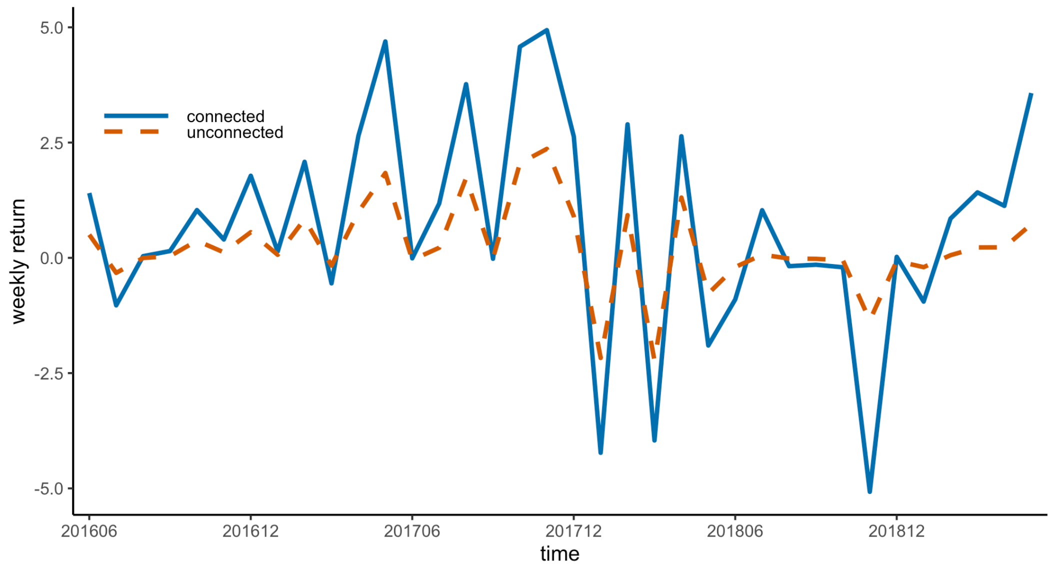

Figure A.4: Median weeky return in connected and unconnected groups, M=10
In this figure, addresses are divided into connected group and unconnected group based on threshold $M = 10$. Then I calculate the median weekly return of all the addresses in each group for every month. The connected group has higher profitability than unconnected group when the whole bitcoin market is experiencing positive returns, but also suffers lower profitability when the whole market is experiencing negative returns.

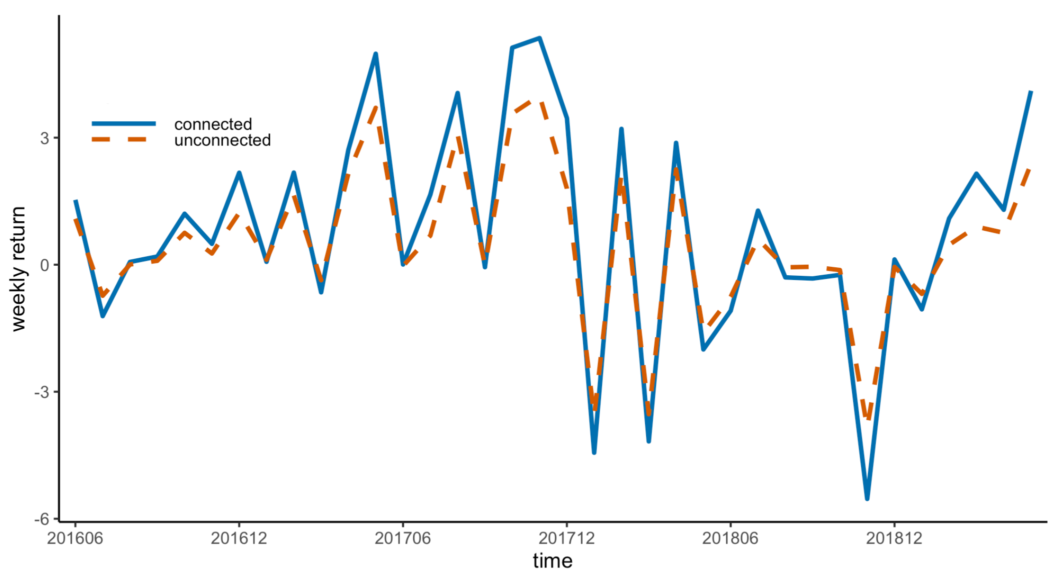

Figure A.5: Median weekly return in connected and unconnected groups, M=30
In this figure, addresses are divided into connected group and unconnected group based on threshold $M = 30$. Then I calculate the median weekly return of all the addresses in each group for every month. The connected group has higher profitability than unconnected group when the whole bitcoin market is experiencing positive returns, but also suffers lower profitability when the whole market is experiencing negative returns.

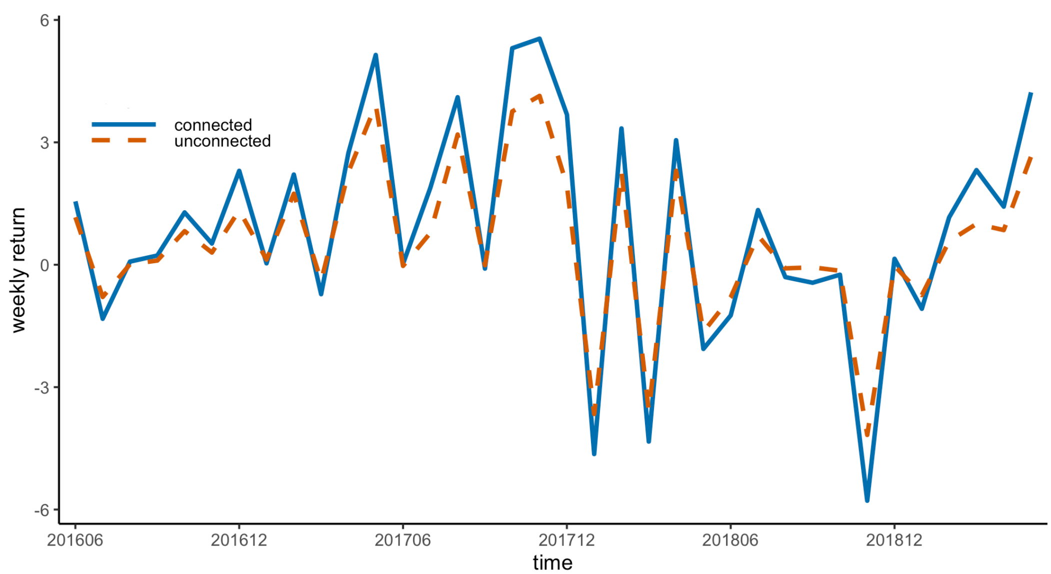

Figure A.6: Median weekly return in connected and unconnected groups, M=50
In this figure, addresses are divided into connected group and unconnected group based on threshold $M = 50$. Then I calculate the median weekly return of all the addresses in each group for every month. The connected group has higher profitability than unconnected group when the whole bitcoin market is experiencing positive returns, but also suffers lower profitability when the whole market is experiencing negative returns.

