DEPARTMENT OF ENGINEERING MATHEMATICS

Using Bitcoin's Transaction Network to Determine Price Trends

Luke Kirwan

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of Master of Science in the Faculty of Engineering.

Friday 9th September, 2022

Supervisor: Dr. John Cartlidge

Declaration

This dissertation is submitted to the University of Bristol in accordance with the requirements of the degree of MSc in the Faculty of Engineering. It has not been submitted for any other degree or diploma of any examining body. Except where specifically acknowledged, it is all the work of the Author.

Luke Kirwan, Friday 9th September, 2022

Contents

List of Figures

List of Tables

Abstract

Blockchain technologies rely on a publicly visible, distributed ledger, which records every transaction ever made. Prior to blockchain, it was difficult to obtain such a rich set of transactional data, due to the sensitive nature of transactions and the privacy of exchanges. Now, with access to the distributed ledger, networks of cryptocurrency transactions can be constructed and analysed via the well-established field of graph theory. This paper analyses the temporal structure of the transaction network for the largest cryptocurrency - Bitcoin - and draws parallels between this information and the conditions of the market, i.e. bullish and bearish markets. Two characteristics of the network are inspected: clustering, and the properties of network hubs. Techniques from econometrics and time series analysis are deployed, to establish market conditions and understand the temporal relationship between these conditions and the selected characteristics of the network. Effective techniques for extracting, constructing, and analysing Bitcoin's transaction network are also documented. Despite promising initial results for the clustering of Bitcoin addresses, there is insufficient statistical evidence to suggest it can be used to predict changing market conditions. However, after mapping the Bitcoin addresses to users, evidence is found that the clustering of Bitcoin can be used to forecast Bitcoin's price. For network hubs, the ratio of in-degree to out-degree is shown to be a useful metric for forecasting changes in Bitcoin's price. A model that successfully utilises these results to forecast changing market conditions would be invaluable to users of digital assets.

Supporting Technologies

- Blockchain data was provided in a *PostgreSQL* database by Mesonomics. I used *SQL* to query the database.
- I used *OpenVPN*, *PuTTY*, and *Byobu* for establishing and maintaining connections to the database server.
- I used the *SQLAlchemy*, *Pandas*, and *Networkx* public-domian Python Library for generating metrics for (small) transaction networks.
- I used the *PySpark* and *GraphFrames* public-domian Python Libraries for generating network metrics for (large) transaction networks.
- I used the *statsmodels* and *SciPy* public-domian Python Libraries for statistical analysis, and *Matplotlib* and *seaborn* for visualising results.
- I used *JupyterLab* for exploratory data analysis, and executed Python scripts for longer running jobs.
- I used *Gephi* for visualising the transaction networks.
- I used *Microsoft PowerPoint* for creating diagrams.
- I used LaTeX to format my thesis, via the online service *Overleaf*.

Notation and Acronyms

CDBC	:	Central Bank Digital Currency
BAN	:	Bitcoin Address Network
BUN	:	Bitcoin User Network
BLN	:	Bitcoin Lightning Network
UTXO	:	Unspent Transaction Output
EOA	:	Externally Owned Account
BTC	:	Bitcoin (unit of currency)
Defi	:	Decentralised Finance
NFT	:	Non-fungible tokens
P2E	:	Play-to-earn
TDA	:	Topological Data Analysis
TSA	:	Time Series Analysis
i.i.d.	:	independent and identically distributed
BB	:	Bry & Boschan method
MS	:	Markov Switching model
RDD	:	Resilient Distributed Dataset
	⋮	
$\mathcal{O}(n^2)$	:	Computational complexity is proportional to the square of the dataset size

Acknowledgements

The author would like to thank Mesonomics for providing reliable access to Bitcoin transaction data as well as a significant amount of processing power. In particular, thanks to Dr. Steve Phelps for his insightful comments and patience in teaching about distributed systems and big data technologies. Also, thanks goes to Dr. John Cartlidge for supervising the project and helping steer it in the right direction.

Chapter 1

Introduction

Graph theory is a well-established field of mathematics, which in it's most basic form is the study of entities (vertices) and the relationships between those entities (edges). Graph theory was first popularised after Leonhard Euler famously solved the Königsberg Bridge problem in 1736 [39]. Since then, techniques from graph theory have majorly evolved and found application in a range of disciplines including biology, computer science and the social sciences [45, Chapters 2,3,4,5]. The importance of graph theory is highlighted by its contribution in steering two of today's tech giants to success: Facebook and Google. Facebook built a highly successful digital social network, whilst Google modelled the world wide web as a network of pages and hyperlinks. Google famously patented an algorithm for determining the importance of web pages, which remains to this day the innovation driving the dominance of their search engine [47].

Despite an extensive history and success of application in multiple disciplines, graph theory has had limited applications in finance, primarily due to a lack of network-like data. Financial transactions are able to be modelled as a graph, however they are highly sensitive and frequently take place on private exchanges, of which there are many. Therefore, even those entities with access to transaction data usually have an incomplete picture of the transaction network, and cannot study its dynamics at the macro level.

Recently, a new financial market has arisen which publicises the entire history of transactional data by design - the market of cryptocurrencies. These markets are based on blockchain technology, a specific type of distributed ledger technology, which stores the entire history of transactions as an immutable sequence of blocks. The most famous application of blockchain technology, Bitcoin, was introduced in 2008 by Satoshi Nakamoto. As explained in more detail in Section 2.1, Bitcoin made the key step of introducing a validation mechanism through a peer to peer (P2P) network of participants who confirm transactions by consensus.[44] This meant that, for the first time, it was possible to issue and exchange currency without the need for a central authority, such as a bank. Also for the first time, the entire history of transactions within a financial market became publicly available. To give some insight into the scale of this, up to a half a million new transactions are added to the blockchain every day.[51]

In the past, graph theory has been applied to Bitcoin's transaction network, however much of this research is purely descriptive and makes no attempt to model the relationship between the properties of the network and economic phenomena. For example, many studies have focused on network construction, network profiling, and network-based detection for tasks such as entity recognition and fraud detection, which are further discussed in Section 3.[58] The papers that do link network properties to economic observations have often demonstrated the importance of information in the transaction network. For example, a landmark paper studied the wealth distribution of Bitcoin users and showed that "the rich get richer", i.e. those with more Bitcoin tend to obtain proportionally more Bitcoin as time progresses.[31] Other studies linking the transaction network to economic variables tend to only consider economic phenomena over short time periods, such as the daily volatility of the price of Bitcoin,[13] or the exchange rate of Bitcoin to fiat currencies.[48, 8, 32, 29, 3, 2] Parallels have also been drawn between the characteristics of the network and longer-term economic phenomena, such as the appearance of bubbles in the price of Bitcoin,[12] however this research is relatively sparse and often limited by the period over which the study was carried out.

Considering long term market conditions is just as important as being able to predict prices in the short term. For market players looking for anything other than a quick profit, insight into the conditions of the market and whether they are likely to change could be vital. In fact, an early warning system which analyses the transaction network to foresee the cycle of booms and busts would be the first of its kind. But who would use such a system?

For starters, those who use Bitcoin as an investment could use an early warning system to mitigate risks and losses, determine when to invest, and ensure an appropriate blend of assets to capitalise on a given set of market conditions. There has long been a debate about whether Bitcoin behaves more like an asset or a currency.[59, 9] No matter the conclusion, some use Bitcoin as an asset for investment, whilst others use it as a means for exchange. Those who use Bitcoin as a means for exchange could use an early warning system to better understand the purchasing power of the currency at a given moment in time. There are also other entities to consider. Regulators, who keep a keen eye on the evolution of cryptocurrencies, could use an early warning system to help protect investors from adverse market conditions. Governments, who in many cases are keen to adopt cryptocurrencies domestically to boost the economy, could use an early warning system to protect their citizens and stabilise the economy by tweaking fiscal and monetary policies. In the unregulated, decentralised world of cryptocurrencies, there is no central planning, so macro economic indicators might not be considered useful. However, as governments increasingly adopt central bank digital currencies (CDBC), such as the electronic Chinese yuan (e-CNY),* insights into the dynamics of these currencies will become increasingly useful. Finally, it's reasonable to hypothesise that the relationship between the transaction network and value of a decentralised currency reflects the relationship between the transaction network and value of a centralised currency. The results of this study could therefore extend to traditional currencies, which may be of interest to academics and the wider economic community. At the very least, discovering whether properties of the transaction network influence (or are influenced by) exogenous events would be an interesting insight.

The main objective of this paper is to determine if the network of transactions can generate useful and accurate economic indicators for determining long-term market conditions, such as periods of bullishness and bearishness. Of particular interest are periods of changing market conditions, for example the exiting of a bull market and entering into a bear market. From the network, there is a huge range of metrics and dynamics to choose from. However, only certain properties are likely to yield valuable information. In this study, clustering metrics and metrics computed on hubs in the network are considered. The clustering of the transaction network, described in Section 2.3.2, has in the past shown a regime change between periods of boom and bust.[13] However, this research only considered the unweighted, undirected transaction network at the daily frequency (edge weight and direction are explained in Section 2.3). This paper tests the hypothesis that the value of the transactions and direction of edges are useful information for determining economic conditions. Additionally, the networks are studied at lower frequencies, because the market conditions under inspection are long-term phenomena.† The second element of the transaction network under consideration are network hubs, described in Section 2.3.3. The importance of network hubs has been shown to increase during bubbles in Bitcoin's price.[12] Also, the net flow of exchanges has been shown to be correlated with movements in Bitcoin's price.[29] This paper considers several methods for identifying hubs in the network, and explores their role over different network frequencies. The metrics selected for hub analysis are the in/out degree ratio, and the net flow of Bitcoin through the hubs.
In summary, this paper aims to answer the following questions:

1. Does clustering in Bitcoin's transaction network provide useful information for determining market conditions?

2. Does studying hubs in Bitcoin's transaction network provide useful information for determining market conditions?

3. Does adding weight and direction to network edges provide extra information for determining market conditions?

In relation to these aims, the primary contributions of the paper are:

1. Confirm previous results that address level clustering changes in line with market shocks, but find little utility for address clustering in determining market conditions. However, statistical evidence is found that user level clustering contains useful information for forecasting Bitcoin's price.

2. Confirm previous results that the the degree distribution of network hubs is related to market shocks, but show that the net flow of hubs is not particularly useful.

*The e-CNY does not have a public, blockchain-based, distributed ledger. It is therefore not a cryptocurrency and transaction network analysis of the type described in this paper is not possible for the e-CNY. However, it is only one early example of a CDBC, and future CDBCs could feasibly be blockchain-based. The UK, for example, invited a discussion on CDBCs and the prospect of a DLT based CDBC.[52]

†The daily network contains approximately 144 blocks and 300-400 thousand transactions at current levels (August 2022). This network could have a very different structure to the weekly network, which contains approximately 1008 blocks and 2-3 million transactions.

3. Find that edge direction adds useful information for determining market conditions, whilst edge weight does not.

Given the scale of the networks under analysis, data engineering was a major consideration of this paper. Several secondary contributions are mentioned below:

- Establish a mechanism for labelling market conditions in cryptocurrency markets.
- Establish alternative mechanisms for identifying hubs in the network.
- Establish a method for analysing transaction networks at scale.
- Extend the functionality of an open-source package for large scale data network analysis (GraphFrames).
- Show that user level metrics should not be used as a proxy for address level metrics.

In future, as cryptocurrencies become more widely adopted, the network of transactions will grow and a richer picture of interactions will emerge. Events such as El Salvador's adoption of Bitcoin as legal tender in September 2021 have the potential to greatly change the transaction network.[34] Whilst it's possible that the findings here will become outdated as the transaction network evolves, evidence has been found that Bitcoin's transaction network is close to a mature, steady state, and that the characteristics of the network have already converged.[31] If the characteristics and trends of the network found in this paper persist as the network grows, the results of the paper will continue to provide utility to users of digital assets.

The remainder of the paper is structured as follows. Chapter 2 gives a technical background of the relevant fields for this study. Chapter 3 conducts an extensive review of the literature that exists to date. Chapter 4 describes the design considerations and methodology, whilst Chapter 5 discusses the technical challenges that arose in practice. Chapter 6 explores the network properties of Bitcoin's transaction network and examines their relationship with established market conditions. Chapter 7 summarises the findings of the paper, considers the limitations with the approach, and makes suggestions on how future efforts could be focused.

Chapter 2

Technical Background

2.1 Bitcoin

The information in this section largely comes from the textbook *Mastering Bitcoin: Programming the open blockchain*, by *A. Antonopoulos*.[5]

Fundamentally, Bitcoin combines three innovations to enable the operation of a secure, verifiable, decentralised digital currency. These are a decentralised peer-to-peer (P2P) network, a public transaction ledger (the blockchain), and a consensus mechanism (Proof-of-Work).

2.1.1 The P2P Network

The term peer-to-peer refers to the fact that all participants within the network are equal. This, along with the flat structure of the network, gives rise to the concept of decentralisation, since there are no central servers with extra privileges or authority. All participants in the network must be able to route transactions by forwarding them to neighbouring nodes, a process known as *flooding*. This is vital for the rapid dissemination of transactions through the network. Some nodes, known as *full nodes*, maintain a complete copy of the blockchain. Since multiple nodes perform this function, a common consensus on the current state of the blockchain can be achieved. Participants in the P2P network can also perform other functions, such as mining, which is discussed in Section 2.1.4.

2.1.2 Transactions

Transactions are fundamental to Bitcoin and other cryptocurrencies. In fact, the entire Bitcoin system is dedicated to creating, propagating, validating, and storing transactions; the blockchain itself is a record of all the transactions that have ever occurred. In its most basic form, a transaction takes Bitcoin from one set of (input) addresses and deposits it to another set of (output) addresses. That is, a transaction is a many-to-many mapping, moving Bitcoin from a set of input addresses to a set of output addresses. Figure 2.1 depicts several common transactions.

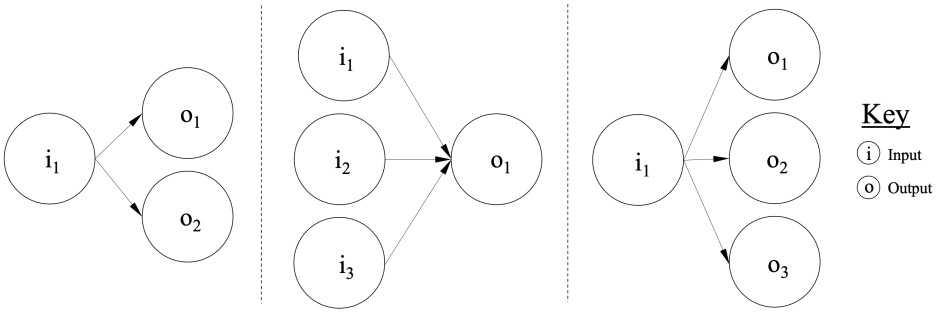

Figure 2.1: Three of the most common transactions in Bitcoin. On the left is a payment from i_1 to o_1, generating *change* (described in Section 2.1.2) which is sent to o_2. In the middle is an aggregating transaction, which groups funds from $i_{1,2,3}$ and deposits them at o_1. On the right is a distributing transaction, which distributes funds from i_1 to $o_{1,2,3}$. Note that none of these transactions are many-to-many.

Inputs and Outputs

Interestingly, a transaction *input* is just a reference to the output of a previous transaction, so transaction outputs are covered first. A transaction *output* contains two essential pieces of information: an amount of Bitcoin, and a locking script which determines the conditions under which the Bitcoin can be spent. When a transaction occurs, its outputs are initially unspent and referred to as unspent transaction outputs (UTXO). To spend a UTXO, a subsequent transaction must contain an input that references it and includes a valid unlocking script. Once the unlocking script has been applied and the output has been spent, it is no longer considered a UTXO, and cannot be used as input in subsequent transactions. This solves the double spending problem, which was a significant problem for digital currencies prior to Bitcoin.

The amount of Bitcoin in a UTXO is indivisible, meaning it must be spent in its entirety. This means most transactions generate *change*, which can be transferred to a *change address* in an output of the transaction. Change addresses can be used to associate Bitcoin addresses with known users, as discussed in Section 2.4.2. The sum of input values in a transaction is always less than the sum of output values.* The difference is a fee paid to the miner (mining is discussed in Section 2.1.4), to compensate them for the work they must perform to verify the transaction and include it in the blockchain.

2.1.3 Addresses and Wallets

A Bitcoin *address* is derived from the public part of a public/private key pair generated using techniques from cryptography. Transaction outputs are often locked to an address through the locking script mentioned in 2.1.2. The private key corresponding to this address can be used to generate a digital signature which unlocks the script. Bitcoin *wallets* are pieces of software that store the private keys associated with Bitcoin addresses. They provide the interface for users to interact with the Bitcoin system.

2.1.4 Blocks and Mining

As discussed in Section 2.1.1, functions other than routing (of transactions) and storing (of the blockchain) are performed by certain nodes in the P2P network. *Mining* is one such function, which is vital for decentralised consensus in Bitcoin. Mining nodes maintain a pool of propagated transactions which have not yet been included in a block (hence these transaction are considered *unconfirmed*). They group transactions from the pool into a candidate block, which usually contains somewhere between 300 and 600 transactions. Priority is given to transactions offering a higher *fee* (transaction fees were discussed in Section 2.1.2). The miner also adds a special transaction, called the *coinbase* transaction, which allocates the transaction fees, as well as a block reward, to the miner. Additionally, the hash of the previous block in the blockchain is included in the candidate block, creating an immutable sequence of links between blocks in the chain. Once the block has been created, the miner must solve a computationally difficult problem before any of the other miners to have their block appended to the blockchain. This process is called *Proof-of-Work* and is one of the fundamental innovations of Bitcoin mentioned at the start of the chapter. The work here refers to finding a special number (the *nonce*) that, when included in the block header, results in a hash (using SHA256) with a pre-determined leading number of zeros. The first miner to achieve this propagates their new block to the rest of the network, who validate the hash and propagate it themselves. As the new block floods through the network, it is appended to the blockchain on each full node. When other miners see the new block, they immediately begin work on the next block, and the process starts over. The transactions in the successful block are removed from the transaction pools of the miners, so that they don't try to add them to subsequent blocks. Since the block is now included in the blockchain, the transactions are deemed to be *confirmed*.

2.2 Other Cryptocurrencies

The information in this section comes from the textbook *Mastering ethereum: building smart contracts and dapps*, by *A. Antonopoulos* and *G. Wood*.[6] Although Bitcoin was the first cryptocurrency and remains the largest by market capitalisation, it is by no means the only one.[50] The second largest is Ethereum, which appeared in 2014. Similarly to Bitcoin, Ethereum has a P2P network, a public distributed ledger, and a consensus mechanism which relies on Proof-of-Work (although Ethereum is

*Whilst it is possible for a transaction to be processed with no fee, it is highly unlikely, since it will not be prioritised by Bitcoin miners.

transitioning to a different consensus algorithm, Proof-of-Stake, in September 2022). Despite these similarities, Ethereum is fundamentally different to Bitcoin. The differences stem from the inclusion of a Turing complete programming language, which enables highly flexible programs called *smart contracts* to be deployed and executed. Recall from Sections 2.1.2 and 2.1.3 that Bitcoin transactions occur between addresses, which are controlled by users through wallets. In Ethereum, addresses can belong to users or they can have smart contracts deployed behind them. These are known as *externally owned accounts* (EOA) and *contract accounts* respectively. Only EOAs can trigger a transaction, but any transaction sent to a contract account can invoke a chain of subsequent transactions invoking other contracts. In this way, Ethereum is viewed as a global decentralised computer. So, despite relying on similar underlying technologies, Ethereum is inherently different from Bitcoin. Importantly, due to the huge differences in transactions, the Ethereum transaction network is structurally different to the Bitcoin transaction network. This is a significant motivation for studying cryptocurrency transaction networks individually.