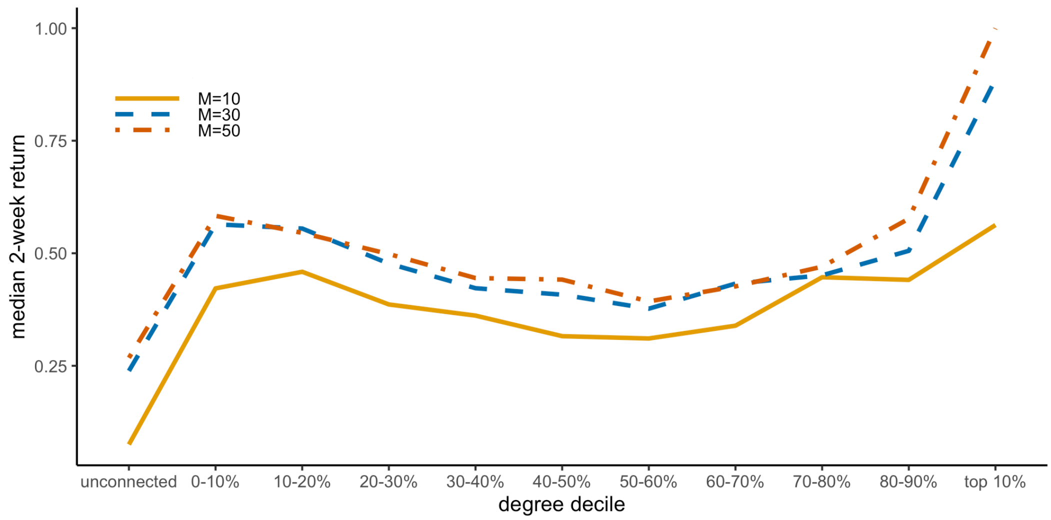

Figure A.7: Median 2-week return in connected and unconnected groups In this figure, I further divide the addresses in the connected group into 10 subgroups based on their degree values. I calculate the median 2-week returns of each group over the full time period and compared them with the median return of the unconnected group. This figure shows connected subgroups always have higher returns no matter which threshold M I choose.

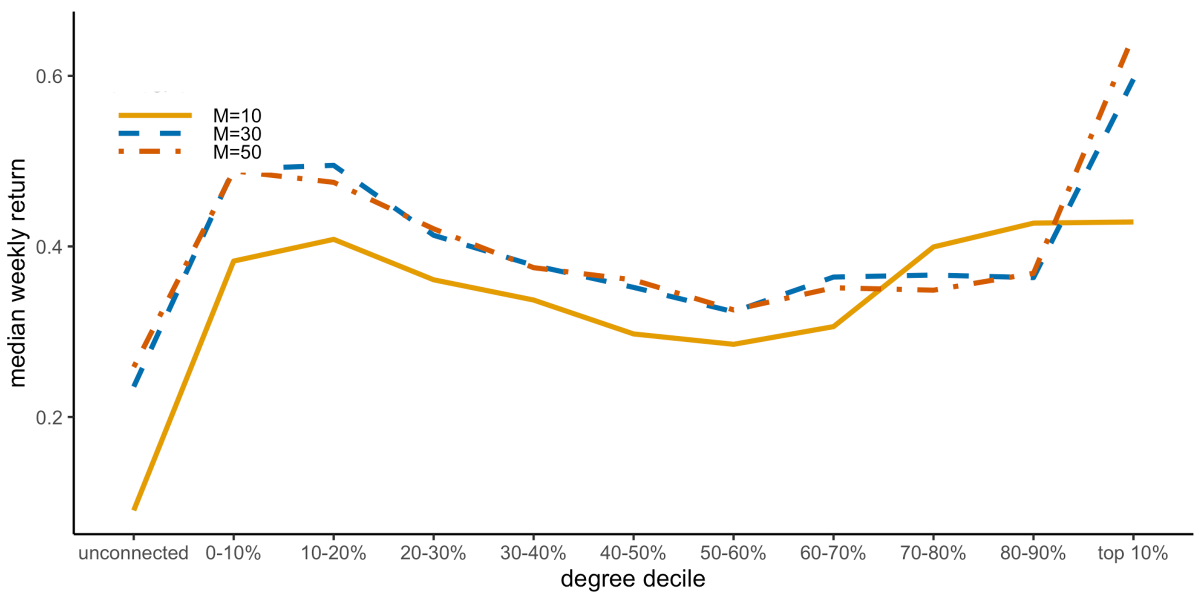

Figure A.8: Median weekly return in connected and unconnected groups In this figure, I further divide the addresses in the connected group into 10 subgroups based on their degree values. I calculate the median weekly returns of each group over the full time period and compared them with the median return of the unconnected group. This figure shows connected subgroups always have higher returns no matter which threshold M I choose.

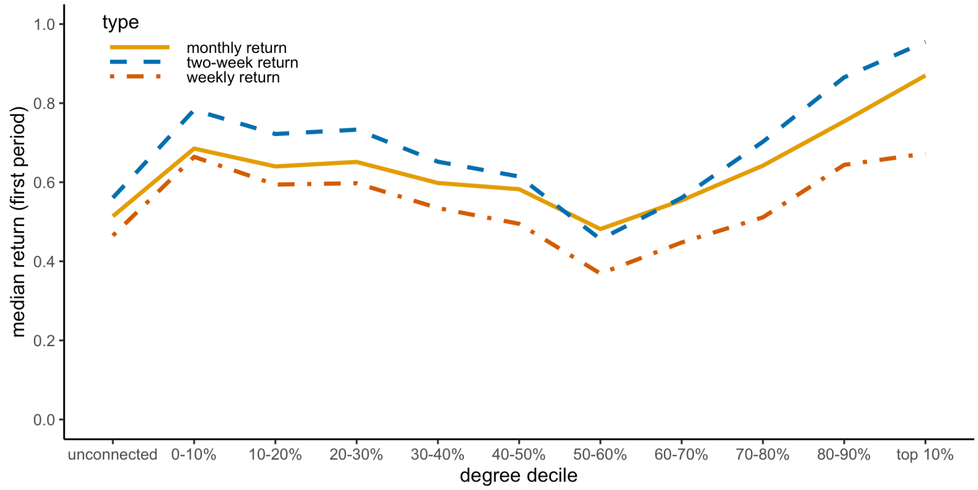

Figure A.9: Median returns in groups with degree centrality, first period In this figure, I divide the addresses in the connected group into 10 subgroups based on their degree centrality values. I calculate the median monthly returns of each group in the first period and compare them with the median return of the unconnected group. This figure shows connected subgroups always have higher returns than the unconnected. Also the top 20% has the highest return among all the subgroups. These results are consistent no matter which return data I use.