2.3 Graph Theory

The information in this section largely comes from the textbook *Networks: An Introduction*, by *M. Newman.*[45]

A graph, or network, is a structure containing a set of vertices connected by a set of edges. Graph theory is the branch of mathematics dedicated to the study of graphs. Figure 2.4 depicts two different graphs representing the same Bitcoin transaction. The graph on the left contains no edges starting and ending at the same vertex (called *self-loops*), and is *bipartite*, because the vertices fall into two classes (addresses and transactions) and each vertex is only connected to vertices of the opposite class. The right-hand graph in Figure 2.4 is neither simple nor bipartite. Edges in a network can be *directed*, meaning they point towards one vertex and away from another. Vertices and edges can also have metadata associated with them, enabling graphs to contain lots of complex information. For example, the edges in the left-hand graph of Figure 2.4 are *weighted*, because they have a value attached to them, which in this case represents the amount of Bitcoin transferred.

2.3.1 Centrality

The *centrality* of a vertex describes the importance of that vertex within the network. The most basic measure of centrality is the *degree* of a vertex, which is the number of edges attached to it. The *degree distribution* describes the probability distribution of degree values over all vertices in the network. If this follows a power law, the network is said to be *scale-free*. For a vertex in a directed network, the *in-degree* is the number of inward edges, and the *out-degree* is the number of outward edges. For example, vertex t_1 in the left-hand graph of Figure 2.4 has the highest in-degree and out-degree in the network, and clearly plays an important role. Many other methods for determining centrality exist, such as the PageRank algorithm developed by Google.

2.3.2 Clustering

Clustering describes groups of densely connected vertices within a network. These groups can come in many shapes and sizes, such as a triangle. Three connected vertices are called a *triangle* if there is an edge between each pair of vertices, as in Figure 2.2. If the vertices are only connected by two edges, the group is referred to as an *incomplete triple*. The *local clustering coefficient* measures the proportion of triangles that a vertex is part of, out of all triangles it could possibly be part of given its neighbouring vertices. That is, the local clustering coefficient measures the proportion of neighbours of a vertex that are also connected. This is defined in Equation 2.1, where C_i is the local clustering coefficient for vertex i.

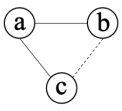

Figure 2.2: A triple in an undirected network. If the dotted line exists, the triple is closed and called a triangle.

$$C_i = \frac{\text{(number of pairs of neighbours of } i \text{ that are connected)}}{\text{(number of pairs of neighbours of } i\text{)}} \tag{2.1}$$

Clustering can also be considered across the whole network. One method is to average the local clustering coefficient across all vertices, producing the *average local clustering coefficient*. Another method is to compute the *global clustering coefficient*, or network *transitivity*, which measures the proportion of complete triangles over the entire network. This is defined in Equation 2.2, where T is the transitivity.

$$T = \frac{\text{(number of triangles)} \times 3}{\text{(number of triples)}} \tag{2.2}$$

It is simplest to assume the network is undirected when calculating the local or global clustering coefficients. However, both metrics can be computed for directed networks too, by considering the fraction of complete directed triples. There are eight possible ways for a set of three vertices to be connected by directed edges, shown in Figure 2.3, but only six are considered in the calculation of transitivity.[45]

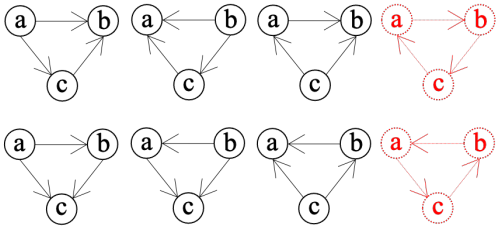

Figure 2.3: All possible complete triples in a directed network. The red triples are ignored in the calculation of transitivity.

2.3.3 Hubs

In graph theory, a network *hub* is a vertex with significantly higher degree than average. Hubs cannot exist in random networks, however in scale-free networks they are an expected phenomenon. Bitcoin's transaction network has been shown to be scale-free, as discussed in Section 3.3.1. Hubs are sometimes defined less precisely as vertices with high centrality. Section 6.1 and Figure 6.1 discuss the difficulties of using the degree to identify hubs in the Bitcoin transaction network.

2.4 Constructing Bitcoin's Transaction Network

As discussed in Section 2.1.2, a transaction is a many-to-many mapping from input addresses to output addresses. Taking the set of transactions that occurred over a given period (say, a week), a picture begins to emerge of the interactions between addresses. This is called the *Bitcoin Address Network* (BAN), where each vertex is an address, and each edge connects an input and output address from the same transaction. An important property of Bitcoin addresses is that many of them can belong to a single user. For example, hierarchical deterministic wallets contain multiple addresses which are all derived from the same key, and therefore all belong to a single user.[5] By mapping addresses to users, a second representation of the transaction network can be formed, in which vertices represent users and edges represent transfers of value between users. This is called the *Bitcoin User Network*.

2.4.1 Bitcoin Address Network

For one-to-many transactions, such as those depicted in Figure 2.1, it is possible to tell how much Bitcoin was sent from an input address to an output address. For many-to-many transactions, such as the one depicted in Figure 2.4, this is not possible. All that can be derived from such a transaction is that all inputs transacted with all outputs. That is, the Cartesian product of inputs and outputs for a transaction represents all the pairs of addresses that were input and output in the same transaction. By taking the Cartesian product over multiple transactions, the BAN can be built. The edges of the BAN are directed, since they point from an input address to an output address, and there can be multiple edges in either direction between the same vertices, because addresses can be paired in more than one transaction. The

network can also contain self-loops, since an address can be both an input and an output in the same transaction. This is often the case with change addresses, although it is generally best practice to use a new address as a change address.[5] To summarise in the terminology of graph theory, the BAN is a *directed multigraph*.

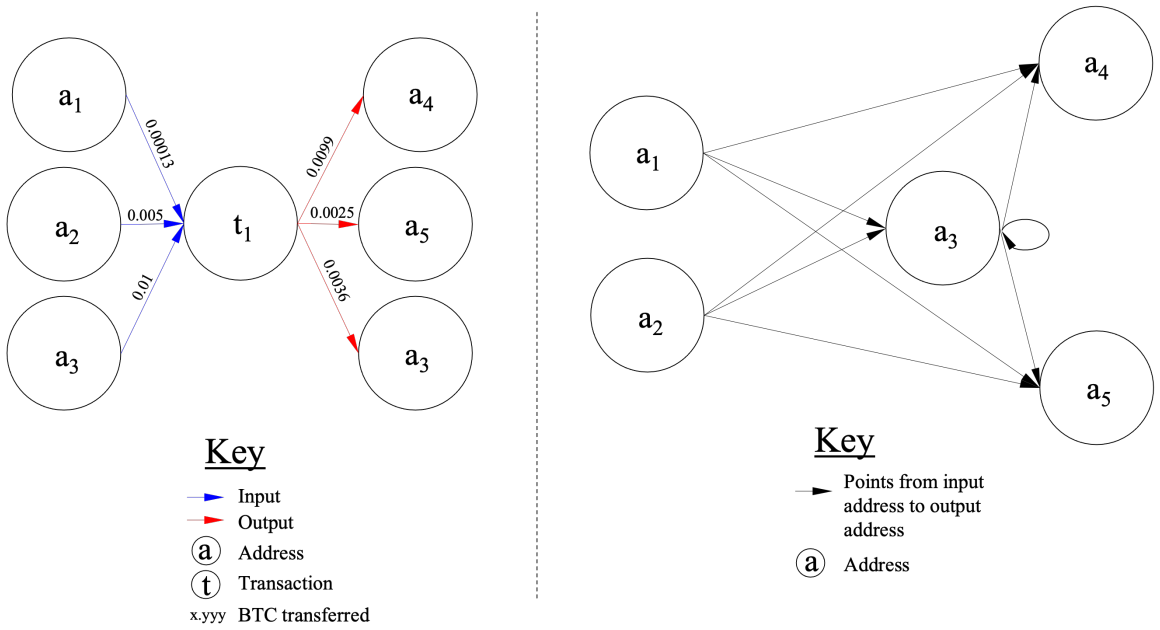

Figure 2.4: The Bipartite (left) and Cartesian (right) representations of Bitcoin transaction with hash ending 2331c14, which was recorded on the blockchain on the 20th April 2020. Address a_3 is both an input and an output of the transaction. Note that the the Cartesian representation has only one class of vertex (an address), whereas the Bipartite representation has both addresses and transactions as vertices. Note also that the Cartesian representation has more edges. Having a single class of vertex can lead to more meaningful metrics, whilst having more edges can make computation of network metrics more difficult.

The problem with the BAN is that it doesn't show the amounts transferred. That is, the BAN is thus far unweighted. Due to the many-to-many nature of transactions, it is impossible to say how much was transferred from a given input address to an output address. Consider the Bipartite representation on the left of Figure 2.4. It is clear how much was sent by each input address, and how much was received by each output address. However, it's impossible to determine precise information such as where the 0.0099BTC a_4 received came from. It could all have come from a_3, or half from a_3 and the other half from a_2, or some other combination of input values. Therefore, weights cannot be accurately assigned to the Cartesian representation. However, an *approximation BAN* with estimated weights can be derived by using Equation 2.3.

$$\text{weight}_{i,j} = \text{input}_i \times \frac{\text{output}_j}{\sum \text{outputs}} \tag{2.3}$$

2.4.2 Bitcoin User Network

Numerous studies have considered the problem of *entity resolution*, for mapping Bitcoin addresses to established entities. This is discussed further in Section 3.3.2. The first step of entity resolution is *address translation*, which involves clustering Bitcoin addresses together as those belonging to a single user. For this, the *multi-input* and *change-address* heuristics have become widely adopted. The first is based on the assumption that all input addresses in a transaction belong to the same user, because whoever constructed the transaction holds the private keys to all the input addresses. This heuristic has the result of transforming previously many-to-many transactions into one-to-many transactions (demonstrated in Figure 2.5), which is useful because edge weights can be easily determined. The second heuristic, the change address heuristic, is based on the assumption that a change address appearing only once belongs to the same user as the set of inputs. This heuristic is less robust and identifies a much smaller subset of the addresses than the multi-input heuristic.

After performing address translation the BUN can be constructed. As with the BAN, the BUN is a directed multigraph that allows self-loops. The BUN is advantageous because it's possible to tell exactly how much transferred from one user to another (shown in Figure 2.5), due to the one-to-many nature of the mapped transactions. However, valuable information can be lost during address translation.

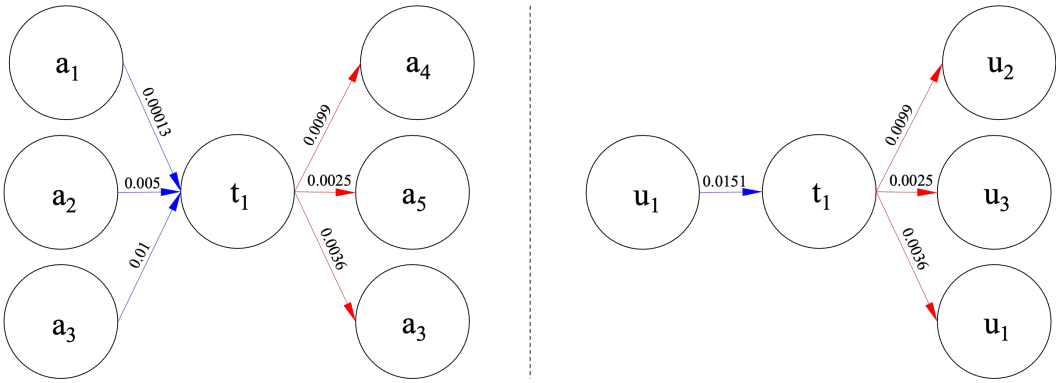

Figure 2.5: BAN and BUN representations of the same transaction. On the left, the BAN. On the right, the BUN after the multi-input and change-address heuristics have been applied. Notice that all input addresses $a_{1,2,3}$ are mapped to a single user u_1, and change address a_3 is also mapped to u_1. Notice also that the input weights are summed.

2.5 Time Series Analysis

The information in this section largely comes from the textbook *Forecasting: Principles and Practice*, by *R. Hyndman* and *G. Athanasopoulos.*[30]

All the data analysed in this project is *time series* data. That is, the data takes the form a sequence of values with equal time spacing in between them. For example, Figure 4.1 shows the time series of Bitcoin prices, spaced at daily intervals, over a five-year period.

2.5.1 Stationarity

A key concept in time series analysis is *stationarity*. A time series is said to be stationary if the process that generates it has statistical properties that do not change over time. Many techniques in time series analysis (TSA) assume the time series of interest to be stationary, or require transformations to make a non-stationary series stationary. The most common way of doing this is through a technique called *differencing*, which is the process of finding the difference between consecutive values in a time series. A time series is *integrated* if some number of differencing transformations produces a stationary series. Sometimes it is difficult to tell if a time series is stationary or not. Fortunately, statistical hypothesis tests such as the *Augmented Dickey Fuller* (ADF) test can be used to test for stationarity. The null hypothesis of the ADF test is that a unit root is present in the time series, which implies a reliance on previous values of the series and thus non-stationarity. If the test statistic is significant enough, the null can be rejected and the series determined to be stationary.

2.5.2 Correlation and Causality

When dealing with multiple time series it is common to consider whether there is correlation between them. However, one must be careful when correlating time series, as they are generally not independently and identically distributed (i.i.d.), and instead have some temporal dependence. Consider Figure 2.6, which plots three independent trials of flipping a coin. From visual inspection, the series of cumulative scores for the three coins appear to be correlated. The correlation coefficients are also suggestive of this. However, the correlation is not at all meaningful – there's no connection between the series at all. This apparent but meaningless correlation is called *spurious correlation*. The lower row of Figure 2.6 shows that, by differencing at a lag of one, the cumulative scores can be transformed into stationary series which do not display between-series correlation. This highlights the importance of removing within-series dependence before testing for correlation. The within-series dependence on display here is known as

autocorrelation. The cumulative sum of coin flips is dependent on the past observations of the series, whereas the differences (the observed values) are not. It is common to model autocorrelation using an *autoregressive* model. In multivariate TSA, the observations of a series can also be dependent on the past observations of other series. In this scenario, a *vector autoregressive* (VAR) model is required. A VAR model has an autoregressive equation for each of the time series in the model, and each equation can be based on past observations from any of the other time series.

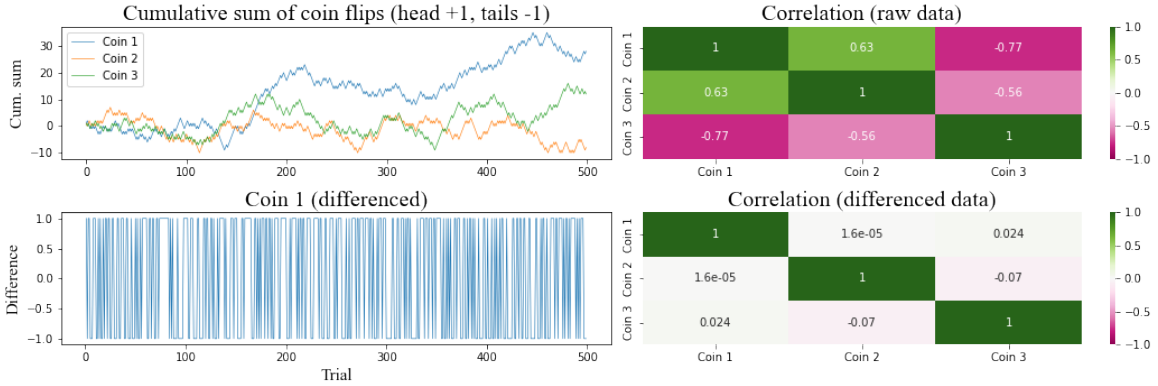

Figure 2.6: Top-left: time series data of the cumulative sum of successive coin flips, for three independent coins. Top-right: the Pearson correlation coefficient between each pair of cumulative sum time series. Bottom left: the differenced series for Coin 1, which is stationary. Bottom right: the Pearson correlation coefficient between each pair of differenced time series.

Despite the difficulties, some methods for testing correlation between series do exist. To use a correlation coefficient such as Pearson's (or preferably Spearman's, since it is non-parametric and does not assume i.i.d. observations), one must remove any within-series dependence before testing for correlation between the series. Methods such as these give a single value for correlation across the entire length of the series. They are therefore limited in that they do not provide information as to which series comes first (i.e. which series provides the signal). For this, a technique called *cross-correlation* can be employed. The *Granger causality test* is a popular method for testing for cross-correlation between two series. The null hypothesis is that the past values of one time series X do not provide statistically significant information about the future values of another time series Y. If statistically significant evidence is found, the null can be rejected and X is said to *Granger cause* Y. This test is somewhat confusingly named. Rather than testing for causality, what it really tests for is whether one time series can be used to forecast another.

Sometimes there is a need to test for Granger causality among non-stationary time series. The most rigorous approach for doing this is called the *Toda & Yamamoto procedure*, which relies heavily on VAR.[54] The procedure has thirteen steps which incorporate several tests including Granger causality, the ADF test, Johansen's test and the Durbin-Watson test.[†] The procedure is too lengthy to summarise here, so the reader is referred to this article which provides an elegant insight.[28]

[†]Johansen's test is for *cointegration*, which occurs when two time series are integrated together and follow a relationship in the long-term. The Durbin-Watson test is for within-series correlation and is usually applied to a set of residuals produced by a time series model. If the residuals show correlation, some useful information has likely not been incorporated into the model.

Chapter 3

Literature Review

As established in Chapter 1, the aim of this paper is to determine whether Bitcoin's transaction network can provide signals about long-term market conditions, such as extended periods of price rise and decline. The relevant areas of research for this study are graph theory, cryptocurrencies, and econometrics. Graph theory has been in active development for centuries and the depth of the literature reflects this. The same is true for econometrics. Cryptocurrencies, on the other hand, are a phenomenon of the 21st century. Despite this, they have piqued significant academic interest during their short existence. Ultimately, cryptocurrencies such as Bitcoin represent an amalgamation of research from different fields, such as cryptography and network science, which long predate the cryptocurrencies themselves. This chapter explores the relevant literature on graph theory, cryptocurrencies, and econometrics, before converging on a narrower view of the application of graph theory to Bitcoin's transaction network. This type of analysis has only been possible for thirteen years since the creation of Bitcoin, in which time the network changed significantly. In light of this, a close eye is kept on how the techniques applied in the literature have changed as the network has evolved.

3.1 Graph Theory

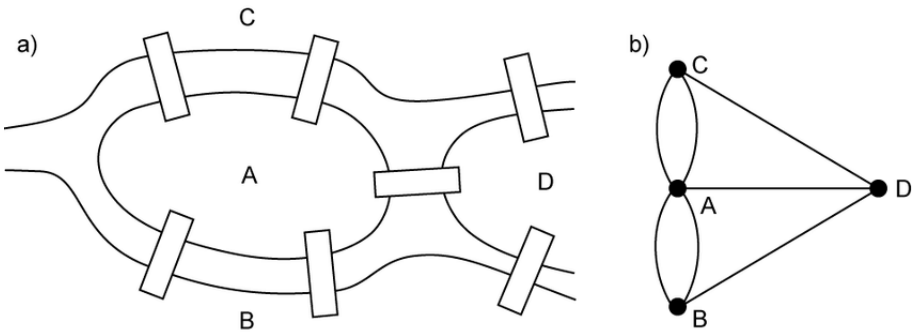

Figure 3.1: The Königsberg Bridge Problem (taken from [11]). Diagram *a* shows the geographical representation of the four land-masses connected by seven bridges. Diagram *b* shows the corresponding graph, with vertcies representing land-masses and edges representing bridges. Graphs and graph theory are described in Section 2.3.

3.2 Cryptocurrencies

In the wake of the financial crisis of 2008, the pseudonymous Satoshi Nakamoto tied together knowledge from multiple disciplines to create the first cryptocurrency - Bitcoin.[44] Nakamoto made the key step of combining a specific type of DLT with a consensus algorithm run by a network of miners, resulting in a secure method for executing and recording transactions, whilst also avoiding the troublesome double spending problem. In later years other cryptocurrencies emerged, such as Ethereum in 2014.[15] Both cryptocurrencies have seen tremendous growth since their introduction, indicated by the increasing market

capitalisation of the circulating coins and largely attributable to increased user adoption.[19, 20] Some developing nations are rapidly adopting cryptocurrencies, often to combat rising inflation in their domestic currencies.[21, 1] Although they are becoming more widespread, it is not always with the best intentions. As recently as 2019, Foley et al. found that one-quarter of Bitcoin users are involved in illegal activity, which is a lot of users given that there were over 100 million Bitcoin wallets in 2020 [26, 42]. This is likely driven by anonymity and lack of regulation. The increase in uptake of cryptocurrencies has been accompanied by an increase in knowledge, thanks to the release of texts by authors such as A. Antonopoulos, which enable new adopters to mount the steep learning curve.[5, 6] Cryptocurrencies are increasingly supported by an online community of users and hobbyists. This has been especially evident on the Ethereum platform, which has seen the rise of decentralised finance (DeFi), non-fungible tokens (NFTs), and play-to-earn (P2E) games, largely thanks to the Turing complete programmability of the platform through smart contracts.

A significant amount of research has focused on whether cryptocurrencies can really be considered currencies or if they are in fact assets.[9, 59] These studies found Bitcoin is more often used as an alternative investment than a medium of exchange, probably due to the high potential for significant returns. There has also been a considerable amount of research on the shortfalls of cryptocurrencies and how to improve them.[55, 60] Another area of research focuses on financial analysis of cryptocurrencies, considering the historical prices and the impacts of socio-economic data.[27, 16, 4] Most research focuses solely on Bitcoin, perhaps because it is the oldest and largest cryptocurrency (by market cap, July 2022).[19] However, Bitcoin's dominance is decreasing, highlighting the need for more extensive research into other currencies.[50]

3.3 Transaction Network Analysis

3.3.1 Network Profiling

Cryptocurrencies offer a unique opportunity to study the structural and temporal properties of the transaction network. The most broad studies have attempted to profile the properties of the network, without a specific goal in mind. Ron and Shamir conducted an early network analysis of Bitcoin transactions, and found that many Bitcoins lie dormant in addresses with no outgoing transactions.[48] They suggested that these coins are hoarded, which correlates with findings of other papers that Bitcoin is often used as an investment vehicle.[9, 59] An alternative explanation for this finding is that owners regularly lose the private key to their wallet, which is reasonably likely given the steep learning curve for new users of cryptocurrencies. These alternative explanations highlight how insights from the transaction network must be treated with care; the insights may be interesting but they do not necessarily reveal the full story. Ron and Shamir also studied a subgraph of large transactions, and found significant evidence of suspicious behaviour, possibly related to money laundering. Baumann et al. analysed the Bitcoin transaction network, with several interesting results. Firstly, they found that the network becomes more scale-free over time. Also, by analysing the clustering coefficient, they concluded that the network showed small-world* properties.[8] These results were the among the first signs that interesting insights could be drawn from studying the transaction network, and have since been repeated elsewhere in the literature.[43, 33]

Many of the early graphical analyses of transaction networks were focused solely on Bitcoin and only considered a static view of the network, without considering the evolution of the network over time. Recently, an increasing number of studies have considered the evolutionary dynamics of transaction networks. Motamed and Bahrak found that the transaction networks for different cryptocurrencies evolve in much the same way.[43] For example, they discovered that in both Bitcoin and Ethereum new vertices are increasingly likely to connect to vertices with different properties (a property called disassortativity), a result which was backed up by Liang et al.[33] This is interesting given the differences in transaction anatomy between the cryptocurrencies (discussed in Section 2.2). Liang et al. suggested that there are in fact differences between the transaction networks of different cryptocurrencies. For example, Bitcoin was shown to display small-world properties, whereas Ethereum was not. It's possible that this finding is a result of Ethereum's transaction network being further from convergence, since the cryptocurrency is younger. However, the more likely explanation is that the result reflects the different transaction structures in Bitcoin and Ethereum. This result highlights the importance of using different techniques for different cryptocurrencies.

*A small-world network is one in which the mean shortest path between two randomly selected vertices grows proportionally to the logarithm of the number of vertices

3.3.2 Entity Resolution

A fascinating problem tackled in much of the literature is that of de-anonymising transactions. Bitcoin has in the past been used as a means of illicit exchange and illegal activity, such as for payments on the Silk Road platform.[18] Users of Bitcoin originally believed that by creating multiple addresses and spreading transactions across them, one could wash money through the system without detection. However, a landmark paper showed that this is not the case by introducing two heuristics for address resolution, described in Section 2.4.[40] This result has been especially useful for constructing the BUN. Recent studies have constructed the BAN and the BUN separately and compared properties between them.[56] Although progress has been made in linking addresses to users, the final step of linking users to entities has proven more difficult. Several studies have identified hubs in the network by considering properties such as the degree distribution and clustering coefficient.[8] Fleder et al. took the work on entity resolution a step further by enriching the transaction data with metadata from external sources, such as social networks. They found that it was possible to recognise certain entities and suggested using more metadata, such as geographical and timestamped data, to improve the results.[25] Using the transaction network to perform entity recognition and fraud detection serves as a strong reminder of the value of information in the transaction network.

3.3.3 Forecasting Economic Variables

The research more relevant to this paper is that which has used the transaction graph to forecast economic data, such as the wealth of vertices,[31, 22] the exchange rate of the cryptocurrency,[48, 8, 32, 29, 3, 2] and the presence of bubbles in the price of the cryptocurrency.[12] Kondor et al. studied the degree distribution and clustering coefficient, as well as temporal patterns in the weekly transaction data, and found that sublinear preferential attachment determines the wealth distribution of vertices. That is, they found that the wealth of vertices with a large Bitcoin balance increased faster than those with a low balance.[31] De Collibus et al. applied a similar study to four of the largest tokens on Ethereum, namely USDT (Tether), WBNB (Binance), LINK (Chainlink) and ETH (Ether). They came to a similar conclusion; that the growth in wealth of these assets is driven by super-linear preferential attachment.[22] These studies show that the economic phenomenon "the rich get richer" holds for different cryptocurrencies. A possible explanation is that miners in the network hoard newly issued coins, driving up their relative wealth in the network. Again, this is an interesting insight only made possible by the publicity of the transaction network; it would be very difficult to draw such a conclusion in traditional financial markets.

There is a significant depth of research dedicated to forecasting the price of a cryptocurrency by studying the transaction network. This comes as no surprise, since such a tool would be invaluable to investors. Early papers, which mainly considered Bitcoin, made the simple observation that the number of users and transactions are highly correlated with the price.[8] An influential paper by E. Cheah and J. Fry found that Bitcoin has no inherent value.[16] This finding suggests that Bitcoin's value is based on the perceived value of the network and its utility as a medium of exchange or asset, and helps explain why the value of Bitcoin has grown as adoption has increased. Garcia et al. came to a similar conclusion, finding that the growth of Bitcoin has been driven by successive waves of new user adoption.[27]

Other research has linked more complex properties of the transaction network to Bitcoin's price. Brown found that the transitivity of Bitcoin's transaction network experienced a regime change during a crash in Bitcoin's price.[13] According to Brown, speculative flipping of assets is more likely to occur during bubbles, leading to a higher frequency of triadic motifs[†] and therefore transitivity in the network. This is an especially interesting result given its connection to long-term market conditions. This result is limited in that it only considered the unweighted, undirected representation of the transaction network and only considered the daily frequency. Given that the true transaction network is both weighted and directed, valuable information could have been lost. It would be interesting to see if these results extend to the weighted, directed network, constructed over a range of frequencies.

Greaves and Au were among the first to link vertices with high centrality to Bitcoin's price.[29] The study found that when certain users have a negative net flow (i.e. they send more Bitcoin than they receive) the price of Bitcoin tends to increase, and when these users have positive net flow the price of Bitcoin tends to decrease. The conclusion drawn from this was that these users are exchanges (such as Mt Gox, the largest exchange at the time). The logic for this argument is that exchanges having a negative net flow implies high demand for the cryptocurrency, leading to a rise in price. By studying the properties of vertices with high centrality, the authors came to meaningful conclusions about the relationship between

[†]Network motifs are defined by Milo et al. as patterns that occur much more frequently in the network of interest than in an ensemble of randomised networks.[41] Triadic motifs are those patterns that link three vertices.

the transaction network and economic variables. It would be interesting to understand if these results extend to longer-term price movements. The paper went on to demonstrate that machine learning models based on metrics from Bitcoin's transaction network were better at predicting price movements than a baseline model based on historical prices. This was one of the significant early results that provided motivation for Bitcoin transaction network analysis.

Market conditions, such as periods of bullishness and bearishness, are usually considered over weeks, months, and even years. Whilst there are many studies that look into long-term economic phenomena in Bitcoin markets, very few of them consider the role of the transaction network. Bovet et al. were among those who did, and found several interesting results by contrasting structural properties of the BUN to known periods of price rise and decline in the cryptocurrency.[12] Firstly, they found that the number of triadic interactions experienced a strong change before, during, and after the appearance of a bubble. This result is strongly related to the transitivity analysis of Brown. Secondly, they found that the frequency of network motifs changed significantly during bubbles. Thirdly, they found that during bubbles the out-degree distribution widened whilst the in-degree distribution narrowed. These changes indicate the importance of hubs during bubbles, by showing that a small number of hubs provide liquidity to a large number of users entering the network during the bubble. Given that the behaviour of hubs can easily be monitored via analysis of the transaction network, hub analysis seems a good avenue for exploration.

Recently, research has moved on from static analysis of a single cryptocurrency to dynamic analysis across multiple cryptocurrencies. For example, Motamed and Bahrak found that the growth rate of vertices and edges, and the density of the graph, are correlated to the price for multiple different cryptocurrencies.[43] In 2020, Vallerano et al. conducted an overview of the past studies on Bitcoin's transaction networks. They found that many of the studies had not considered the network's evolution in detail, and those that had tended not to link the evolutionary properties to empirical observations from proper models, indicating a gap in the research for evolutionary analysis.[56] As well as evolutionary analysis, topological data analysis (TDA) has emerged as a popular technique for transaction network analysis. Li et al. applied techniques from TDA, such as persistent homology and functional data depth, to the Ethereum transaction network, and avoided traditional graph analysis metrics such as transitivity.[32] The paper also introduced some new topological descriptors, called Betti limits and pivots, and found that Betti pivots help in detecting anomalous events in Ethereum's price.[‡] These results built on the findings of Abay et al., which used Betti derivatives to capture changing network topology in the Bitcoin transaction network and applied machine learning methods to effectively predict the price of Bitcoin over a seven day lookahead.[2] Another technique from TDA is the study of network motifs. Akcora et al. hypothesised that chainlets, an alternative term for network motifs, contain useful information about the price of a cryptocurrency. They found that certain types of chainlets, which they called extreme chainlets, had high predictive utility for Bitcoin.[3] These results indicate that TDA is fast becoming the state-of-the-art method for analysing cryptocurrency transaction networks.

3.4 Identifying Market Conditions

There has long been discussion about how to identify the conditions of a financial market, such as bull and bear markets. The terms 'bull' and 'bear' markets first appeared in Dow Theory and were popularised by the Wall Street Journal.[46] Bull markets were described as broad upward movements, and bear markets as the opposite. Despite the lack of clear definition, many attempts have been made to label historical market conditions as 'bullish' or 'bearish'. The simplest approach employs a moving average model to label the market based on the mean return over a pre-defined number of periods.[§][17] Beyond this, two popular approaches have emerged. The first is a parametric approach using a Markov Switching model (MS), which models the probability of transitioning between binary states representing bull and bear markets. The model takes historical returns and variance as inputs. This technique was established by Maheu and McCurdy, who used it to successfully identify all significant downturns over 160 years of historical market data, and found 14 months to be the optimal historical window over which to consider returns and variance.[36] This implementation of the Markov Switching approach was met with criticism for including dividends in the returns and only allowing two market states, thus ignoring conditions like bull market corrections and bear market rallies.[46, 37] The second technique, known

[‡]Betti limits and pivots are not explained in the technical background since they are not used in this study. For more information on these topological descriptors and TDA in general please refer to the original study [32]

[§]For example, if the mean return over the previous three months is greater than zero, the market is labelled a bull market.

as the Bry and Boschan method (BB), is a non-parametric method which identifies turning points and labels the intermediate periods as bullish or bearish.[14] The BB method has few assumptions and is relatively straight-forward to implement algorithmically. Whilst both the MS and BB approaches have been used extensively for labelling phases of traditional financial markets, only the BB approach has been applied to cryptocurrency markets, and only on one known occasion. Zhang et. al argued that since the BB approach focuses on the level of price change, it can be applied to any financial market, including cryptocurrency markets.[61] Generally, cryptocurrency markets have proven to be more volatile and have exhibited shorter cycles than traditional financial markets. As a result, techniques developed for labelling market conditions in traditional financial markets may need to be adapted for application in cryptocurrency markets.

3.5 Summary

Ten years ago, knowledge about cryptocurrency transaction networks was limited to a shallow understanding of static properties. However, over the past decade the emergence of evolutionary analysis and TDA have helped establish a deeper understanding. On more than one occasion the transaction network has been shown to contain highly valuable information for forecasting economic variables, such as the price of a cryptocurrency. As a result, researchers are devoting more time to comparing properties of the transaction network to economic phenomena. However, there is still relatively little research that has considered long-term economic phenomena, such as periods of bullishness and bearishness. The most promising findings relating the transaction network to long-term market conditions have come from evolutionary analysis of network transitivity and network hubs, as well as from techniques such as motif analysis within the field of TDA. Whilst the results of TDA are both promising and exciting, TDA is not explored further in this paper due to limited time and resources, but is suggested as an avenue for future work in Section 7.1. Instead, evolutionary analysis of transitivity and the behaviour of network hubs is explored. Out of the approaches for identifying market conditions, the BB approach is chosen, because it has previously been applied to cryptocurrencies and is straight-forward to implement.

Chapter 4

Design

This chapter details the design of experiments necessary to achieve the objectives outlined in Chapter 1. Recall that the main objective is to establish whether information from Bitcoin's transaction network can be used to predict changing market conditions in Bitcoin's price. Below is the high level process set out for achieving this objective:

1. Choose a set of network metrics based on intuition and past research.
2. Establish a mechanism for identifying market conditions.
3. Construct a sequence of graphs from transactions recorded on the blockchain.
4. Compute the metrics of interest on each graph to form a time series.
5. Use time series and statistical analysis to describe the relationship between the network metrics and market conditions.

4.1 Choosing Network Metrics

As discussed in Chapter 1, clustering and hub analysis were chosen as the areas of graph theory of interest. Clustering was chosen to test the theory that the transaction network becomes more tightly clustered during times of speculation. Hub analysis was chosen to test the theory that network hubs play an important role during booms and busts. For example, exchanges provide Bitcoin to speculative traders during bull runs, and facilitate the liquidation of Bitcoin assets during bear runs. Both theories have been given weight by promising results from past studies, as discussed in Section 3.3.3.

4.1.1 Clustering

For clustering, the transitivity was selected, as it had shown promising results previously.[13] Transitivity is an aggregate metric computed across the entire network, which is beneficial because it considers every transaction. Also, transitivity has both a directed and an undirected implementation, which should make it possible to assess the information add of edge direction. Whilst it is possible to compute a weighted version of the transitivity, most popular open-source packages don't have an implementation. As a result, an alternative clustering metric is chosen to study the usefulness of edge weights - the average local clustering coefficient.

4.1.2 Hub Analysis

For hub analysis, a method for identifying hubs needed to be established and a set of metrics for these hubs chosen. The obvious choice for identifying hubs was to rank the vertices by their degree and choose the top $x\%$. However, since the BAN is a Cartesian product of inputs and outputs for each transaction, input (output) addresses that occur in transactions with many output (input) addresses automatically get a high degree. This point is discussed further in Section 6.1. So, the decision was made to introduce a second method for selecting hubs, by ordering vertices by their highest transaction count (the number of transactions they appeared in), and choosing the top $x\%$ for $x \in [1, 5, 10]$. Larger values of x give a greater chance of including hubs of interest, such as small or emerging exchanges, although it also increases the

chance of including unwanted vertices, such as Bitcoin mixing services. Choosing multiple values ensured any findings would not be overly dependent on one choice of parameter; this was a design decision used repeatedly. The intuition behind hub analysis is that the flows in and out of Bitcoin exchanges are informative of market conditions.[29, 12] For example, if lots of Bitcoin is flowing into exchanges, and not much is flowing out, this could be indicative of a sell signal and, if maintained over several periods, a bear market. In an unweighted network, this might be represented by a high in/out transaction ratio, indicating hubs are outputs of transactions more often than they are inputs. In a directed network, the net flow can be used instead of counting transactions, giving the additional benefit of quantifying how much Bitcoin is flowing into, or out of, the hubs.

4.2 Constructing the Networks

4.2.1 Representations

The next design decision was to determine which graphs to look at. As explained in Section 2.4, the BAN and the BUN are the two main representations of Bitcoin's transaction network.* These networks can be weighted, directed, and constructed over different frequencies. Longer frequencies often result in huge networks, especially for the BAN. This creates a real engineering challenge, discussed in Chapter 5. The decision was made to study directed and undirected versions of both the BAN and the BUN. The weighted BAN, for which edge weights can be approximated (discussed in Section 2.4.1), was not considered due to slow loading times. The BUN is much smaller than the BAN, making analysis more tractable. However, some information was lost in the address to user mapping, which may be detrimental to the results of the BUN.†

4.2.2 Frequencies

The daily, weekly, and monthly frequencies were considered, to ensure the results were not dependent on one particular choice of parameter.‡ These low frequency networks are well suited to the objective of identifying long-term market trends and conditions; high frequency networks could be too granular to pick up on such trends. Network frequency and the total time period considered determine the style of processing that needs to take place. High frequency networks make for more tractable analysis since they contain fewer vertices and edges, however there are many more of these networks to analyse over a fixed time period. On the other hand, lower frequency networks are larger, making analysis more difficult, but there are fewer of them in total. Traditional parallelisation techniques such as multiprocessing and multithreading are good for analysing lots of small networks, whilst distributed computing techniques such as Apache Spark are better suited to analysing fewer, larger networks.