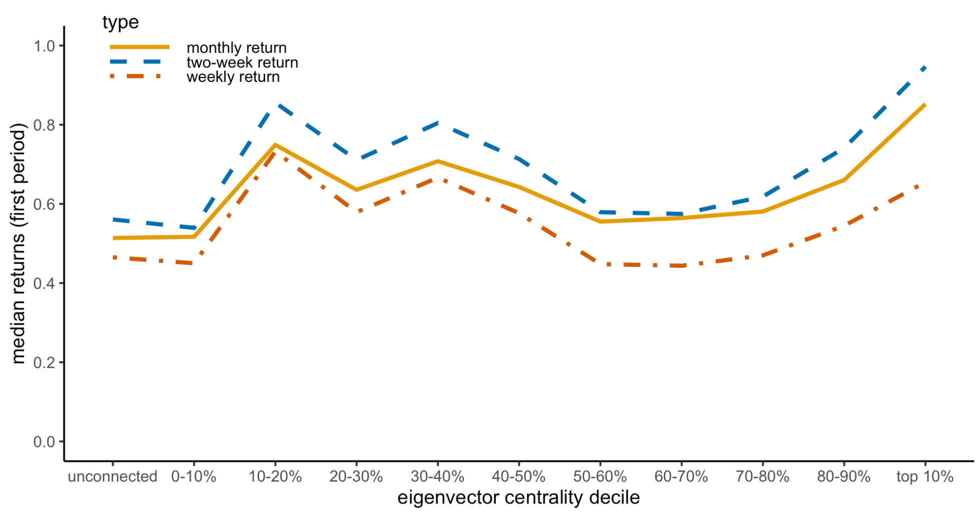

Figure A.10: Median returns in groups with eigenvector centrality, first period In this figure, I divide the addresses in the connected group into 10 subgroups based on their eigenvector centrality values. I calculate the median monthly returns of each group in the first period and compare them with the median return of the unconnected group. This figure shows connected subgroups always have higher returns than the unconnected. Also the top 20% has the highest return among all the subgroups.These results are consistent no matter which return data I use.

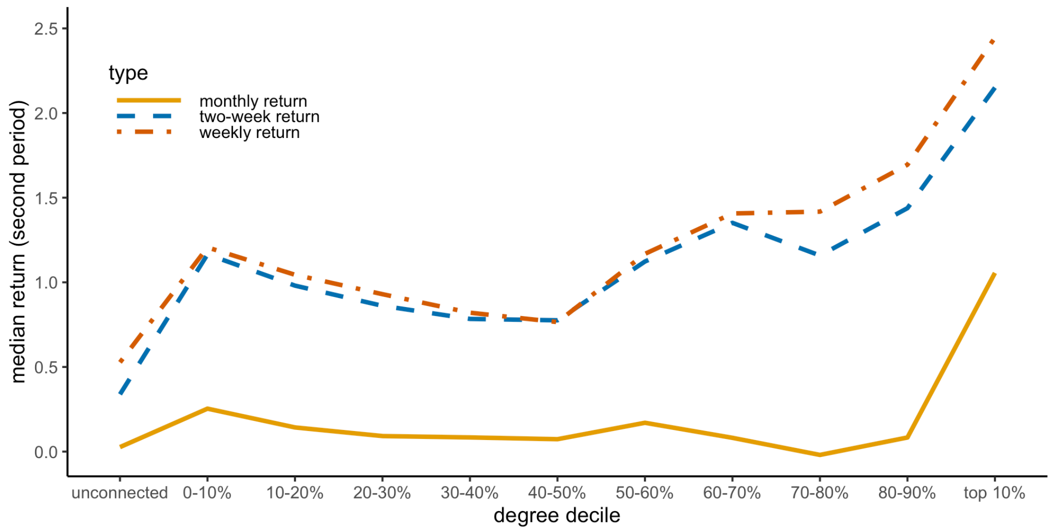

Figure A.11: Median returns in groups with degree centrality, second period In this figure, I divide the addresses in the connected group into 10 subgroups based on their degree centrality values. I calculate the median monthly returns of each group in the second period and compare them with the median return of the unconnected group. This figure shows connected subgroups always have higher returns compared with the unconnected group. Also there is another return jump around the top 20%. These results are consistent no matter which return data I use.

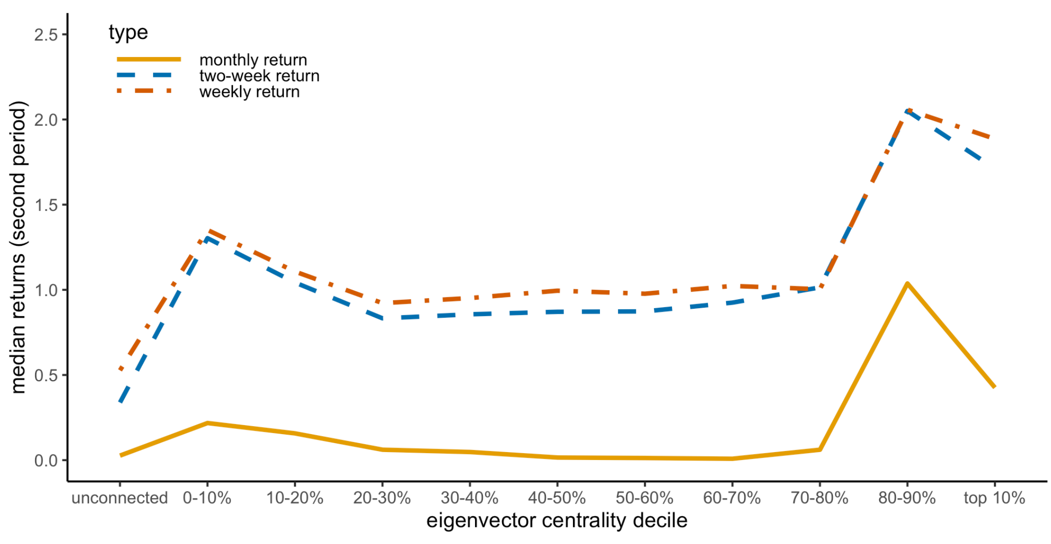

Figure A.12: Median returns in groups with eigenvector centrality, second period In this figure, I divide the addresses in the connected group into 10 subgroups based on their eigenvector centrality values. I calculate the median monthly returns of each group in the second period and compare them with the median return of the unconnected group. This figure shows connected subgroups always have higher returns compared with the unconnected group. Also there is another return jump around the top 20%. These results are consistent no matter which return data I use.