4.2.3 Filters

Another design decision made was to set a minimum threshold on the size of transaction.§ The reason for this decision was two-fold. Firstly, larger transactions are more likely to have an impact on economic variables such as the price of Bitcoin. Secondly, a filtered network is smaller and easier to analyse. To avoid dependence on a single choice of parameter, the filter was set at different BTC amounts in $[1, 100, 10000]$. Figure A.1 shows how the size of the BAN shrinks dramatically when a filter is applied. One caveat to this approach is that the value of 1BTC (in dollar terms) is significantly more now than it was ten years ago. An alternative option is to calculate the threshold in dollar terms, by retrieving the exchange rate at the time of the network being constructed.

*Another choice is the Bitcoin Lightning Network (BLN), which is a layer 2 scaling solution enabling off-chain transactions for Bitcoin, to increase transaction throughput. It works as follows: two Bitcoin users wanting to transact open a BLN channel, where they partake in a sequence of transactions with one another. These transactions are consolidated into a single transaction between the users and published to the main Bitcoin blockchain when the channel is closed. The decision was taken not to consider BLN transactions in their raw form, since the consolidated versions of the transactions are already included in the other network representations.

†When addresses are thought to be associated with mixing services (for example because there are too many addresses associated with one user) they are marked unresolved.

‡It's worth noting here how transactions are filtered by a given date, or set of dates, to construct the desired network. The timestamp of a transaction is the same as the timestamp of the block that it is included in, because transactions only truly happen when they are confirmed, i.e. when they are included in a block that is appended to the blockchain. So, to construct the network for a set of dates, the transactions are obtained from the blocks that occurred on those dates.

§In practice, the threshold was applied to transaction outputs rather than the transactions themselves. This meant that small outputs inside large transactions were also pruned.

4.3 Establishing Market Conditions

The BB method was used to establish turning points in Bitcoin's price as well as periods of bullishness and bearishness, because it was simple to implement and had previously been applied to cryptocurrency markets. The window size (time period in which only a single turning point can exist) was set at 24 weeks. A short window size was selected on purpose due to the shorter cycles and higher volatility exhibited in cryptocurrency markets compared to traditional financial markets. However, care should be taken when appraising any results given the dependence on this choice of parameter, and suggestions are made in Section 7.1 for how to avoid this limitation in future.

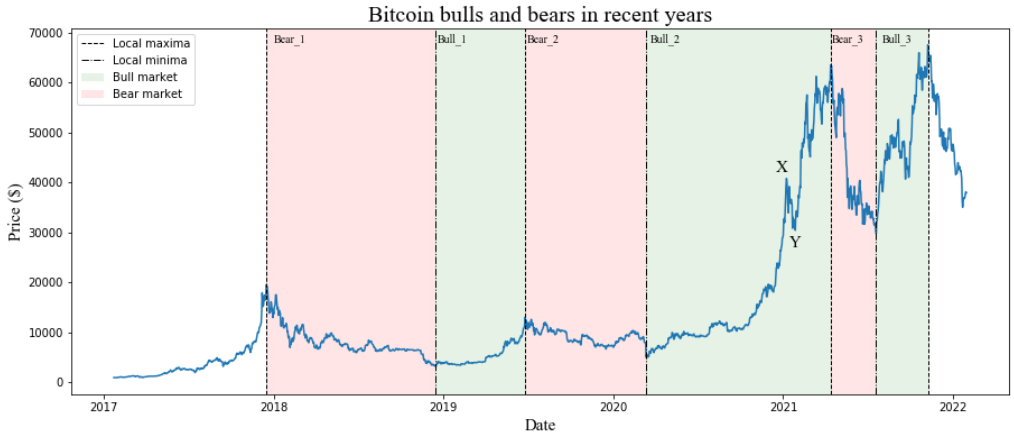

Figure 4.1: Bull and bear markets in Bitcoin in the past five years, identified using the Bry & Boschan method. The unlabelled period at the start (end) appears to be a strong bull (bear) market, however these periods could not be labelled as they do not fall between two turning points. In 2021, a rapid sell-off is followed by a strong bull market. Notice that the local maxima and minima labelled by X and Y respectively are ignored by the Bry & Boschan algorithm, resulting in the intermediate period (a bull market correction) being labelled as a bull market.

4.4 Assessing Relationships Between Metrics and Market Conditions

Correlation and cross-correlation were chosen chosen to provide insights into the relationships between network features and the market price. As discussed in Section 2.5.2, correlation between time series is highly vulnerable to spurious correlation if autocorrelation is not first removed. Therefore, the ADF test was applied to each series and transformations were made to non-stationary series to remove autocorrelation. Also, the Spearman correlation coefficient was chosen over the Pearson coefficient because it is non-parametric and makes less-strong assumptions about the distribution of the time series observations. Correlation was only intended to provide some initial insights, because it gives no information as to the cause and effect relationship between two time series. For determining such relationships, cross-correlation and the Granger causality test were chosen. Some of the series, such as Bitcoin's price, are clearly non-stationary. Therefore, the Toda & Yamamoto procedure for testing for Granger causality between non-stationary series was applied. This test was applied to subsets of the time series during identified periods of bullishness and bearishness, as well as the full time period, to determine if metrics are cross-correlated only under certain market conditions.

Chapter 5

Implementation

This chapter discusses the difficulties of implementing the design from the previous chapter, focusing especially on the construction of networks and the computation of metrics. The results of several experiments undertaken to understand network size and processing time are displayed in Appendix A.2. These results, as well as code snippets in Appendix C are referred to throughout this chapter.

5.1 Network Construction

The first challenge involved loading Bitcoin's raw blockchain data into a suitable format for analysis and consumption. Mesonomics provided access to a structured database containing all of the blockchain data from January 3rd 2009 (the date of the first block) to February 1st 2022. This database resided on a machine with significant compute and memory resources, which was powerful enough to not only store the blockchain data but also process it at scale.* By running both graph extraction and analysis on the same machine as the database, all network latency associated with querying the database was avoided. This benefit in speed, along with the cost saving of utilising provisioned resources, outweighed the benefits of increased compute power that come with a cloud solution such as Amazon Web Services (AWS).

Despite having the blockchain data in a structured format, it was not yet ready for processing the network metrics of interest. First, the desired representations of the network had to be extracted from the database and constructed. Most network packages have built-in methods for constructing a network from a list of edges. Therefore, the first step in constructing the network was to extract the list of edges from the database. The decision was taken to perform all heavy lifting, such as joining tables and filtering, using the database engine, as opposed to loading the data and transforming it in memory.† The final SQL queries (which can be found in Appendix C.4) were designed to filter the edges between two dates and by a minimum threshold amount of BTC, to reduce the amount of data pulled from the database. At a daily frequency with no minimum threshold, the extraction time was approximately five minutes, or just over thirty hours for a full year. This is a significant improvement on the method of extraction used in [13], which took up to two weeks to load the same set of networks. Figure A.3 shows load times for the BAN, filtered by a threshold of 1BTC.

5.2 Metric Computation

After network extraction and construction, the next step was the computation of metrics. The process for clustering analysis and hub analysis are documented in Appendix A.1.

As shown by Figure A.3, the computation time was very metric dependent. The metrics for hub analysis could be computed in linear time by iterating over the edges in the network. However, computing the network transitivity was more difficult. The code for computing transitivity from the popular open-source Python package NetworkX is included in Appendix C.1, along with an interpretation in pseudocode. Notice the nested loop pointed out in the caption of Figure C.2. In the worst-case scenario, when every vertex is connected to every other vertex, this results in a computational complexity of $\mathcal{O}(n^2)$, where

*The machine had 64 cores and 256GB RAM.

†This was possible for all network representations except for the weighted BAN, which had a slightly more involved method for computing edge weights (as discussed in Section 2.4.1). In the end, the additional table joins and processing required for the approximated BAN proved too costly, so the decision was made not to analyse this network.

n is the number of vertices in the network. As a result, calculating the clustering coefficient becomes increasingly difficult for larger networks (shown by Figure A.3).

In practice, it was only possible to use NetworkX to compute network transitivity for networks with up to several million edges. After this, computing the transitivity takes too long and eventually consumers too much memory, so an alternative solution is required. Apache Spark was chosen as the tool of choice, due to its ability to run large scale data engineering jobs using the MapReduce framework.[‡] Software packages built on top of Spark must be able to operate in parallel across resilient distributed datasets (RDDs), the fundamental data unit in Spark.[§] NetworkX is not designed for this. However, there are alternative options for network analysis in Spark, namely GraphX and GraphFrames. The decision was made to use GraphFrames, since it is built on top of Spark DataFrames, which are conceptually equivalent to database tables. This made it easy to load the network straight from the database into a graph. Another decision made was to use the Spark Python API instead of the Scala API, because it made it easier to convert the network analysis scripts written for use with NetworkX into scripts compatible with GraphFrames.[¶]

Whilst GraphFrames has useful built-in methods for computing degree distributions and certain centrality measures, there is no method for computing network transitivity. The algorithm in C.3 was developed to fill this gap, based on a translation from a similar algorithm written in Scala for GraphX,[38], however it is limited to computing only the unweighted, undirected transitivity. Figure A.4 compares the performance of the custom GraphFrames implementation of transitivity to NetworkX's transitivity for increasingly large networks. Interestingly, NetworkX performed better for smaller graphs, probably because the extra overhead of parallelising jobs across workers in Spark was unnecessary for such small graphs. However, the GraphFrames solution scaled significantly better for networks of increasing size. As a result, small to medium size graphs were processed in parallel using Python multiprocessing and NetworkX, whilst large graphs were processed in a distributed fashion with Spark and GraphFrames.

5.3 Identifying Market Conditions

Obtaining the data relating to market conditions was straight-forward in comparison to generating the network metrics. Bitcoin's daily historical prices were downloaded from Yahoo Finance.[24] This data set could be analysed using standard Python libraries.

[‡]Fundamentally, Spark works by splitting up large datasets into many partitions, and executing transformations across them in parallel across multiple workers. This is difficult for graphs, whose metrics are often calculated iteratively across vertices or edges, since they depend on the topology of the entire graph. Spark packages such as GraphX adopt a vertex-cut approach for partitioning graphs.[49] This results in a similar number of edges in each partition, which makes for much more efficient processing than having a similar number of vertices in each partition.

[§]In recent releases of Spark, there are alternative storage mechanisms to RDDs, such as Spark DataSets and DataFrames. Although the implementations are slightly different, they are still distributed across machines and offer the same benefits as RDDs.

[¶]Apache Spark must be configured based on the available hardware. The performance of Spark is determined by the available hardware, suitability of configuration, and effectiveness of key concepts such as partitioning and caching which require fine-tuning. As mentioned, the hardware for this project consisted of a single, but powerful, machine. When running Spark locally in such a manner, Spark distributes jobs across cores instead of across machines. In this circumstance, the total memory allocated to the Spark driver needed to be carefully chosen, so as not to exhaust the memory of the machine.

Chapter 6

Results

6.1 Visualising Networks

This section explores an example Bitcoin transaction network. Figures 6.1 and 6.2 depict the BAN and the BUN respectively for the block at height 625332 on the Bitcoin blockchain, which was mined on April 10th 2020.* What's interesting about this block is that it contains the largest transaction ever recorded - 161,500 BTC, worth over $1 billion at the time. Despite its huge value, the transaction cost approximately $0.68 to process.

6.1.1 The BAN

The first thing of note in Figure 6.1 is the sheer number of vertices, which represent Bitcoin addresses. Considering this is only one block, and a new block is mined every ten minutes, one can only begin to imagine the scale of the weekly and monthly transaction networks. Notice also that the vast majority of vertices are grey, indicating they were involved in no more than five transactions. In fact, 93% of vertices were involved in only one transaction, and 98.5% in fewer than two. This results in a very sparse network, with the majority of vertices being connected to few others.

Some vertices, such as those depicted in the upper right magnification, have very high degree. These vertices could easily be considered to be important within the network, given their high degree. However, as can be seen by the colour coding of the edges, the majority of these vertices are linked only to vertices that were part of the same transaction. For example, the orange cluster represents the Cartesian product of a transaction containing 45 input addresses and 92 output addresses. Since every input is connected to every output, each vertex in this transaction automatically gets a high degree. Despite this, of all the 137 vertices in this transaction, only seven occur in other transactions. Therefore, a vertex having high degree in the BAN does not necessarily mean it is important.

Other techniques for identifying hubs ought to be considered. The number of transactions a vertex appears in is one possibility. The upper left magnification contains two vertices deemed important by this metric, shown by their colour and size. One of these vertices appeared in 70 transactions, which was more than twice as many transactions as any other vertex. Interestingly, almost all of the edges incident to these two vertices were outbound, suggesting that these addresses were sending out lots of Bitcoin (unfortunately it's difficult to see the direction of edges in Figure 6.1 without the ability to zoom). This address could be an exchange, which during this block sent lots of Bitcoin to users in return for fiat currency received off-chain. The lower left magnification shows another address with high transaction count (22 to be precise). The interesting thing about this vertex is that it has both inbound and outbound edges, and its neighbours are spread throughout the network. Whilst this vertex has smaller degree than the vertices in the upper right magnification, it could easily be argued that it has more influence in the network. This justifies the use of transaction count for finding network hubs.

As discussed, the block in Figure 6.1 contains the largest transaction ever recorded. This is shown by the blue vertex displayed in the lower right magnification, which has a self-loop, indicating that it sent Bitcoin to itself. This vertex represents Bitfinex's (an exchange) cold wallet. At the time, Bitfinex confirmed that they topped up their hot wallet with 15,000 BTC from their cold wallet, and the remaining

*The height of a block refers to the number of blocks preceeding it in the blockchain. Since blocks are appended one after another, the blockchain can be thought of as a stack with the first block at the bottom. Hence the word height.

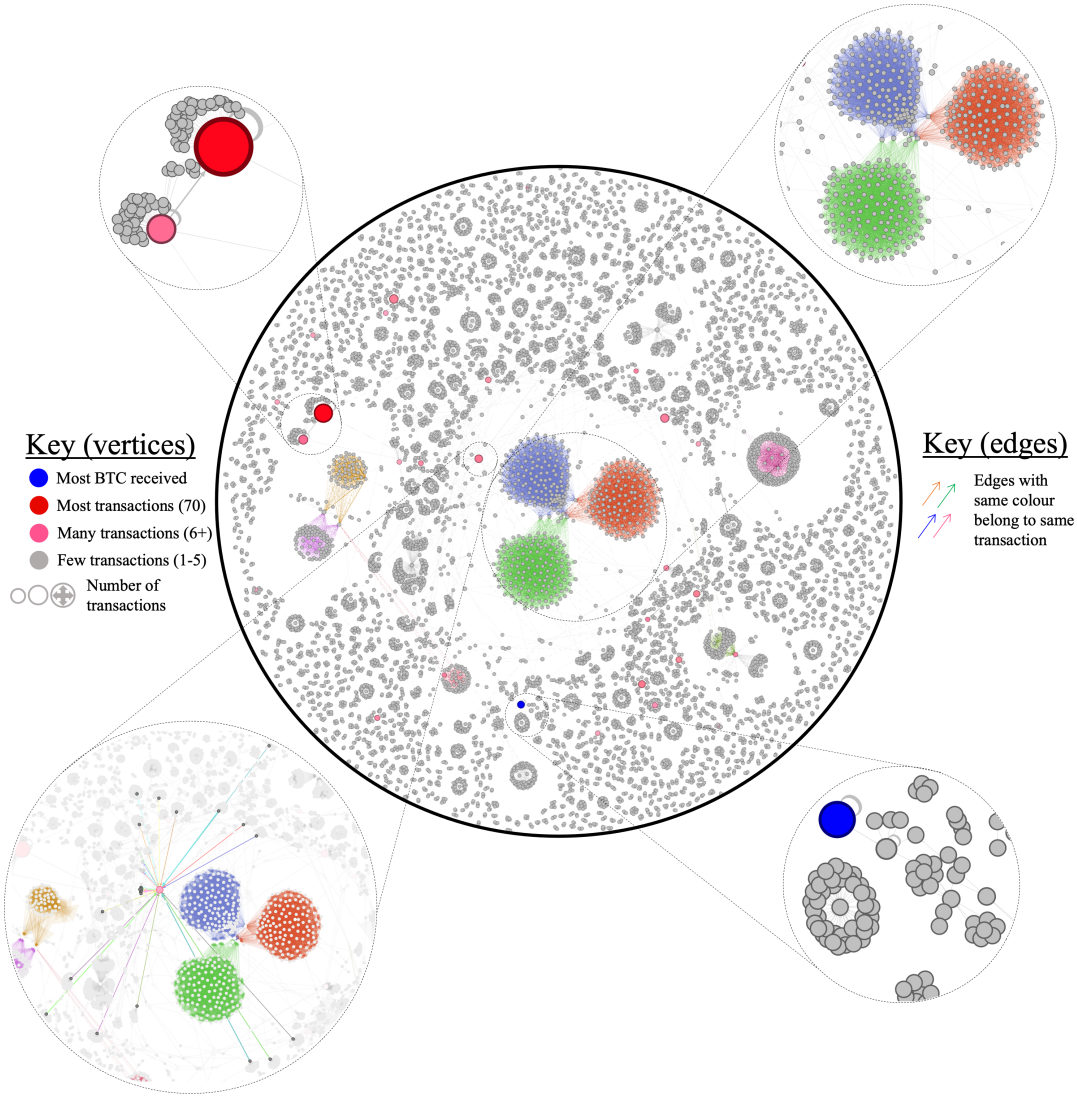

Figure 6.1: The BAN for block at height 625332 in the Bitcoin blockchain.

141,500 BTC was returned to their cold wallet.[†][7] This insight highlights another way of identifying important vertices; by measuring the amount of Bitcoin that flowed through them.

As the size of the window over which the BAN is considered increases, the topology of the transaction network will change. The degree could become a more useful metric for centrality for a network containing more transactions, because the high degree transactions carry less overall weight. Hopefully, this visualisation has highlighted the scale of the BAN as well as the need for careful consideration when determining the importance of vertices. The BUN is considered next.

6.1.2 The BUN

The most obvious difference between the BAN (Figure 6.1) and the BUN (Figure 6.2) is the difference in size. The BUN contains far less vertices than the BAN over the same time period. This is a direct result of the many-to-one mapping of addresses to users, made possible by the heuristics described in Section 2.4.2. The upper left magnification shows the largest transaction, the same as was shown in the BAN, however there is now only one vertex and one edge. A single user sends Bitcoin back to themselves. This makes sense, since both addresses observed in the BAN belong to Bitfinex (corresponding to their hot and cold wallets). Figure 6.3 compares a selection of addresses in the BAN to the corresponding users in the BUN. The central vertices in the BAN have all been mapped to one user, who sends Bitcoin to all (but

[†]Cold wallets store Bitcoin private keys offline, for extra security. The addresses the keys are associated with tend to be used infrequently. Hot wallets, on the other hand, are connected to the internet.