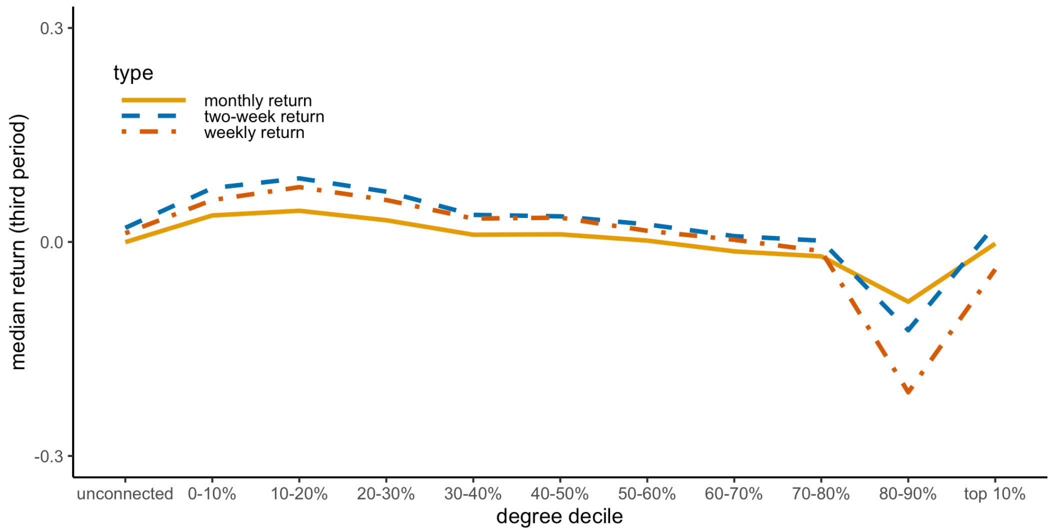

Figure A.13: Median returns in groups with degree centrality, third period In this figure, I divide the addresses in the connected group into 10 subgroups based on their degree centrality values. I calculate the median monthly returns of each group in the third period and compare them with the median return of the unconnected group. This figure shows, in the third period, the top 20% are having lower return than their peers. These results are consistent no matter which return data I use.

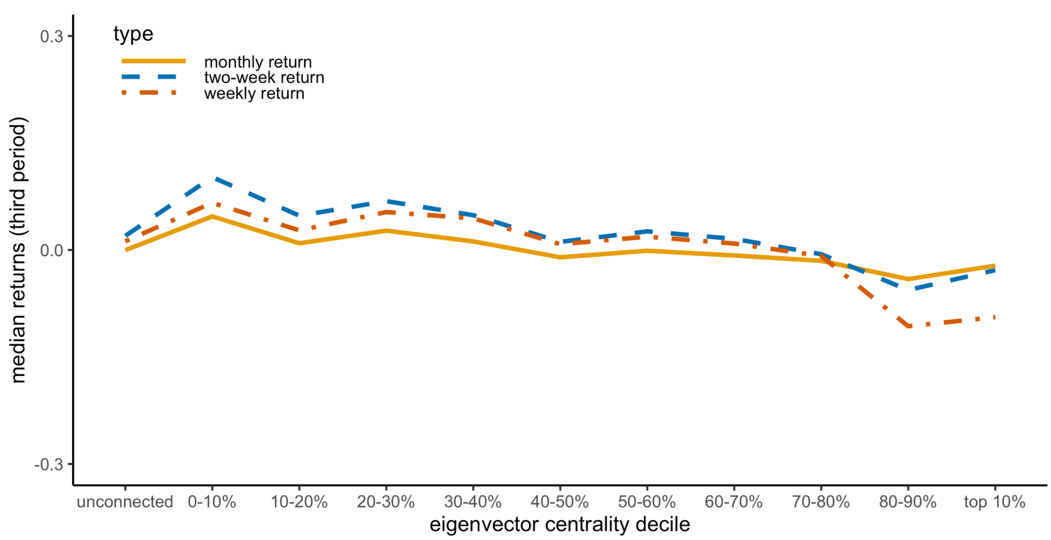

Figure A.14: Median returns in groups with eigenvector centrality, third period In this figure, I divide the addresses in the connected group into 10 subgroups based on their eigenvector centrality values. I calculate the median monthly returns of each group in the third period and compare them with the median return of the unconnected group. This figure shows, in the third period, the top 20% are having lower return than their peers. These results are consistent no matter which return data I use.

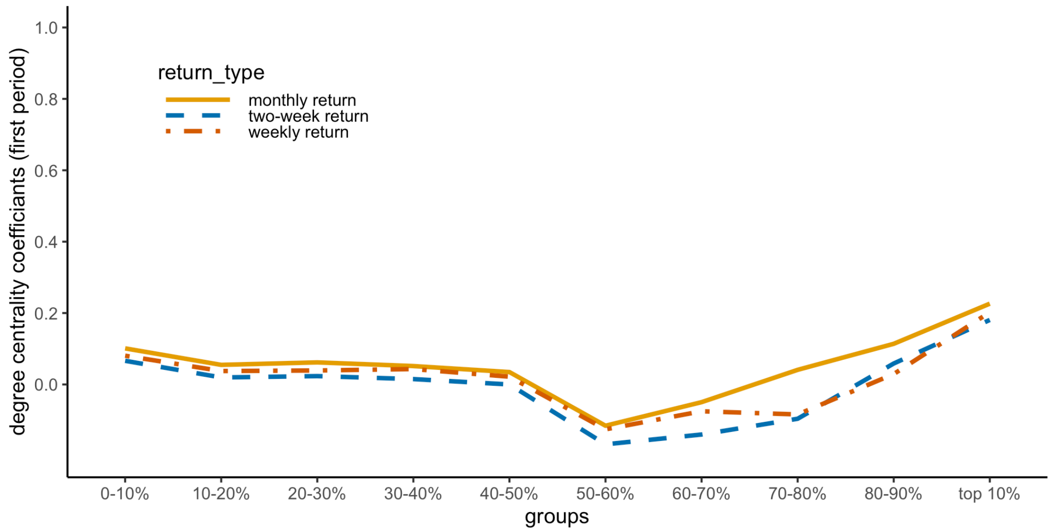

Figure A.15: Regression coefficients for returns on groups with different degree centralities, first period This figure shows the coefficients in the regression for returns on groups with different degree centralities. This regression focus on the first period. Three different returns: monthly returns, two-week returns and weekly returns are used to represent the return data. If the coefficient is not significant, then it will be replaced with 0.

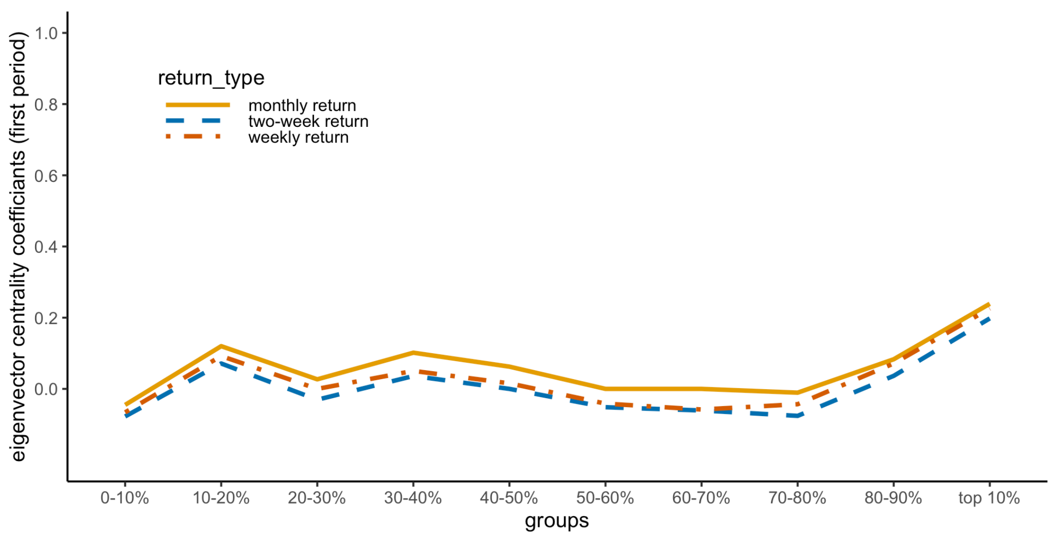

Figure A.16: Regression coefficients for returns on groups with different eigenvector centralities, first period This figure shows the coefficients in the regression for returns on groups with different eigenvector centralities. This regression focus the first period. Three different returns: monthly returns, two-week returns and weekly returns are used to represent the return data. If the coefficient is not significant, then it will be replaced with 0.

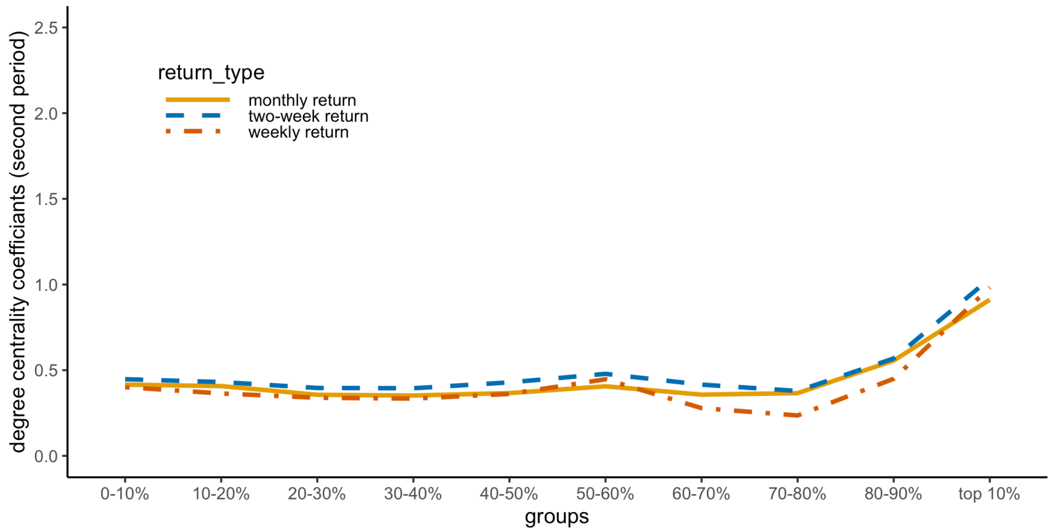

Figure A.17: Regression coefficients for returns on groups with different degree centralities, second period This figure shows the coefficients in the regression for returns on groups with different degree centralities. This regression focus on the second period. Three different returns: monthly returns, two-week returns and weekly returns are used to represent the return data. If the coefficient is not significant, then it will be replaced with 0.

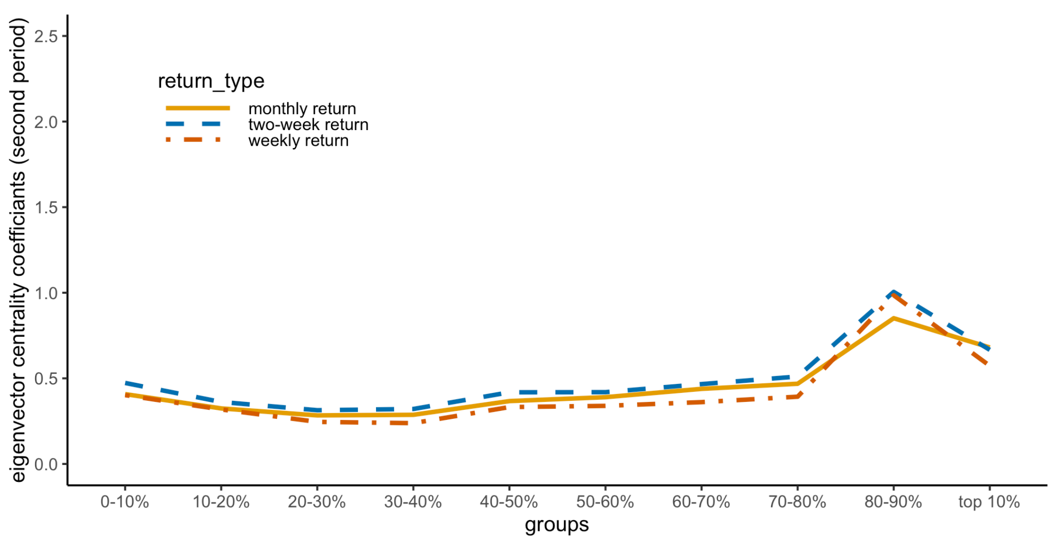

Figure A.18: Regression coefficients for returns on groups with different eigenvector centralities, second period This figure shows the coefficients in the regression for returns on groups with different eigenvector centralities. This regression focus the second period. Three different returns: monthly returns, two-week returns and weekly returns are used to represent the return data. If the coefficient is not significant, then it will be replaced with 0.

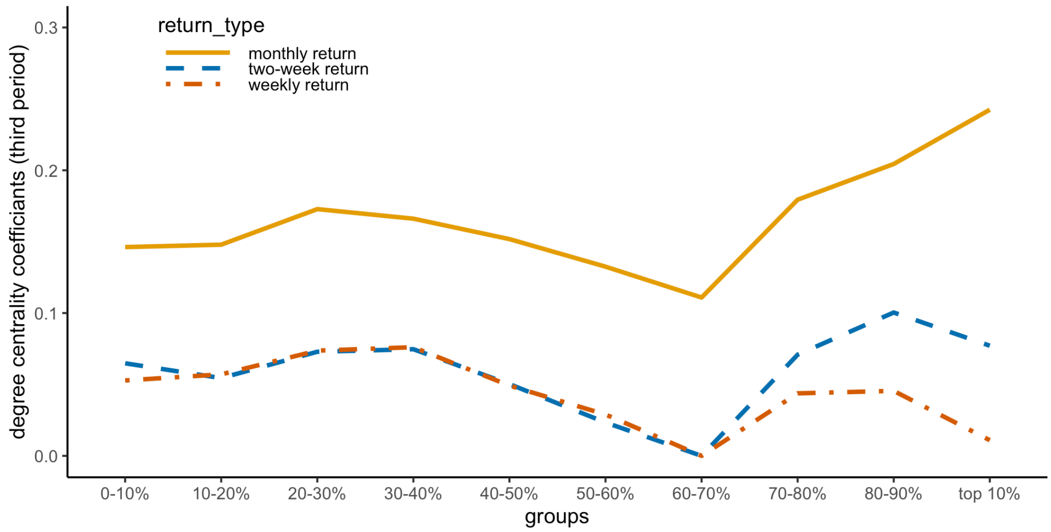

Figure A.19: Regression coefficients for returns on groups with different degree centralities, third period This figure shows the coefficients in the regression for returns on groups with different degree centralities. This regression focus on the third period. Three different returns: monthly returns, two-week returns and weekly returns are used to represent the return data. If the coefficient is not significant, then it will be replaced with 0.