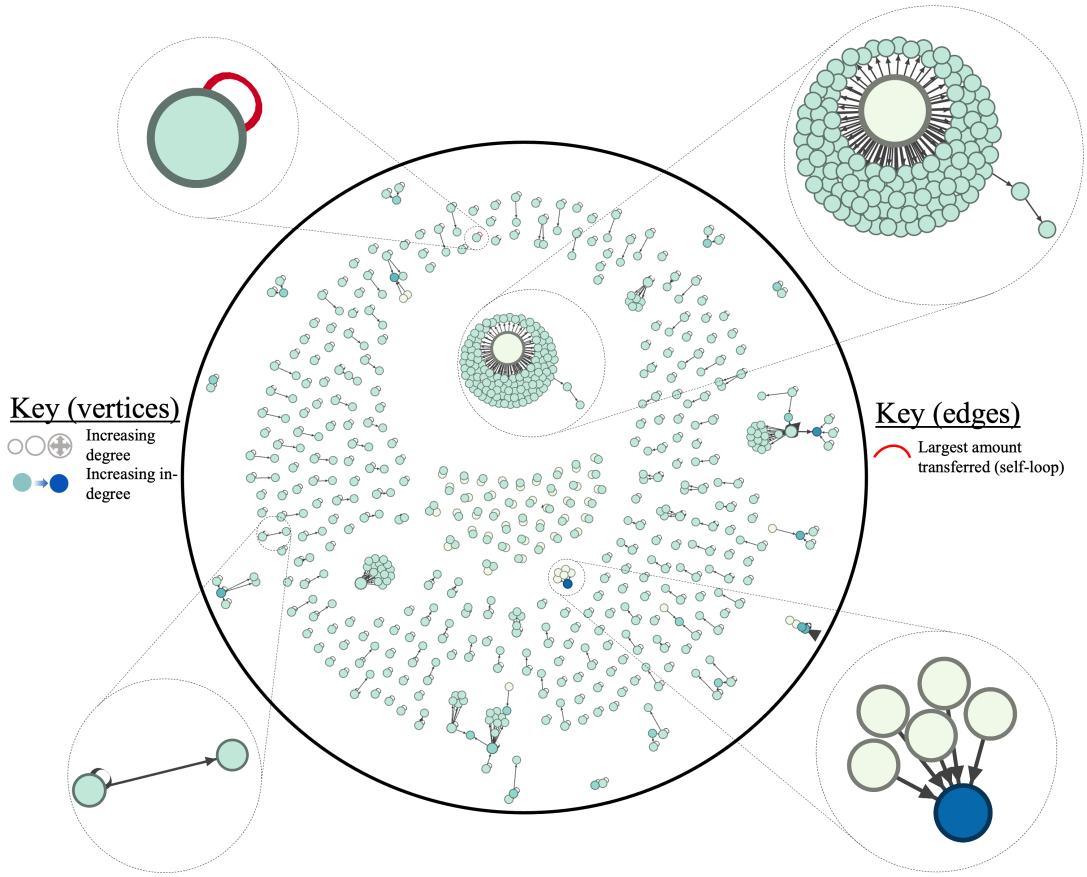

Figure 6.2: The BUN for block at height 625332 in the Bitcoin blockchain.

one) of the other users. From the colour of the edges, it's clear that this all happens in one transaction. The remaining magnifications of Figure 6.2 highlight other topological features, whose interpretations are left as an exercise for the curious reader.

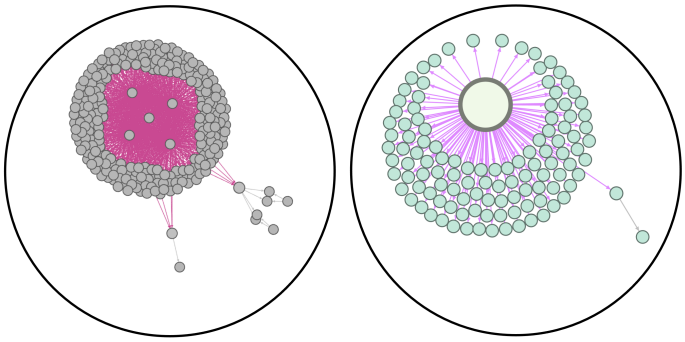

Figure 6.3: Addresses in the BAN (left) mapped to users in the BUN (right). Edges with the same colour belong to the same transaction. The vertex colours are purely aesthetic in this visualisation.

6.2 Clustering

Figure 6.4 displays the transitivity of the daily BAN for different thresholds. The results for the unfiltered network (top left) confirm the findings of Brown.[13] The regime change in transitivity occurs in close proximity to the turning point identified using the BB method. In fact, the turning point marginally

precedes the regime change. This could suggest that a change in price causes a change in clustering, rather than the other way around. As the threshold is increased, the distribution of the transitivity changes significantly. At a threshold of 1BTC (top right), the regime change is still apparent, although it appears to occur several months earlier in January 2021. Interestingly, there was also a slump in Bitcoin's price at this time, although it was not as significant as the May crash. The slump lies in the midst of a strong bull market and so was not identified as a turning point using BB, highlighting the need for careful appraisal of any results given BB's parameters. This sort of event gives weight to the argument set out by Maheu, McCurdy, and Song that additional market states should be considered, such as bull market corrections and bear market rallies.[37] At a threshold of 10,000BTC, the network is so heavily filtered that no complete triples exist for most of the series. The difference in transitivity at different thresholds makes it difficult to justify using transitivity as a metric for filtered networks. Since transitivity is inherently a global metric (a property of the full network), it may be misleading to consider the transitivity of filtered networks.

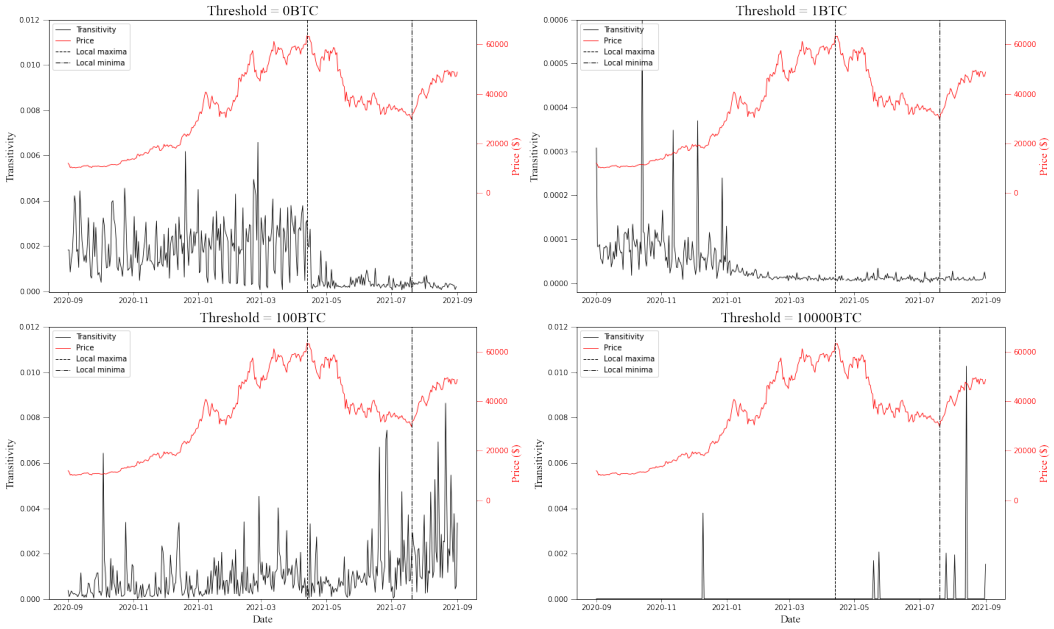

Figure 6.4: The transitivity for different thresholds of the daily BAN, with the turning points from Figure 4.1 also plotted. The threshold is the minimum transaction size applied when constructing the network. There transitivity appears to take very different values for different filters applied to the network.

Figure 6.5 considers the transitivity and average local clustering coefficient of the weekly BUN. Note the difference in network representation and frequency to Figure 6.4. The Figure makes for easy comparison of the two clustering metrics, as well assessment of the impact of edge direction. Firstly, the transitivity (green) and the local average clustering (blue) appear to move in tandem. This makes sense, because they are effectively measuring the same thing - the ratio of triangle to triples in the network. Secondly, the undirected (solid lines) and directed (dashed lines) version of each metric also moves in tandem, with the directed version taking smaller values. This suggests that edge direction does not add much information when considering clustering.

Figure 6.6 shows the pairwise Spearman correlations for each of the time series metrics of interest. Figure B.5 in Appendix B.2 shows the correlations for the BUN. As expected, the directed and undirected versions of the same metric show positive correlation. This is shown by the transitivity metrics for the BAN, and both the transitivity and average local clustering metrics for the BUN. Figure B.6 in Appendix B.2 shows the Spearman correlation between BAN and BUN metrics. Interestingly, the BAN metrics do not display much correlation with the corresponding BUN metrics. For example, the undirected transitivity in the BAN shows no correlation with the undirected transitivity in the BUN. This is bad news, because it means the BUN metrics (which are simpler to compute) do not make good proxies for the for the BAN metrics. Figure B.7 shows this result generalises to other frequencies and thresholds.

Interestingly, none of the network metrics display correlation with Bitcoin's price metrics (price difference, percentage return, log return). As discussed in Section 2.5.2, computing a single correlation value between two time series doesn't help determine a cause and effect relationship. For this, cross-correlation

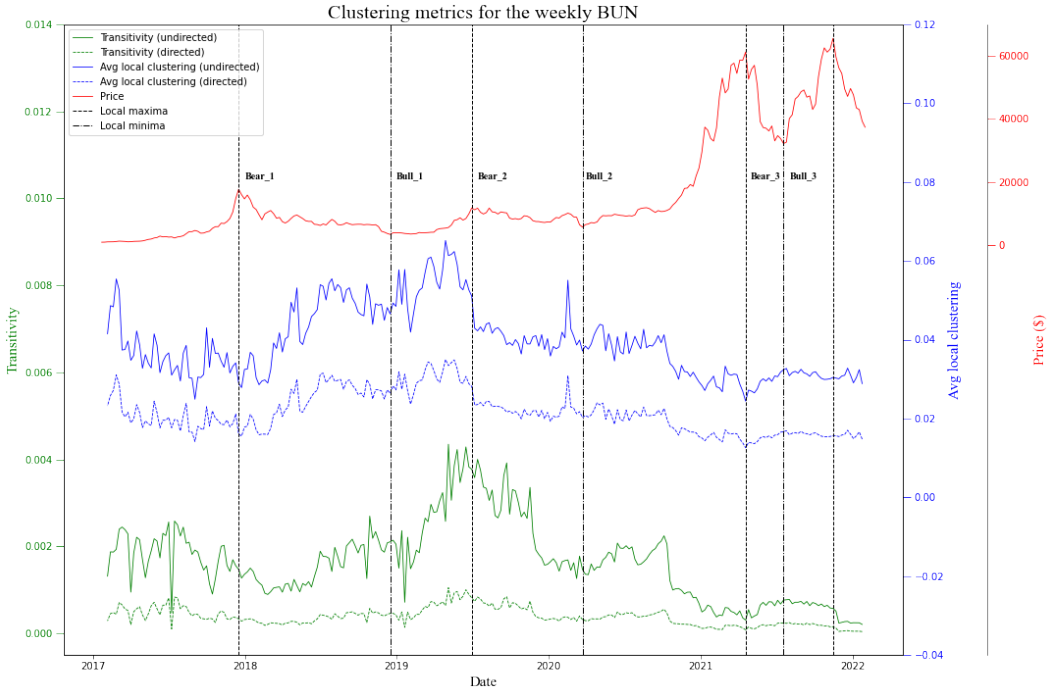

Figure 6.5: The transitivity and average local clustering coefficient calculated at weekly intervals against Bitcoin's average weekly price. Notice that the transitivity and average local clustering coefficient follow similar patterns. The directed versions of each metric also follow a similar pattern, but with smaller values. For the 2017 and late 2020 bull markets, there is a clear downwards trend in the transitivity. Conversely, the transitivity displays an upwards trend during the bear markets of 2018 and mid-2021.

and Granger causality are used. Table 6.1 displays the results of the Granger causality test for the BAN metrics. There is little evidence to suggest that any of the BAN metrics Granger cause Bitcoin's price. The Granger causality test results for the BUN metrics can be found in Appendix B.3, as well as the results of applying the test in the other direction. Interestingly, for the BUN, there is statistically significant evidence that the directed transitivity can be used to forecast Bitcoin's price. A positive result for transitivity in the BUN, but not the BAN, is in line with Brown's observation that it may be more useful to consider triadic interactions at the level of traders, rather than addresses.[13] A positive result for directed transitivity in the BUN, and not undirected transitivity, suggests that edge direction adds valuable information.

	Bull_1	Bull_2	Bull_3	Bear_1	Bear_2	Bear_3	Full timespan
Number of vertices	0.2554	0.6619	0.4858	**0.037**	0.3158	0.4189	0.1299
Number of edges	0.1346	0.7255	0.6149	**0.0065**	0.1223	0.3493	0.0763
Mean degree	0.0684	0.8091	0.4579	**0.0001**	0.0579	0.3699	0.3884
Mean in-degree	0.0684	0.8091	0.4579	**0.0001**	0.0579	0.3699	0.3884
Std in-degree	0.7399	0.2835	0.8592	0.7795	0.4561	0.1927	0.1769
Mean out-degree	0.0684	0.8091	0.4579	**0.0001**	0.0579	0.3699	0.3884
Std out-degree	0.4444	0.3064	0.1481	**0.0007**	0.0348	**0.0193**	0.701
Transitivity (undirected)	**0.0005**	0.2636	0.6049	0.1773	0.2103	**0.0337**	0.9575
Transitivity (directed)	0.0165	0.6044	0.6829	0.8965	0.0963	0.227	0.921
Price	1.0	1.0	1.0	1.0	1.0	1.0	1.0

Table 6.1: Test results for Granger causality of clustering metric on Bitcoin price, for metrics computed on the daily BAN with minimum transaction value of 1BTC. The values in the table are p-values; significant values are in bold. For each metric, the test was applied over seven timespans, corresponding to the periods of bullishness and bearishness identified in Fig 4.1 as well as the full period. Metrics were computed on the daily BAN at a threshold of 1BTC. The Toda & Yamamoto procedue to test for Granger causality was followed.

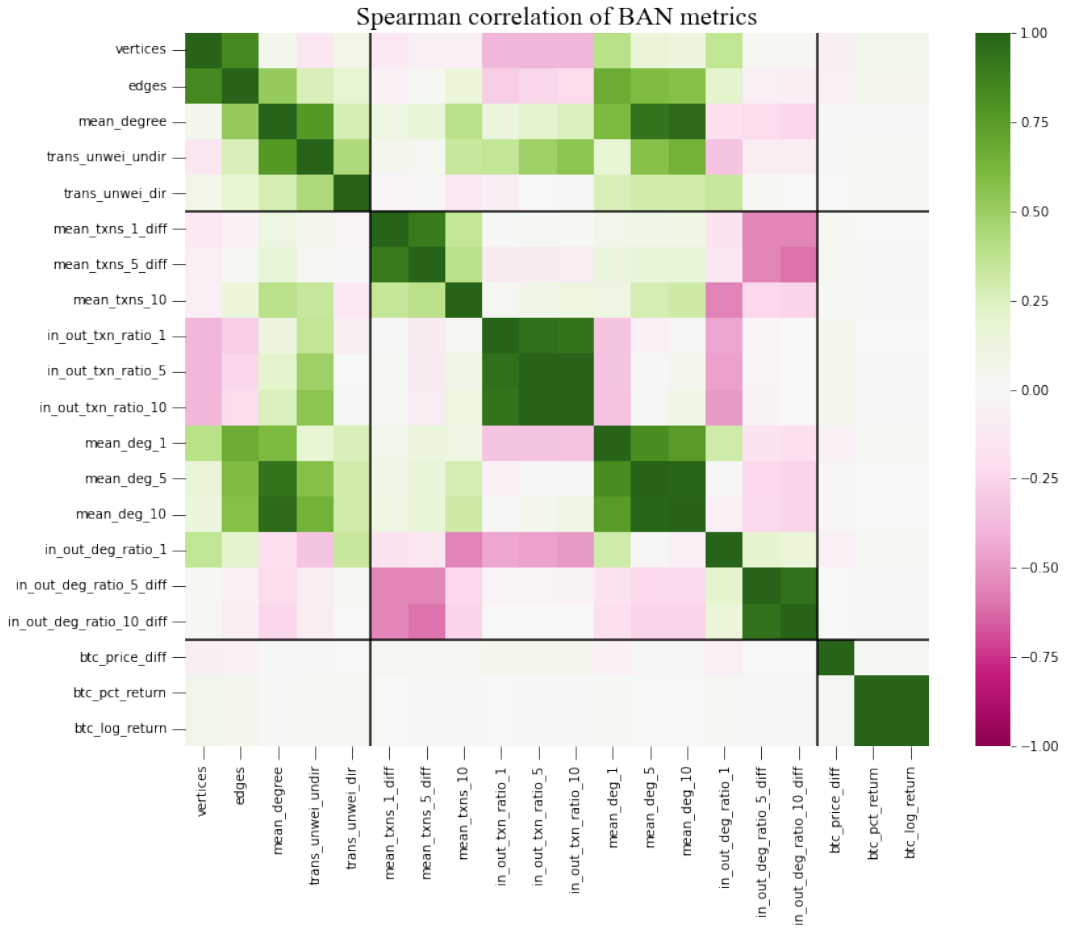

Figure 6.6: Spearman correlation coefficients between the metrics of interest, computed on the daily BAN from 2017 to 2022 at a threshold of 1BTC. Each axis is split into three by the black lines, which indicates the type of metric. The first region corresponds to aggregate metrics, computed across the entire network. The second region corresponds to percentile metrics, computed across the top $x\%$ of vertices by transaction count or degree. The third region corresponds to exogenous metrics, which in this case are various transformations of Bitcoin's price. Metric names containing *_diff* have been differenced (first order) to make the series stationary. Numbers in metric names (e.g. *_1*) describe the percentage of vertices considered as hubs, ordered by transaction count or degree.

6.3 Hub Analysis

Figures 6.7 and 6.8 depict the network hub metrics for hubs filtered by transaction count and degree respectively. For both hub identification methods, the metrics follow similar patterns when computed for the top 1% or 5% of vertices, shown by the tight coupling of solid and dotted lines. This result is confirmed by the strong positive correlations between the 1% and 5% metrics shown in Figures 6.6 and B.5.