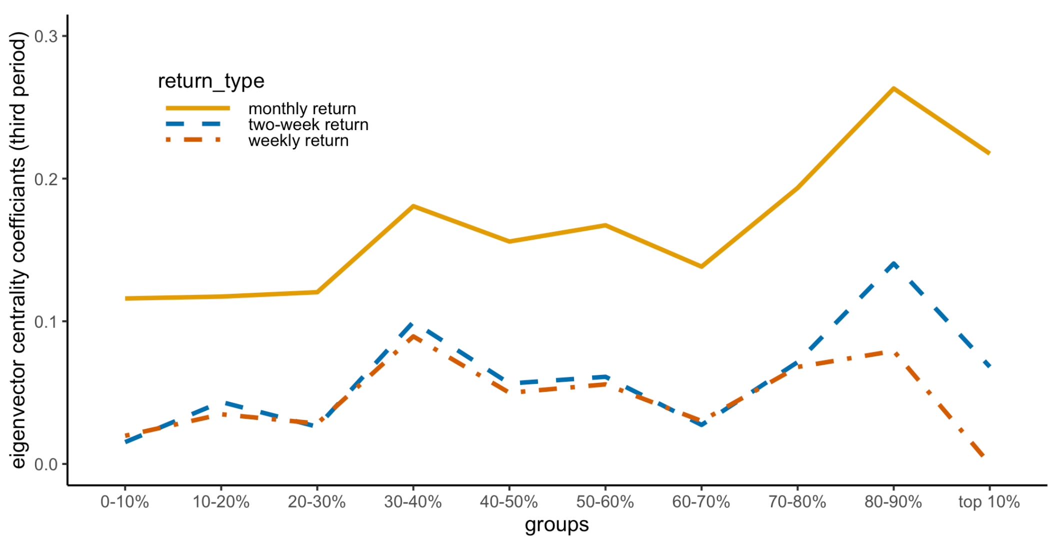

Figure A.20: Regression coefficients for returns on groups with different eigenvector centralities, third period This figure shows the coefficients in the regression for returns on groups with different eigenvector centralities. This regression focus the third period. Three different returns: monthly returns, two-week returns and weekly returns are used to represent the return data. If the coefficient is not significant, then it will be replaced with 0.

Table A.1: Stability of the bitcoin investor network, M=10

Compared periods	1-2	2-3	1-3
Number of links in period 1, k_1	702,360,012		702,360,012
Number of links in period 2, k_2	742,308,301	742,308,301	
Number of links in period 3, k_3		770,470,599	770,470,599
Overlap links, y	48,022,189	64,121,487	6,268,211
$E_{random}[y]$	31,306	34,342	32,494
$y/E_{random}[y]$	1533.95	1867.13	192.90
$E_{degree-adjusted}[y]$	4,377,357	2,700,984	4,543,429
$y/E_{degree-adjusted}[y]$	10.97	23.74	1.38

The table shows the stability of the bitcoin investor network across the three periods. The network is constructed when the connection threshold is set as $M = 10$. Period 1 is from Jun 2016 to May 2017, period 2 is from Jun 2017 to May 2018, and period 3 is from Jun 2018 to May 2019. Two agents are linked if they trade in the same direction for multiple time M. Overlap links, y, shows the number of intersecting links between two different periods. $E_{random}[y]$ is the expected number of intersecting links between two periods if the network is random. $E_{degree-adjusted}[y]$ is another measure of expected number of intersection links between two periods, but it takes the degree distributions of the original networks into consideration.

Table A.2: Stability of the bitcoin investor network, M=30

Compared periods	1-2	2-3	1-3
Number of links in period 1, k_1	209,602,037		209,602,037
Number of links in period 2, k_2	214,921,858	214,921,858	
Number of links in period 3, k_3		262,626,789	262,626,789
Overlap links, y	16,303,616	22,031,201	5,876,037
$E_{random}[y]$	2,705	3,389	3,305
$y/E_{random}[y]$	6027.26	6500.26	1777.72
$E_{degree-adjusted}[y]$	425,191	319,101	519,569
$y/E_{degree-adjusted}[y]$	38.34	69.04	11.31

The table shows the stability of the bitcoin investor network across the three periods. The network is constructed when the connection threshold is set as $M = 30$. Period 1 is from Jun 2016 to May 2017, period 2 is from Jun 2017 to May 2018, and period 3 is from Jun 2018 to May 2019. Two agents are linked if they trade in the same direction for multiple time M. Overlap links, y, shows the number of intersecting links between two different periods. $E_{random}[y]$ is the expected number of intersecting links between two periods if the network is random. $E_{degree-adjusted}[y]$ is another measure of expected number of intersection links between two periods, but it takes the degree distributions of the original networks into consideration.

Table A.3: Connectedness and monthly returns, M=30

	Full Sample	First Period	Second Period	Third Period
Address Level Variables				
Connectedness	0.2401	0.1150	0.4169	0.1959
	(>20)	(>20)	(>20)	(>20)
Trade Volume	-0.0533	-0.0749	-0.0985	0.0001
	(<-20)	(<-20)	(<-20)	(0.4553)
# of Trades	-0.0625	0.0691	-0.0787	-0.1194
	(<-20)	(>20)	(<-20)	(<-20)
Market Level Variables				
Price Volatility	0.0535	0.1645	0.8576	-0.4944
	(>20)	(>20)	(>20)	(<-20)
Trade Volume	1.7645	0.3756	7.0990	0.6902
	(>20)	(>20)	(>20)	(>20)
# of Trades	2.2483	4.8171	-8.2662	0.5964
	(>20)	(>20)	(<-20)	(>20)
$\bar{R}^2$	0.1207	0.1405	0.2059	0.2467
obs.	26,181,341	8,026,523	9,296,250	8,858,550

This table display the results from the regression of monthly returns on connectedness, and other control variables (log-form) . All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table A.4: Connectedness and monthly returns, M=10

	Full Sample	First Period	Second Period	Third Period
Address Level Variables				
Connectedness	0.4114	0.3683	0.4739	0.4132
	(>20)	(>20)	(>20)	(>20)
Trade Volume	-0.0507	-0.0686	-0.0978	0.0017
	(<-20)	(<-20)	(<-20)	(6.788)
# of Trades	-0.0580	0.0541	-0.0590	-0.1155
	(<-20)	(>20)	(<-20)	(<-20)
Market Level Variables				
Price Volatility	0.0502	0.1665	0.8589	-0.4939
	(>20)	(>20)	(>20)	(<-20)
Trade Volume	1.7690	0.3760	7.0990	0.6902
	(>20)	(>20)	(>20)	(>20)
# of Trades	2.2602	4.7901	-8.2644	0.6226
	(>20)	(>20)	(<-20)	(>20)
$\bar{R}^2$	0.1223	0.1441	0.2058	0.2498
obs.	25,924,373	7,954,317	9,199,468	8,770,535