Figure 6.7 appears to show a regime change in the transaction in/out ratio. On further analysis, the reason for this became clear. The address to user mapping was performed on transactions from February 2021 onwards, so BUN networks before this date were smaller as fewer of the addresses were mapped. The net flow appears to come from a stationary process centered at zero. By the theory outlined at the start of the paper, the net flow should take on positive values during bear markets, when a lack of confidence drives users to sell their Bitcoin to the exchanges, and negative values during bull markets, when new users receive Bitcoin from the exchanges. However, the opposite of this seems to occur in Figures 6.7 and 6.8. During crashes, such as at the start of *Bear_3* and the end of *Bull_3*, the net flow of the hubs experienced heavy negative spikes. Overall, the net flow for the two hub identification methods follow very similar patterns, suggesting that the different hub identification techniques result in many of the same vertices being selected. This is difficult to validate without an address level inspection of the

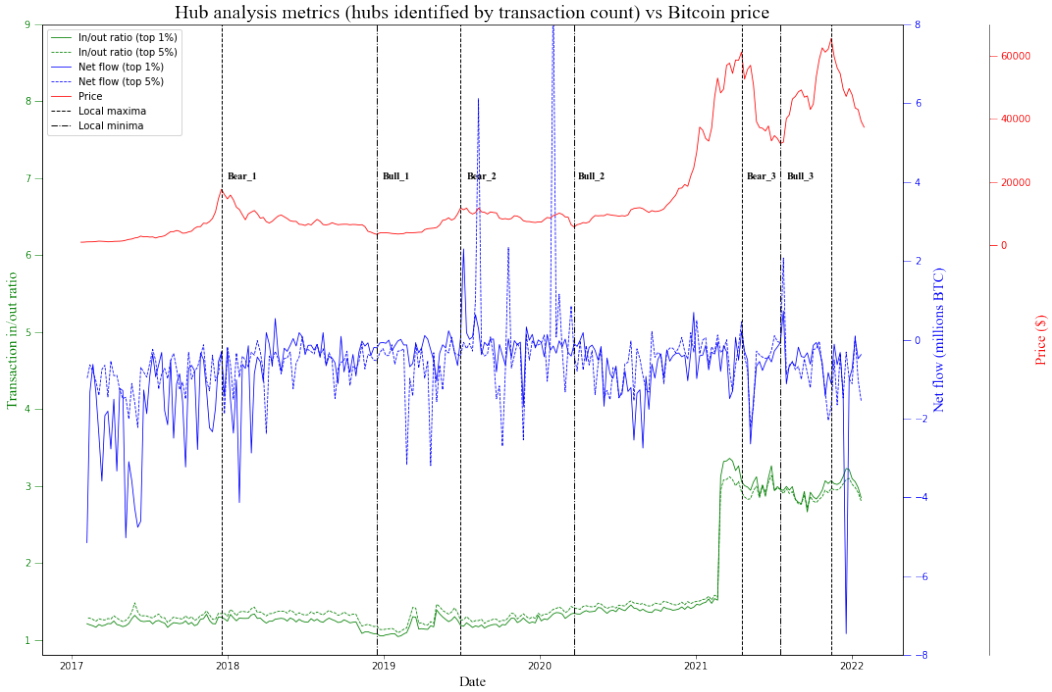

Figure 6.7: The in/out ratio of transactions and net flow for hubs in the BUN, identified by transaction count. Results for the top 1% and 5% of vertices are plotted. Metrics computed at daily frequency and with minimum transaction amount of 1BTC.

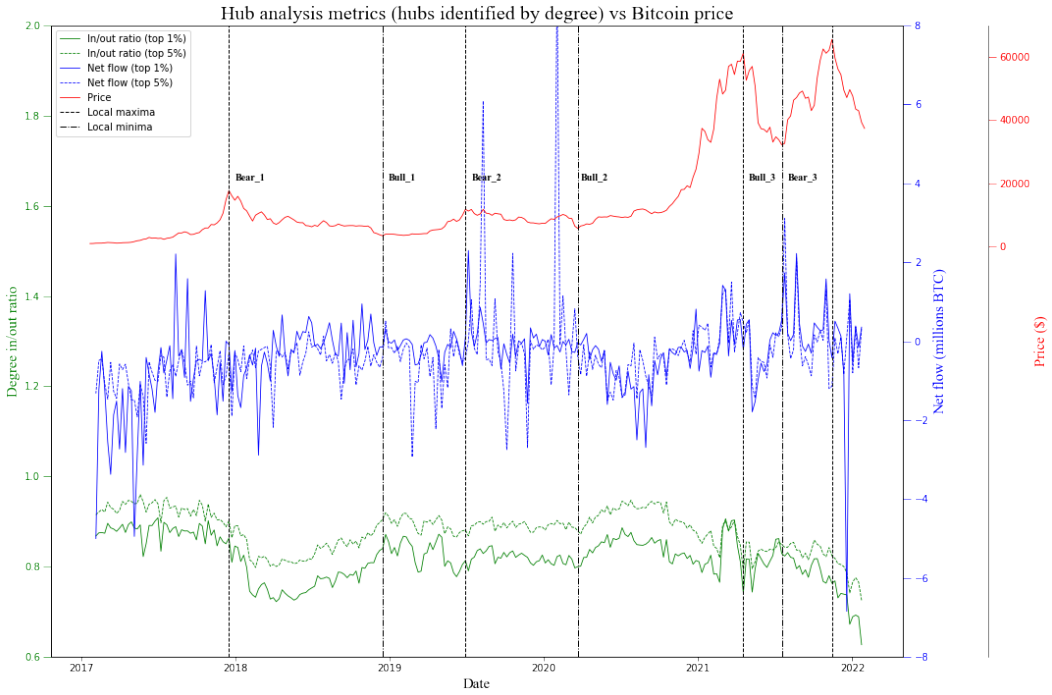

Figure 6.8: The in/out ratio of degree and net flow for hubs in the BUN, identified by degree. Results for the top 1% and 5% of vertices are plotted. Metrics computed at daily frequency and with minimum transaction amount of 1BTC.

hubs.

From Figure 6.8, the in/out degree ratio appears somewhat less stationary than the net flow, displaying elements of trend and seasonality, as well as moderate signs of synchrony with Bitcoin's price. Table 6.2 displays the results of the Granger causality test for hub metrics and Bitcoin's price in the BAN. Table

B.2 shows results of the same test applied to the BUN. The in/out degree ratio for the top 1% of vertices in the BAN showed statistically significant evidence of Granger causality on Bitcoin's price, and the top 5% of vertices in the BUN, suggesting that a changing degree distribution of network hubs precedes a change in Bitcoin price. The positive result for this metric, but not the net flow, suggests that edge weight does not add much information for determining market conditions.

	Bull_1	Bull_2	Bull_3	Bear_1	Bear_2	Bear_3	Full timespan
Mean in-transactions (1%)	0.8023	0.8669	0.2206	0.0916	0.0932	0.1308	0.1045
Mean out-transactions (1%)	0.3189	0.4384	0.5890	0.5100	**0.0045**	0.1175	0.2879
In/out transaction ratio (1%)	0.4961	0.683	0.4433	0.4565	0.2573	0.1213	0.6997
In/out degree ratio (1%)	**0.0438**	0.8118	0.3574	**0.0088**	0.0983	0.0564	**0.0069**
Mean in-transactions (5%)	0.1782	0.7835	0.2428	0.0739	**0.0077**	0.1110	0.1525
Mean out-transactions (5%)	0.1616	0.3670	0.3708	0.4654	**0.0324**	0.1915	0.3935
In/out transaction ratio (5%)	0.4031	0.7572	0.4421	0.396	0.1561	0.12	0.8277
In/out degree ratio (5%)	0.4238	0.9713	0.8421	**0.0455**	0.0693	0.5687	0.0822

Table 6.2: Test results for Granger causality of hub analysis metrics on Bitcoin price, for metrics computed on the daily BAN with minimum transaction value of 1BTC. The values in the table are p-values; significant values are in bold. For each metric, the test was applied over seven timespans, corresponding to the periods of bullishness and bearishness identified in Fig 4.1 as well as the full period. Metrics were computed on the daily BAN at a threshold of 1BTC. The Toda & Yamamoto procedue to test for Granger causality was followed.

Chapter 7

Conclusion

In this study, Bitcoin's transaction network has been placed under the microscope and tested for information about long-term swings in the market price. It was of particular interest to discover which, if any, features of the network contained information about market conditions, and whether the weight and direction of edges could enhance this information. Past research had shown that clustering and the actions of network hubs were correlated with significant events such as market shocks and crashes, so these areas were selected for analysis, with the goal of detecting patterns that could be related to periods of price rise and decline. To make this possible, methods were first established for identifying Bitcoin market conditions and hubs in the transaction network. After this, a sequence of tests for correlation and cross-correlation were used to assess the cause-and-effect relationships between the network properties and Bitcoin's price, across several periods of identified price rise and fall.

For transitivity, a regime change for the daily BAN was observed during the May 2021 price crash, in accordance with previous results. However, no such change was observed in the transitivity of the BAN when filtered by different transaction amounts, suggesting that transitivity is only a useful metric for determining market conditions when a complete picture of the transaction network is available. No evidence of cross-correlation between the transitivity of the BAN and Bitcoin's price was found, however the transitivity of the BUN did display cross-correlation, suggesting that transitivity is more useful for determining market conditions when considered at the user level. In particular, this positive result was for the directed transitivity, suggesting that edge direction does add valuable information. As for network hubs, the in/out degree ratio was shown to contain useful information for forecasting Bitcoin's price in both the BAN and the BUN, supporting the hypothesis that a changing degree distribution of network hubs (to accommodate for changes in demand) precedes a change in Bitcoin price. The net flow, on the other hand, was not shown to be useful for forecasting Bitcoin prices. Given that the net flow depends on edge weights, and the in/out degree ratio does not, this suggests that edge weight does not add useful information for predicting market conditions. However, one must consider that these results came from representations of the BUN filtered by a minimum transaction size, which could have significantly obscured the true net flow of hubs, especially those that deal in lots of small transactions. Interestingly, a distinct lack of correlation was observed between BAN and BUN metrics, suggesting that the BUN should not be used as a proxy for the BAN.

Due to the sheer size of Bitcoin's transaction network, the most significant challenges of the project were constructing the network and generating the time series of network metrics. These difficulties were highlighted throughout the paper, with the design decisions and engineering solutions highlighted in Chapters 4 and 5 resulting in several unforeseen contributions. Firstly, the queries for extracting four different representations of the transaction network have been included in Figure C.4, including for the novel approximately weighted Cartesian BAN introduced in Section 2.4.1. Secondly, alternative mechanisms were established for analysing the networks at scale depending on the size of the network. The recommended solution for smaller networks was to process them in-memory using a multi-processing solution. For larger networks, the solution was to utilise the open-source distributed computing software, Apache Spark. The Spark package for network analysis, GraphFrames, did not contain code for computing network transitivity, and thus required extension. The code in Figure C.3 was written and tested, and has been made available on Github.[53]

Overall, this paper has given a comprehensive overview of the research to date, identified the features of the transaction network most likely to be related to Bitcoin price swings, established a mechanism for identifying Bitcoin market conditions, provided engineering solutions for analysing networks at scale,

and provided insights into the evolution of the transaction network over the past five years. Several suggestions for further research are made in Section 7.1.

7.1 Future Work

7.1.1 Alternate Methods for Identifying Market Conditions

Firstly, this paper relied heavily on the BB method for finding turning points and labelling market conditions. Despite the justifications for this choice outlined in Section 3.4, dependence on a single method and a single set of parameters was a significant limitation. The end results, especially those of the Granger causality test, will have been heavily influenced by these choices. Other methods for establishing market conditions exist. The simplest of these is a naive moving average model that labels bull and bear markets based on a window of recent returns. Another simple method is to detect outliers in the log returns and label periods of time with a high density of outliers accordingly. For a more advanced method, a Markov Switching model (described in Section 3.4) could be deployed.[36] Finally, one could use the Lunde & Timmermann filtering approach for finding peaks and troughs, which has been applied to cryptocurrencies in the past.[35, 61] To assess the performance of each approach, it would be useful to have some gold standard data, for example data labelled by financial experts or cryptocurrency analysts. This would enable the comparison of approaches and would significantly improve the reliability of results.

7.1.2 Inspect Known Addresses

In Figures 6.1 and 6.2, the largest transaction to have ever occurred in Bitcoin was visualised. This information was easy to find on the Internet due to the publicity of transactions. It's also possible to gather information about specific Bitcoin addresses and who they belong to. This is certainly the case for known Bitcoin exchanges, who publicise their addresses by necessity to enable wallet applications to integrate with them. Two sites containing information about known Bitcoin exchanges are left here for reference.[57, 10] This information could be used in two ways. Firstly, over a long enough period any suitable hub identification method should be expected to include the addresses of large Bitcoin exchanges. Knowledge of specific exchange addresses could be used to validate the hub identification method. Secondly, properties of these specific addresses could be inspected. Section 6.3 on hub analysis considered the in/out degree ratio, in/out transaction ratio, and net flow of network hubs. It would be an interesting study to consider these metrics on an address by address basis for known exchanges.

7.1.3 Employ a Machine Learning Approach

This paper made use of time series analysis to study the cross-correlation of network metrics with the price of Bitcoin, and ran separate Granger causality tests for time periods grouped by labelled market condition. An alternative and perhaps more reliable method for utilising this labelled data would be to employ a machine learning classifier, such as logistic regression in the case of two market states (bullish and bearish). The input data for the classifier would be the network metrics, and the target variable would be the market state label. Comparing the accuracy of such a model to a baseline classifier trained only on historical price data would enable the assessment of the value of information added by the network metrics. Non-linear classifiers, such as a nearest neighbours approach or a neural network, could also be experimented with. Of course, the performance of a machine learning model is determined by the quality of data provided to it. Therefore, using gold labelled data would be essential for this approach.

7.1.4 Alternative Network Metrics

Graph theory is a broad field with a huge range of unexplored properties. As discussed in Section 3.3.3, TDA has been a hot topic of research lately and topological features of Bitcoin's transaction network, such as network motifs, have shown promising results for medium to long-term price prediction.[32, 3, 2] Flow dynamics of Bitcoin through the network is another area that could yield interesting results. These areas are highlighted as needing further investigation in relation to long-term movements in Bitcoin's price.

7.1.5 Use the Approximate BAN

In this paper, a method for approximating the weights of edges in the Cartesian BAN was proposed (Section 2.4.1) but not used. The approximately weighted BAN incorporates information about the amount of Bitcoin transferred, yet doesn't suffer from the problems of missing or unmapped data observed in the BUN. It is therefore the most complete representation of Bitcoin's transaction network in terms of utilised information, and as a result could provide insights that the unweighted BAN and BUN were not capable of. Figure C.4 contains the SQL query for extracting the required information, Equation 2.3 contains the formula for approximating the weights, and the code for computing the approximated weights is made available on Github.[53]

Bibliography

[1] Bitcoin becomes official currency in central african republic. *BBC*, April 2022. https://bbc.in/3d4JiiI Accessed: 2022-07-05.

[2] Nazmiye Ceren Abay, Cuneyt Gurcan Akcora, Yulia R Gel, Murat Kantarcioglu, Umar D Islambekov, Yahui Tian, and Bhavani Thuraisingham. Chainnet: Learning on blockchain graphs with topological features. In *2019 IEEE international conference on data mining (ICDM)*, pages 946–951. IEEE, 2019.

[3] Cuneyt G Akcora, Asim Kumer Dey, Yulia R Gel, and Murat Kantarcioglu. Forecasting bitcoin price with graph chainlets. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 765–776. Springer, Springer International Publishing, 2018.

[4] Laura Alessandretti, Abeer ElBahrawy, Luca Maria Aiello, and Andrea Baronchelli. Anticipating cryptocurrency prices using machine learning. *Complexity*, 2018:1–16, November 2018.

[5] Andreas M Antonopoulos. *Mastering Bitcoin: Programming the open blockchain*. O'Reilly Media, Inc., 2017.

[6] Andreas M Antonopoulos and Gavin Wood. *Mastering ethereum: building smart contracts and dapps*. O'reilly Media, 2018.

[7] Paulo Ardoino. Yep, we refilled hot wallet with 15k, rest went back to original address. https://twitter.com/paoloardoino/status/1248702664650772480, April 2020.

[8] Annika Baumann, Benjamin Fabian, and Matthias Lischke. Exploring the bitcoin network. *WEBIST*, 2014:369–374, 2014.

[9] Dirk G Baur, Kihoon Hong, and Adrian D Lee. Bitcoin: Medium of exchange or speculative assets? *Journal of International Financial Markets, Institutions and Money*, 54:177–189, 2018.

[10] BitInfoCharts. Bitcoin rich list. https://bit.ly/3qpQCIH. Accessed: 2022-09-05.

[11] Pawel Boguslawski. *Modelling and analysing 3d building interiors with the dual half-edge data structure*. University of South Wales (United Kingdom), 2011.

[12] Alexandre Bovet, Carlo Campajola, Jorge F. Lazo, Francesco Mottes, Iacopo Pozzana, Valerio Restocchi, Pietro Saggese, Nicoló Vallarano, Tiziano Squartini, and Claudio J. Tessone. Network-based indicators of bitcoin bubbles, 2018.

[13] Alexander Brown. Network analysis of cryptocurrency transactions. Master's thesis, Department of Computer Science, University of Bristol, May 2022.

[14] Gerhard Bry and Charlotte Boschan. Front matter to "cyclical analysis of time series: Selected procedures and computer programs". In *Cyclical analysis of time series: Selected procedures and computer programs*, pages 13–2. National Bureau of Economic Research, Inc, 1971.

[15] Vitalik Buterin et al. A next-generation smart contract and decentralized application platform. *White Paper*, 3(37):2–1, 2014.

[16] Eng-Tuck Cheah and John Fry. Speculative bubbles in bitcoin markets? an empirical investigation into the fundamental value of bitcoin. *Economics letters*, 130:32–36, 2015.

[17] Shiu-Sheng Chen. Predicting the bear stock market: Macroeconomic variables as leading indicators. *Journal of Banking & Finance*, 33(2):211–223, 2009.

[18] Nicolas Christin. Traveling the silk road: A measurement analysis of a large anonymous online marketplace. In *Proceedings of the 22nd international conference on World Wide Web*, pages 213–224, 2013.

[19] CoinMarketCap. Bitcoin to usd chart. https://bit.ly/3B54QUj. Accessed: 2022-07-05.

[20] CoinMarketCap. Ethereum to usd chart. https://bit.ly/3xaydDs. Accessed: 2022-07-05.

[21] Luke Conway. All the countries where bitcoin adoption is being considered. *The Street*, June 2021.

[22] Francesco Maria De Collibus, Alberto Partida, Matija Piškorec, and Claudio J Tessone. Heterogeneous preferential attachment in key ethereum-based cryptoassets. *Frontiers in Physics*, page 568, 2021.

[23] NetworkX Developers. NetworkX. Github. https://bit.ly/3TW5Y5d. Accessed: 2022-08-21.

[24] Yahoo Finance. Bitcoin usd (btc-usd). https://finance.yahoo.com/quote/BTC-USD/history/. Accessed: 2022-07-10.

[25] Michael Fleder, Michael S. Kester, and Sudeep Pillai. Bitcoin transaction graph analysis, 2015.

[26] Sean Foley, Jonathan R Karlsen, and Tālis J Putniņš. Sex, drugs, and bitcoin: How much illegal activity is financed through cryptocurrencies? *The Review of Financial Studies*, 32(5):1798–1853, 2019.

[27] David Garcia, Claudio J Tessone, Pavlin Mavrodiev, and Nicolas Perony. The digital traces of bubbles: feedback cycles between socio-economic signals in the bitcoin economy. *Journal of the Royal Society Interface*, 11(99):20140623, June 2014.

[28] David E. Giles. Testing for granger causality. https://bit.ly/3RRZbYk, Apr 2011. Accessed: 2022-08-26.

[29] Alex Greaves and Benjamin Au. Using the bitcoin transaction graph to predict the price of bitcoin. 8:416–443, 2015.

[30] Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[31] Dániel Kondor, Márton Pósfai, István Csabai, and Gábor Vattay. Do the rich get richer? An empirical analysis of the bitcoin transaction network. *PloS one*, 9(2):e86197, 2014.

[32] Yitao Li, Umar Islambekov, Cuneyt Akcora, Ekaterina Smirnova, Yulia R Gel, and Murat Kantarcioglu. Dissecting ethereum blockchain analytics: What we learn from topology and geometry of the ethereum graph? In *Proceedings of the 2020 SIAM international conference on data mining*, pages 523–531. SIAM, 2020.

[33] Jiaqi Liang, Linjing Li, and Daniel Zeng. Evolutionary dynamics of cryptocurrency transaction networks: An empirical study. *PloS one*, 13(8):e0202202, 2018.

[34] Oscar Lopez. El salvador adopts bitcoin as currency. *New York Times*, September 2021. https://nyti.ms/3xaeILj Accessed: 2022-07-06.

[35] Asger Lunde and Allan Timmermann. Duration dependence in stock prices: An analysis of bull and bear markets. *Journal of Business & Economic Statistics*, 22(3):253–273, 2004.

[36] John M Maheu and Thomas H McCurdy. Identifying bull and bear markets in stock returns. *Journal of Business & Economic Statistics*, 18(1):100–112, 2000.

[37] John M Maheu, Thomas H McCurdy, and Yong Song. Components of bull and bear markets: bull corrections and bear rallies. *Journal of Business & Economic Statistics*, 30(3):391–403, 2012.

[38] Michael Malak and Robin East. *Spark GraphX in action*. Simon and Schuster, 2016.

[39] Wolfram MathWorld. Königsberg bridge problem. https://bit.ly/3L1XyoS. Accessed: 2022-06-28.

[40] 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.

[41] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[42] Eddie Mitchell. How many people use bitcoin? *Bitcoin Market Journal*, November 2020. https://bit.ly/2NQj7tF Accessed: 2022-08-16.

[43] Amir Pasha Motamed and Behnam Bahrak. Quantitative analysis of cryptocurrencies transaction graph. *Applied Network Science*, 4(1):1–21, 2019.

[44] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2009.

[45] Mark Newman. *Networks: An Introduction*. Oxford university press, 2018.

[46] Adrian R Pagan and Kirill A Sossounov. A simple framework for analysing bull and bear markets. *Journal of applied econometrics*, 18(1):23–46, 2003.

[47] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[48] Dorit Ron and Adi Shamir. Quantitative analysis of the full bitcoin transaction graph. In *International Conference on Financial Cryptography and Data Security*, pages 6–24. Springer, 2013.

[49] Apache Spark. GraphX programming guide. https://bit.ly/2REBLqa. Accessed: 2022-07-20.

[50] Statista. Bitcoin market dominance. https://bit.ly/3ew4WMZ. Accessed: 2022-07-06.

[51] Statista. Daily number of bitcoin transactions. https://bit.ly/3QpVPed. Accessed: 2022-08-15.

[52] Digital Currencies Team. Central bank digital currency: opportunities, challenges and design. Technical report, Bank of England, 2020.

[53] thelk22. Bitcoin network analysis. Github. https://github.com/thelk22/bitcoin-network-analysis. Accessed: 2022-09-09.

[54] Hiro Y Toda and Taku Yamamoto. Statistical inference in vector autoregressions with possibly integrated processes. *Journal of econometrics*, 66(1-2):225–250, 1995.

[55] Andrew Urquhart. The inefficiency of bitcoin. *Economics Letters*, 148:80–82, 2016.

[56] Nicoló Vallarano, Claudio J Tessone, and Tiziano Squartini. Bitcoin transaction networks: an overview of recent results. *Frontiers in Physics*, 8:286, 2020.

[57] WalletExplorer. Walletexplorer.com: Smart bitcoin block explorer. https://www.walletexplorer.com/. Accessed: 2022-09-05.

[58] Jiajing Wu, Jieli Liu, Yijing Zhao, and Zibin Zheng. Analysis of cryptocurrency transactions from a network perspective: An overview. *Journal of Network and Computer Applications*, 190:103139, 2021.

[59] David Yermack. Is bitcoin a real currency? an economic appraisal. In *Handbook of digital currency*, pages 31–43. Elsevier, 2015.

[60] Jesse Yli-Huumo, Deokyoon Ko, Sujin Choi, Sooyong Park, and Kari Smolander. Where is current research on blockchain technology?—a systematic review. *PloS one*, 11(10):e0163477, 2016.

[61] Yuanyuan Zhang, Stephen Chan, Jeffrey Chu, and Hana Sulieman. On the market efficiency and liquidity of high-frequency cryptocurrencies in a bull and bear market. *Journal of Risk and Financial Management*, 13(1):8, 2020.

Appendix A

Data Engineering

A.1 Methods

Clustering analysis required the following steps:

1. Compute the transitivity of the entire network.
2. Compute the clustering coefficient for each vertex.
3. Compute the average clustering coefficient of all vertices.
4. Return the transitivity and clustering coefficient.

Hub analysis required the following steps:

1. Compute the transaction count, in-transaction count, out-transaction count, in/out transaction count ratio, and net flow for each vertex.*
2. Sort the vertices by transaction count and select the top $x\%$ for $x \in [1, 5, 10]$.
3. Return the average in/out transaction count ratio and net flow the selected vertices.
4. Repeat the above steps with degree in place of transaction count.

A.2 Experiments

Displayed here are the results of several experiments conducted to better understand the size of the networks and the processing times of various solutions.

A.2.1 Network Size

Figures A.1 and A.2 show the results of an experiment conducted to better understand the size of the networks.

A.2.2 Network Load and Processing Times

Figure A.3 shows the results of an experiment conducted to better understand the load and processing times of the networks.

A.2.3 NetworkX vs GraphFrames

Figure A.4 shows the results of an experiment conducted to compare the custom GraphFrames solution with the traditional NetworkX solution.

*These metrics depend on a complete picture of the graph, so they must be calculated before filtering.

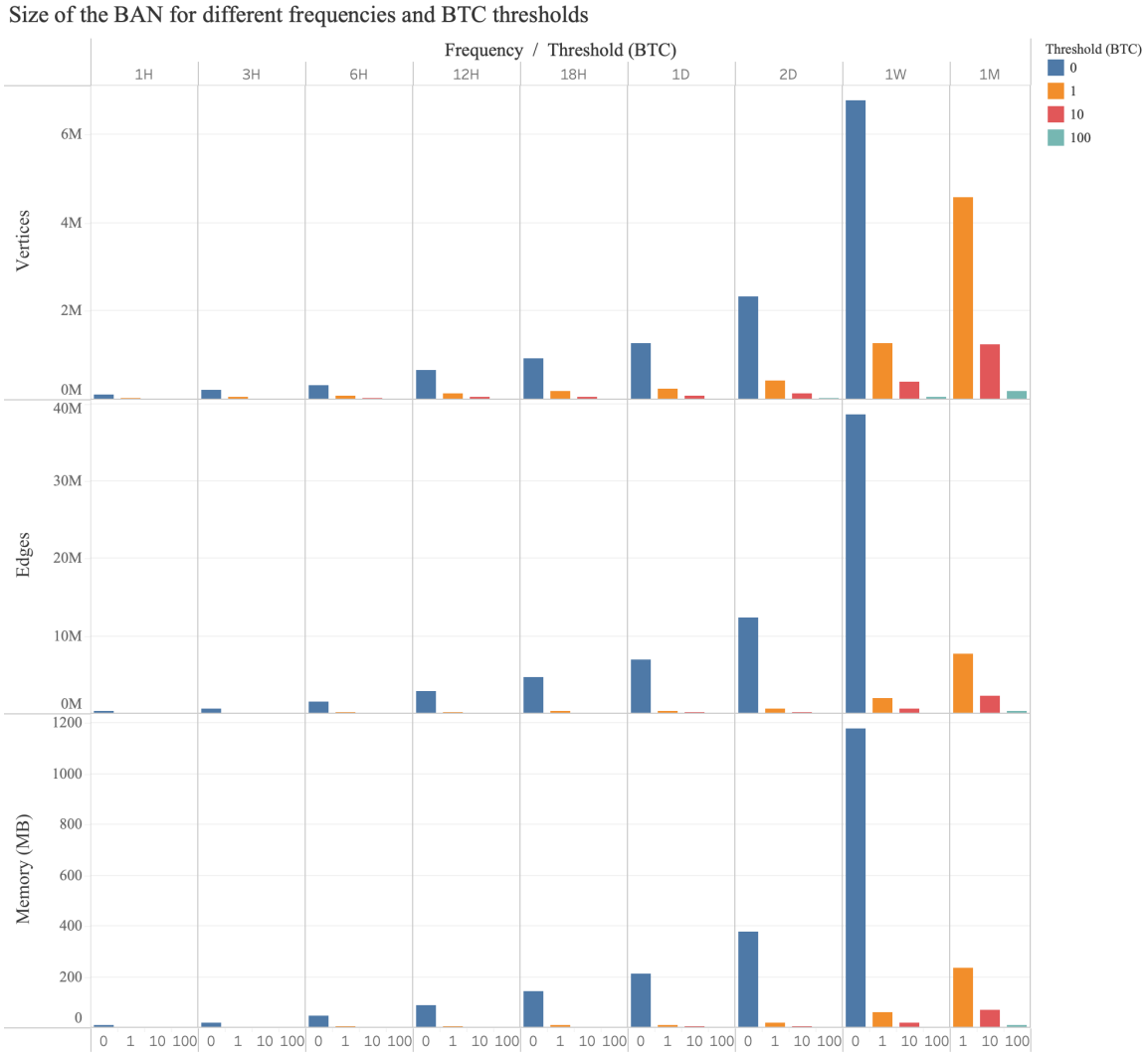

Figure A.1: The size of the BAN for different frequencies (window of time in which to consider blocks) and thresholds (minimum size of transaction to consider). The memory is the size of the in-memory object storing the network. Notice that the size of the network increases for lower frequencies; the weekly BAN contains approximately 7 million vertices and 40 million edges. Notice also that the size of the network decreases for higher thresholds of BTC; filtering the weekly BAN by just 1BTC reduces the number of vertices by over 5 million and the number of edges by over 35 million. At a threshold of 0BTC, the monthly BAN was too large to load in a reasonable amount of time, hence the omission of data.

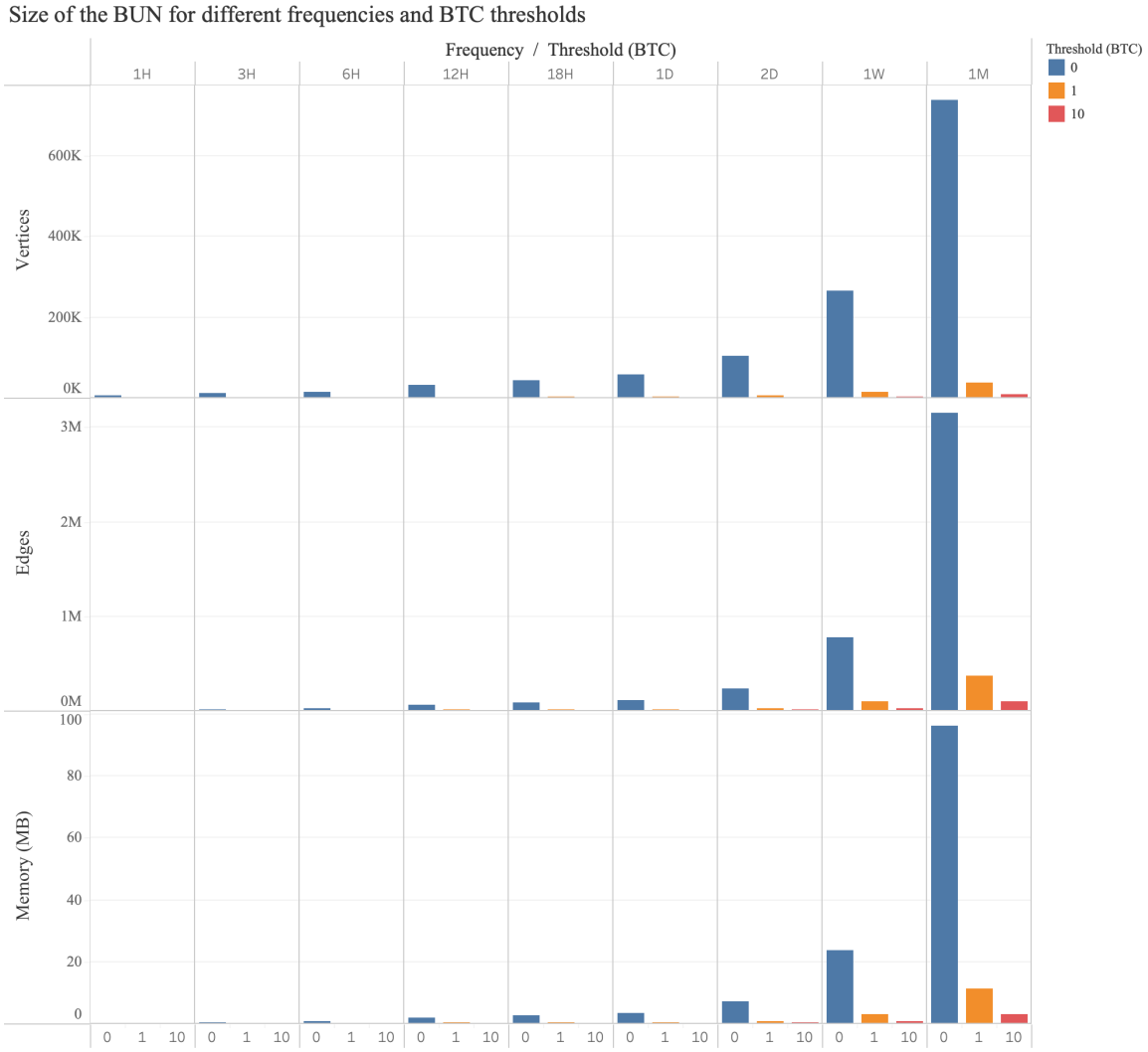

Figure A.2: The size of the BUN for different frequencies and thresholds. Frequencies, thresholds, and memory are described in the caption of Figure A.1, and the same values of these parameters are used. Notice that, compared to the BAN, the BUN is significantly smaller. The weekly BUN with no threshold has approximately 750 thousand vertices and 3.1 million edges; the BAN with the same parameters has 7 million vertices and 40 million edges.

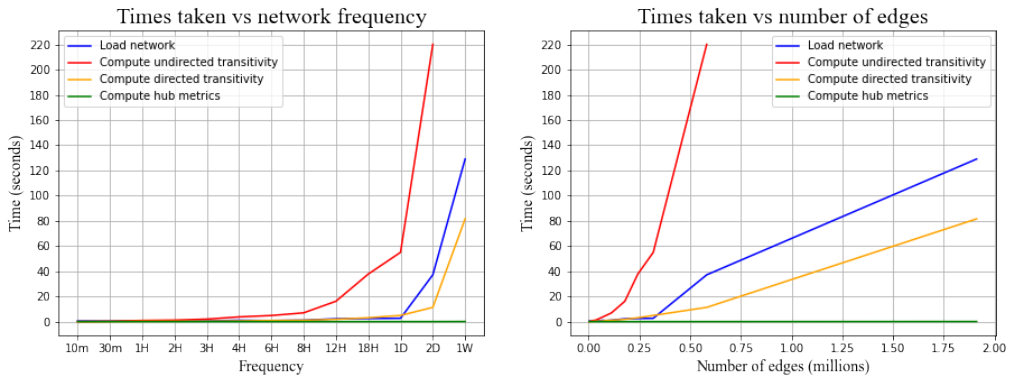

Figure A.3: Average load and processing times for the BAN filtered at a threshold of 1BTC. The left hand plot displays the time taken to load the network from the database (blue line) and compute the metrics of interest (red, orange, and green lines) for different frequencies of BAN. The plot on the right shows the same data plotted against the number of edges in the corresponding network, which has the effect of making the the horizontal axis linear. Notice how, depsite the linearity of the horizontal axis, the time taken to compute the undirected transitivity still increases exponentially.

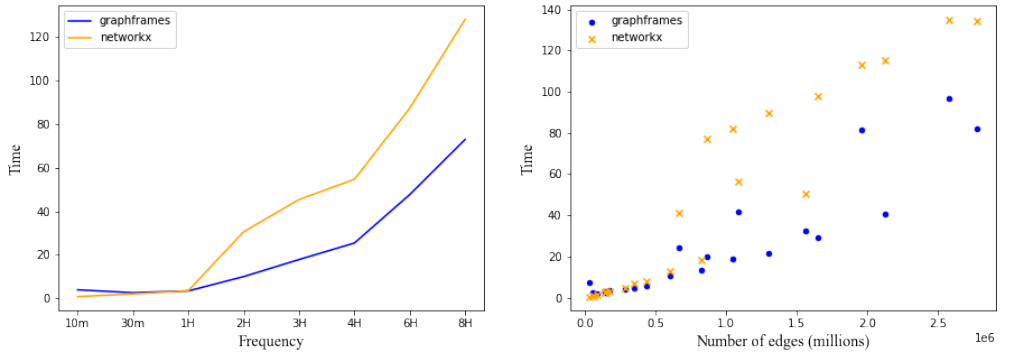

Figure A.4: Time taken to compute the undirected transitivity of the unfiltered BAN using NetworkX and GraphFrames. The plot on the left shows the average time taken for multiple instances of the BAN at different frequencies. The plot on the right shows the same data plotted against the number of edges in the corresponding network, making the horizontal axis linear. This axis is continuous, thus a grouped average could not be applied. Note how the performance of NetworkX is marginally better for smaller networks. GraphFrames' poor performance on these networks suggests that the overhead of parallelising jobs across workers in Spark outweighs the benefits. However, Spark performs significantly better on larger networks.

Appendix B

Additional Results

B.1 Time Series of Network Metrics

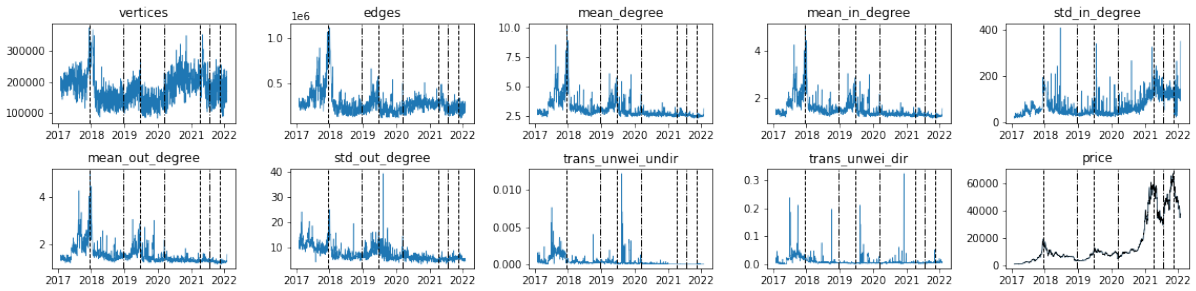

Figure B.1: Exploratory visualisations for the daily BAN.

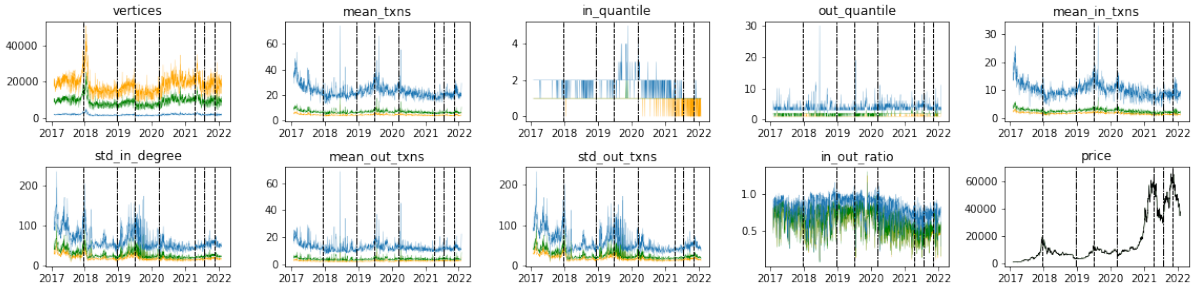

Figure B.2: Hub analysis metrics in the daily BAN, hubs identified by transaction count.