This table display the results from the regression of monthly returns on connectedness, and other control variables (log-form) . All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table A.5: Connectedness and 2-week returns, M=50

	Full Sample	First Period	Second Period	Third Period
Address Level Variables				
Connectedness	0.2082	0.0074	0.4934	0.0610
	(>20)	(3.295)	(>20)	(>20)
Trade Volume	-0.0585	-0.0807	-0.1129	-0.0019
	(<-20)	(<-20)	(<-20)	(-5.429)
# of Trades	-0.0536	0.1097	-0.0887	-0.0610
	(<-20)	(>20)	(<-20)	(<-20)
Market Level Variables				
Price Volatility	0.1030	0.5761	2.0104	-0.9339
	(>20)	(>20)	(>20)	(<-20)
Trade Volume	1.9569	0.0848	9.1081	6.0760
	(>20)	(>20)	(>20)	(>20)
# of Trades	2.3324	2.7412	-11.5520	0.8139
	(>20)	(>20)	(<-20)	(>20)
$\bar{R}^2$	0.0906	0.1276	0.1684	0.1536
obs.	26,242,261	8,046,594	9,314,410	8,881,202

This table display the results from the regression of 2-week returns on connectedness, and other control variables (log-form) . All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table A.6: Connectedness and 2-week returns, M=30

	Full Sample	First Period	Second Period	Third Period
Address Level Variables				
Connectedness	0.2390	0.0769	0.4647	0.0984
	(>20)	(>20)	(>20)	(>20)
Trade Volume	-0.0575	-0.0792	-0.1118	-0.0017
	(<-20)	(<-20)	(<-20)	(-4.926)
# of Trades	-0.0671	0.0959	-0.1039	-0.0707
	(<-20)	(>20)	(<-20)	(<-20)
Market Level Variables				
Price Volatility	0.1048	0.5787	2.0086	-0.9324
	(>20)	(>20)	(>20)	(<-20)
Trade Volume	1.9643	0.0888	9.1059	6.0769
	(>20)	(>20)	(>20)	(>20)
# of Trades	2.3225	2.7158	-11.5408	0.8127
	(>20)	(>20)	(<-20)	(>20)
$\bar{R}^2$	0.0908	0.1274	0.1685	0.1534
obs.	26,181,341	8,026,523	9,296,250	8,858,550

This table display the results from the regression of 2-week returns on connectedness, and other control variables (log-form) . All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table A.7: Connectedness and 2-week returns, M=10

	Full Sample	First Period	Second Period	Third Period
Address Level Variables				
Connectedness	0.4105	0.3628	0.5330	0.3055
	(>20)	(>20)	(>20)	(>20)
Trade Volume	-0.0550	-0.0723	-0.1113	-0.0005
	(<-20)	(<-20)	(<-20)	(-1.346)
# of Trades	-0.0624	0.0736	-0.0791	-0.0761
	(<-20)	(>20)	(<-20)	(<-20)
Market Level Variables				
Price Volatility	0.1020	0.5817	2.0102	-0.9300
	(>20)	(>20)	(>20)	(<-20)
Trade Volume	1.9695	0.0932	9.1099	6.0760
	(>20)	(>20)	(>20)	(>20)
# of Trades	2.3344	2.6783	-11.5440	0.8327
	(>20)	(>20)	(<-20)	(>20)
$\bar{R}^2$	0.0917	0.1297	0.1683	0.1540
obs.	25,924,373	7,954,317	9,199,468	8,770,535

This table display the results from the regression of 2-week returns on connectedness, and other control variables (log-form). All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table A.8: Connectedness and weekly returns, M=50

	Full Sample	First Period	Second Period	Third Period
Address Level Variables				
Connectedness	0.1827	0.0277	0.4180	0.0472
	(>20)	(10.53)	(>20)	(15.11)
Trade Volume	-0.0609	-0.0698	-0.1242	-0.0065
	(<-20)	(<-20)	(<-20)	(-15.13)
# of Trades	-0.0345	0.0619	-0.0488	-0.0348
	(<-20)	(>20)	(<-20)	(<-20)
Market Level Variables				
Price Volatility	0.1529	0.7357	1.5492	-0.4930
	(>20)	(>20)	(>20)	(<-20)
Trade Volume	1.7971	0.1113	6.7777	3.4794
	(>20)	(>20)	(>20)	(>20)
# of Trades	2.4357	1.0374	-7.3683	2.2461
	(>20)	(>20)	(<-20)	(>20)
$\bar{R}^2$	0.0645	0.0844	0.0899	0.0749
obs.	26,242,261	8,046,594	9,314,410	8,881,202

This table display the results from the regression of weekly returns on connectedness, and other control variables (log-form). All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table A.9: Connectedness and weekly returns, M=30

	Full Sample	First Period	Second Period	Third Period
Address Level Variables				
Connectedness	0.2137	0.0873	0.4147	0.0789
	(>20)	(>20)	(>20)	(>20)
Trade Volume	-0.0600	-0.0684	-0.1230	-0.0064
	(<-20)	(<-20)	(<-20)	(-14.80)
# of Trades	-0.0469	0.0493	-0.0656	-0.0428
	(<-20)	(>20)	(<-20)	(<-20)
Market Level Variables				
Price Volatility	0.1544	0.7377	1.5470	-0.4920
	(>20)	(>20)	(>20)	(<-20)
Trade Volume	1.8034	0.1144	6.7752	3.4814
	(>20)	(>20)	(>20)	(>20)
# of Trades	2.4275	1.0163	-7.3555	2.2434
	(>20)	(>20)	(<-20)	(>20)
$\bar{R}^2$	0.0646	0.0843	0.0901	0.0748
obs.	26,181,341	8,026,523	9,296,250	8,858,550

This table display the results from the regression of weekly returns on connectedness, and other control variables (log-form). All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table A.10: Connectedness and weekly returns, M=10

	Full Sample	First Period	Second Period	Third Period
Address Level Variables				
Connectedness	0.3715	0.3409	0.5002	0.2537
	(>20)	(>20)	(>20)	(>20)
Trade Volume	-0.0579	-0.0623	-0.1226	-0.0054
	(<-20)	(<-20)	(<-20)	(-12.44)
# of Trades	-0.0428	0.0307	-0.0450	-0.0472
	(<-20)	(>20)	(<-20)	(<-20)
Market Level Variables				
Price Volatility	0.1517	0.7389	1.5486	-0.4905
	(>20)	(>20)	(>20)	(<-20)
Trade Volume	1.8077	0.1147	6.7830	3.4866
	(>20)	(>20)	(>20)	(>20)
# of Trades	2.4391	0.9927	-7.3640	2.2547
	(>20)	(>20)	(<-20)	(>20)
$\bar{R}^2$	0.0651	0.0857	0.0901	0.0749
obs.	25,924,373	7,954,317	9,199,468	8,770,535