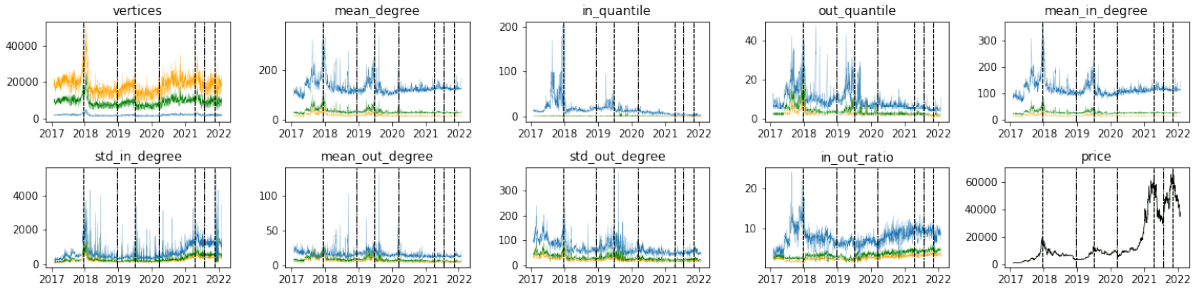

Figure B.3: Hub analysis metrics in the daily BAN, hubs identified by degree.

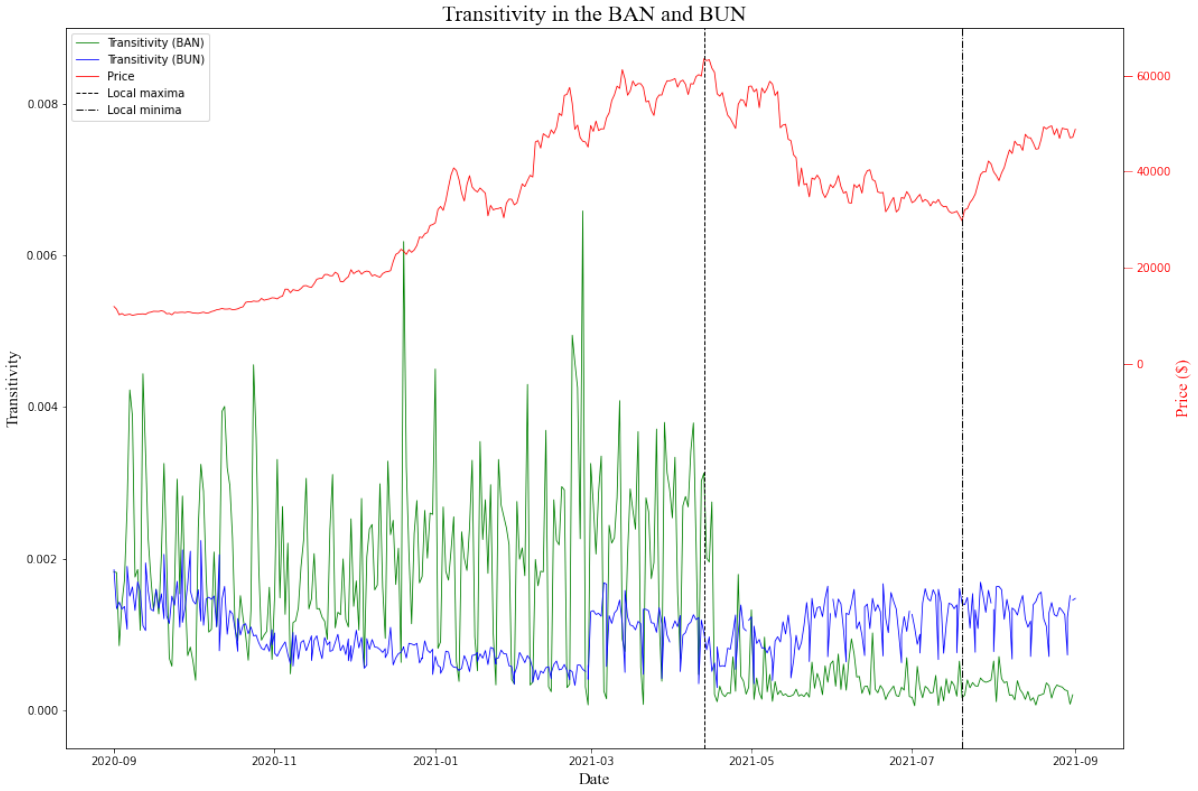

Figure B.4: The transitivity of the daily BAN and BUN, with no filter applied. Notice that while both network representations display transitivity values within a similar range, the transitivity of the BUN doesn't appear to undergo a change in regime, whereas the transitivity of the BAN (Figure 6.4) does. Notice also that the transitivity values increase from March 2021. This is because the mapping from address to users only considered transactions after this date.

B.2 Correlation

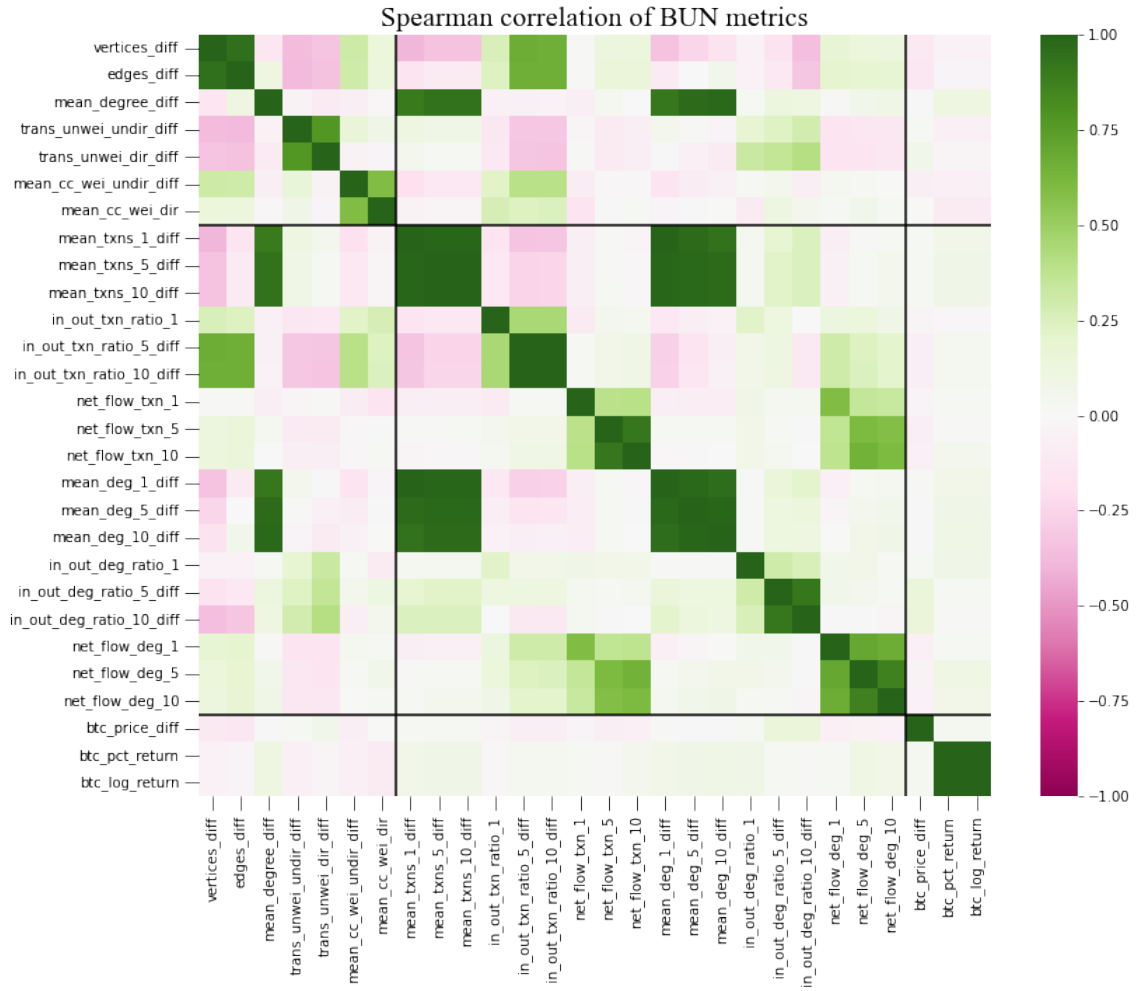

Figure B.5: Spearman correlation coefficients between the metrics of interest, computed on the daily BUN from March 2021 to February 2022 at a threshold of 1BTC. Each axis is split into three by the black lines, which indicates the type of metric. The first region corresponds to aggregate metrics, computed across the entire network. The second region corresponds to percentile metrics, computed across the top $x\%$ of vertices by transaction count or degree. The third region corresponds to exogenous metrics, which in this case are various transformations of Bitcoin's price. Metric names containing *_diff* have been differenced (first order) to make the series stationary. Numbers in metric names (e.g. *_1*) describe the percentage of vertices considered as hubs, ordered by transaction count or degree.

Figure B.6: Spearman correlation coefficients between the metrics of interest for both the BAN and BUN, computed at the daily frequency from March 2021 to February 2022 at a threshold of 1BTC. Each axis is split into three by the black lines, which indicates the type of metric. The first region corresponds to metrics computed on the BAN. The second region corresponds to metrics computed on the BUN. The third region corresponds to exogenous metrics, which in this case are various transformations of Bitcoin's price. Metric names containing *_diff* have been differenced (first order) to make the series stationary. Numbers in metric names (e.g. *_1*) describe the percentage of vertices considered as hubs, ordered by transaction count or degree.

Figure B.7: Spearman correlation coefficients between the metrics of interest for both the BAN and BUN, computed at the weekly frequency from March 2021 to February 2022 at a threshold of 100BTC. Each axis is split into three by the black lines, which indicates the type of metric. The first region corresponds to metrics computed on the BAN. The second region corresponds to metrics computed on the BUN. The third region corresponds to exogenous metrics, which in this case are various transformations of Bitcoin's price. Metric names containing *_diff* have been differenced (first order) to make the series stationary. Numbers in metric names (e.g. *_1*) describe the percentage of vertices considered as hubs, ordered by transaction count or degree.

B.3 Causality

	Bull_1	Bull_2	Bull_3	Bear_1	Bear_2	Bear_3	Full timespan
Number of vertices	**0.0003**	0.0036	0.2454	0.0204	0.1166	0.3739	**0.0**
Number of edges	0.0246	0.3415	0.2003	0.2589	**0.0**	0.252	**0.0253**
Mean degree	0.3343	0.6613	0.8858	0.6913	**0.0331**	0.7853	0.7859
Transitivity (undirected)	**0.0011**	**0.0291**	0.1448	0.5821	0.4493	0.1805	0.0517
Transitivity (directed)	0.0945	**0.0374**	0.8748	0.5078	0.638	**0.0409**	**0.0023**
Avg local clust (undirected)	**0.0078**	0.5696	0.1801	0.5204	0.6621	0.3359	0.5152
Avg local clust (directed)	**0.0073**	0.5938	0.1287	0.5203	0.5676	0.4398	0.5707

Table B.1: Test results for Granger causality of clustering metrics on Bitcoin price, computed on the BUN at the daily frequency with minimum transaction value of 1BTC. The values displayed are p-values; significant values are in bold. For each metric, the test was applied over seven timespans, corresponding to the periods of bullishness and bearishness identified in Fig 4.1 as well as the full period. The Toda & Yamamoto procedue to test for Granger causality was followed.

	Bull_1	Bull_2	Bull_3	Bear_1	Bear_2	Bear_3	Full timespan
In/out transaction ratio (1% transaction count)	0.1532	0.0279	0.5665	0.4925	**0.0027**	0.4422	0.1851
In/out transaction ratio (5% transaction count)	0.1956	**0.007**	0.4971	**0.006**	0.662	0.7897	**0.018**
Net flow (1% transaction count)	0.0	0.4369	0.562	0.2366	0.3509	0.0778	0.1141
Net flow (5% transaction count)	**0.0245**	0.3359	0.6663	0.6551	0.7576	0.0	0.1512
In/out degree ratio (1% degree)	0.1108	0.0	0.6861	0.3641	0.4216	0.3373	0.0426
In/out degree ratio (5% degree)	0.1283	0.0	0.5256	**0.0095**	0.706	0.1015	**0.0089**
Net flow (1% degree)	0.0	0.0	0.5282	0.0	0.3485	**0.023**	0.3003
Net flow (5% degree)	**0.0235**	0.0	0.6099	0.8308	0.7327	0.8979	0.4206

Table B.2: Test results for Granger causality of hub analysis metrics on Bitcoin price, computed on the BUN at the daily frequency with minimum transaction value of 1BTC. The values displayed are p-values; significant values are in bold. For each metric, the test was applied over seven timespans, corresponding to the periods of bullishness and bearishness identified in Fig 4.1 as well as the full period. The Toda & Yamamoto procedue to test for Granger causality was followed.

	Bull_1	Bull_2	Bull_3	Bear_1	Bear_2	Bear_3	Full timespan
Number of vertices	**0.0002**	0.8751	0.0762	0.0020	0.2538	**0.0104**	0.7138
Number of edges	**0.0203**	0.8525	0.1675	**0.0001**	0.1887	0.0465	0.1257
Mean degree	0.7749	0.0681	0.9450	0.9450	0.0000	0.1823	0.0946
Mean in-degree	0.7749	0.0681	0.9450	0.9450	0.0000	0.1823	0.0946
Std in-degree	**0.0108**	0.2622	0.4482	0.4926	0.4933	0.5900	0.2030
Mean out-degree	0.7749	0.0681	0.9450	0.0000	0.0000	0.1823	0.0946
Std out-degree	**0.0049**	0.8246	0.1083	0.1083	**0.0002**	0.5900	**0.0002**
Transitivity (undirected)	0.6157	0.7743	0.6397	0.2792	0.2890	0.0553	0.7263
Transitivity (directed)	0.0650	0.8609	0.2439	0.6708	0.7719	**0.0013**	0.8553
Price	1.0000	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000

Table B.3: Test results for Granger causality of Bitcoin price on network metric (represented by table row), computed on the BAN at the daily frequency with minimum transaction value of 1BTC. The values displayed are p-values; significant values are in bold. For each metric, the test was applied over seven timespans, corresponding to the periods of bullishness and bearishness identified in Fig 4.1 as well as the full period. The Toda & Yamamoto procedue to test for Granger causality was followed.

Appendix C

Code Snippets

C.1 Transitivity

Figure C.1, C.2, and C.3 exhibit code and pseudocode snippets of algorithms for computing network transitivity.

```
def transitivity(G):
    """
    ...
    """
    triangles_contri = [
        (t, d * (d - 1)) for v, d, t, _ in _triangles_and_degree_iter(G)
    ]
    # If the graph is empty
    if len(triangles_contri) == 0:
        return 0
    triangles, contri = map(sum, zip(*triangles_contri))
    return 0 if triangles == 0 else triangles / contri
```

Figure C.1: NetworkX implementation of transitivity.[23]

```
For each node n
    Find neighbours
    For each neighbour n_i of n
        Find neighbours
        Count triangles... How many neighbours does n_i share with n?
        Count triples... How many neighbours does n_i have?
Record the triangle and triple counts
Ensure triangles are not double counted
Return triangles / triples
```

Figure C.2: Pseudocode for computing the transitivity. Notice the nested loop. This leads to slow computation times for large networks.

C.2 SQL Queries

Figure C.4 exhibits the SQL queries for loading the networks.

```
def transitivity(graph):
    """
    Return the global clustering coefficient for a GraphFrames graph.
    Method taken from Chapter 8 of the textbook "Spark GraphX in Action",
    and adapted for Python and GraphFrames.
    """
    triangles = graph \
        .triangleCount() \
        .select(F.sum("count")) \
        .collect()[0][0]
    triples = graph \
        .aggregateMessages(
            F.collect_set(AM.msg).alias("neighbours"),
            sendToSrc=AM.dst["id"],
            sendToDst=AM.src["id"]) \
        .withColumn(
            "neighbours",
            F.size(F.array_except("neighbours", F.array("id"))))
        .withColumn(
            "triplets",
            (F.col("neighbours") * (F.col("neighbours") - F.lit(1))) / F.lit(2))
        .select(F.sum("triplets")) \
        .collect()[0][0]
    return triangles / triples
```

Figure C.3: A custom implementation of network transitivity for Apache Spark using Python and GraphFrames, adapted from the GraphX implementation in.[38] The algorithm was tested over the same set of one-hundred networks as NetworkX and gave the same result for all.

```sql
-- 1) Unweighted BAN
select inputs.tx_id, inputs.pub_key_id as src, outputs.pub_key_id as dst
from inputs
inner join outputs
    on outputs.out_tx_id = inputs.tx_id
    and outputs.time >= 1609718400
    and outputs.time < 1609719600
    and outputs.value >= 100000000

-- 2) Approximated weighted BAN
-- Neither the in- nor out-value represents the true value of the transfer.
-- An approximation must instead be made using the total inputs/outputs.
select inputs.tx_id, inputs.pub_key_id as src, outputs.pub_key_id as dst,
    inputs.value as in_value, outputs.value as out_value
from outputs
inner join (
    select inputs.tx_id, inputs.pub_key_id, outputs.value
    from inputs
    inner join outputs
        on outputs.out_tx_id = inputs.prev_tx_id
        and outputs.out_index = inputs.prev_tx_out_index
) inputs
    on outputs.out_tx_id = inputs.tx_id
    and outputs.time >= 1609718400
    and outputs.time < 1609719600
    and outputs.value >= 100000000

-- 3) Unweighted BUN
select inputs.tx_id, um1.user_id as src, um2.user_id as dst
from inputs
inner join outputs
    on outputs.out_tx_id = inputs.tx_id
    and outputs.time >= 1609718400
    and outputs.time < 1609719600
    and outputs.value >= 100000000
inner join user_mapping um1
    on um1.pub_key_id = inputs.pub_key_id
    and um1.user_id != -1
inner join user_mapping um2
    on um2.pub_key_id = outputs.pub_key_id
    and um2.user_id != -1
group by inputs.tx_id, um1.user_id, um2.user_id

-- 4) Weighted BUN
select inputs.tx_id, um1.user_id as src, um2.user_id as dst,
    sum(outputs.value) as value
from inputs
inner join outputs
    on outputs.out_tx_id = inputs.tx_id
    and outputs.time >= 1609718400
    and outputs.time < 1609719600
    and outputs.value >= 100000000
inner join user_mapping um1
    on um1.pub_key_id = inputs.pub_key_id
    and um1.user_id != -1
inner join user_mapping um2
    on um2.pub_key_id = outputs.pub_key_id
    and um2.user_id != -1
group by inputs.tx_id, um1.user_id, um2.user_id
```

Figure C.4: SQL queries for loading the four different network representations.