This table display the results from the regression of weekly returns on connectedness, and other control variables (log-form). All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table A.11: Centrality and two-week returns in different groups, M=50

	Degree Centrality				Eigenvector Centrality			
	Full Sample	First Period	Second Period	Third Period	Full Sample	First Period	Second Period	Third Period
Group 1	0.1962	0.0663	0.4474	0.0648	0.1320	-0.0774	0.4731	0.0153
(0-10%)	(>20)	(14.00)	(>20)	(10.89)	(>20)	(-12.35)	(>20)	(2.25)
Group 2	0.2029	0.0196	0.4308	0.0545	0.1894	0.0713	0.3618	0.0435
(10-20%)	(>20)	(2.85)	(>20)	(7.05)	(>20)	(12.28)	(>20)	(6.30)
Group 3	0.2083	0.0234	0.3962	0.0729	0.2075	-0.0304	0.3138	0.0261
(20-30%)	(>20)	(4.00)	(>20)	(11.29)	(>20)	(-5.09)	(>20)	(3.84)
Group 4	0.1552	0.0150	0.3946	0.0747	0.1842	0.0362	0.3210	0.0995
(30-40%)	(>20)	(2.68)	(>20)	(11.01)	(>20)	(6.14)	(>20)	(14.49)
Group 5	0.1186	0.0020	0.4288	0.0503	0.2010	-0.0093	0.4184	0.0564
(40-50%)	(>20)	(0.392)	(>20)	(7.56)	(>20)	(-1.61)	(>20)	(8.19)
Group 6	0.1119	-0.1677	0.4786	0.0234	0.1517	-0.0515	0.4193	0.0611
(50-60%)	(>20)	(<-20)	(>20)	(3.44)	(>20)	(-9.48)	(>20)	(9.20)
Group 7	0.1703	-0.1402	0.4158	-0.0019	0.1401	-0.0606	0.4663	0.0274
(60-70%)	(>20)	(<-20)	(>20)	(-0.29)	(>20)	(-11.88)	(>20)	(4.17)
Group 8	0.1999	-0.0961	0.3789	0.0788	0.1547	-0.0758	0.5106	0.0714
(70-80%)	(>20)	(-16.27)	(>20)	(10.89)	(>20)	(-14.78)	(>20)	(11.48)
Group 9	0.2460	0.0587	0.5684	0.1004	0.3834	0.0366	1.0049	0.1405
(80-90%)	(>20)	(11.58)	(>20)	(15.18)	(>20)	(7.80)	(>20)	(>20)
Group 10	0.4639	0.1808	1.0336	0.0771	0.3242	0.1980	0.6664	0.0681
(Top 10%)	(>20)	(>20)	(>20)	(11.33)	(>20)	(>20)	(>20)	(9.65)
$\bar{R}^2$	0.0906	0.1275	0.1684	0.1533	0.0906	0.1274	0.1684	0.1534
obs.	26,242,261	8,046,594	9,314,410	8,881,202	26,242,261	8,046,594	9,314,410	8,881,202

This table display the results from the regression of two-week returns on groups with different centrality, controlling log address-level trade volume, log address-level number of trades, log bitcoin price volatility, log market-level trade volume, log market-level number of trade. Centrality is measured using both degree centrality and eigenvector centrality. I first examine the relationship between return and centrality using the whole sample, then I further break it down to three different periods. All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.

Table A.12: Centrality and weekly returns in different groups, M=50

	Degree Centrality				Eigenvector Centrality			
	Full Sample	First Period	Second Period	Third Period	Full Sample	First Period	Second Period	Third Period
Group 1	0.1905	0.0805	0.4007	0.0528	0.1124	-0.0659	0.4019	0.0197
(0-10%)	(>20)	(14.66)	(>20)	(7.51)	(>20)	(-9.15)	(>20)	(2.55)
Group 2	0.1806	0.0376	0.3643	0.0571	0.1696	0.0922	0.3182	0.0348
(10-20%)	(>20)	(4.71)	(>20)	(6.29)	(>20)	(13.75)	(>20)	(4.32)
Group 3	0.1816	0.0391	0.3390	0.0737	0.1743	-0.0137	0.2457	0.0284
(20-30%)	(>20)	(5.86)	(>20)	(9.74)	(>20)	(-1.98)	(18.95)	(3.57)
Group 4	0.1473	0.0431	0.3342	0.0761	0.1442	0.0505	0.2383	0.0894
(30-40%)	(>20)	(6.62)	(>20)	(9.71)	(>20)	(7.47)	(18.26)	(11.16)
Group 5	0.1124	0.0218	0.3613	0.0492	0.1660	0.0155	0.3322	0.0499
(40-50%)	(>20)	(3.58)	(>20)	(6.40)	(>20)	(2.32)	(>20)	(6.31)
Group 6	0.0956	-0.1258	0.4471	0.0290	0.1267	-0.0414	0.3393	0.0558
(50-60%)	(19.00)	(<-20)	(>20)	(3.72)	(>20)	(-6.41)	(>20)	(7.31)
Group 7	0.1557	-0.0751	0.2786	0.0095	0.1165	-0.0579	0.3610	0.0301
(60-70%)	(>20)	(-11.68)	(>20)	(1.27)	(>20)	(-9.64)	(>20)	(4.06)
Group 8	0.1301	-0.0840	0.2357	0.0438	0.1328	-0.0431	0.3928	0.0680
(70-80%)	(>20)	(-12.15)	(18.21)	(6.09)	(>20)	(-7.23)	(>20)	(9.63)
Group 9	0.1870	0.0292	0.4500	0.0455	0.3873	0.0714	0.9850	0.0792
(80-90%)	(>20)	(4.86)	(>20)	(6.26)	(>20)	(13.23)	(>20)	(11.46)
Group 10	0.4350	0.2023	0.9827	0.0109	0.2873	0.2258	0.5731	0.0067
(Top 10%)	(>20)	(>20)	(>20)	(1.48)	(>20)	(>20)	(>20)	(0.87)
$\bar{R}^2$	0.0645	0.0843	0.0899	0.0748	0.0645	0.0843	0.0900	0.0748
obs.	26,242,261	8,046,594	9,314,410	8,881,202	26,242,261	8,046,594	9,314,410	8,881,202

This table display the results from the regression of weekly returns on groups with different centrality, controlling log address-level trade volume, log address-level number of trades, log bitcoin price volatility, log market-level trade volume, log market-level number of trade. Centrality is measured using both degree centrality and eigenvector centrality. I first examine the relationship between return and centrality using the whole sample, then I further break it down to three different periods. All the results are reported with robust standard errors. The numbers in the parentheses represent the t-statistics.