MA 710 Data Mining

Project Report

on

Network Analysis of Ethereum Exchanges: Insights and Predictions from Transaction Data

Soubhik Chakraborty, Monika Meshram, Lalitha Voruganti, Shweta Dhakare & Prachi Vadhani

Spring 2023

Abstract — Blockchain analytics and smart contracts are now-a-days very popular mode of financial transactions, and because of it being somewhat relatively unregulated by any single or multiple entity, financial irregularities and bitcoin missuses are becoming more common than thought of. Fortunately, due to publicly available transaction ledger of each of these instruments and its history immutably recorded, large scale analysis of these activities can be modeled and future transactions can be evaluated realtime for potential anomaly detection.

We will use network data analysis and modeling to accommodate multi layered multi party distributed nature of trading across various exchanges and individuals. We will further show some evolved ways to find dyadic and triadic behaviors without looking into attributes of everyone as well as identify relationships using entity attributes such as type, identity etc. Social network like interaction allowing to see the overall transactional activity is recent topic of research and evolving rapidly towards Graph Neural Networks. In our given scope, we will use Exponential Random Graphs Models (ERGMs) to model blockchain transactions that can be used to predict validity of new unseen transactions.

1. INTRODUCTION/MOTIVATION

What is blockchain?

A blockchain is a decentralized ledger or distributed database shared across a computer network. While most associated with the secure and decentralized record-keeping of crypto currency transactions, blockchains are versatile and can be applied to a variety of industries to ensure data immutability, meaning that they cannot be altered.

Since the inception of Bitcoin in 2009, the use cases for blockchain technology have expanded significantly, with the emergence of new cryptocurrencies, decentralized finance (DeFi) applications, non-fungible tokens (NFTs), and smart contracts.

Key Take Aways

- Blockchain is a shared database that stores data in blocks connected through cryptography, differing from traditional databases in its architecture.
- While blockchains can store various types of data, the most prevalent use case is as a ledger for transactions.
- In the context of Bitcoin, the blockchain is decentralized, meaning that control is distributed among users, with no individual or group having sole control.
- Decentralized blockchains are immutable, meaning that once data is entered, it cannot be altered or deleted. In the case of Bitcoin, transactions are permanently recorded and publicly visible.

What is AML?

Anti-money laundering (AML) refers to a network of regulations, laws, and procedures designed to uncover attempts to disguise unlawful funds as legitimate income. Money laundering is the practice of hiding various types of crimes, such as tax evasion, drug trafficking, public corruption, and financing for designated terrorist organizations, among others.

AML legislation emerged as a response to the growth of the financial industry, the removal of international capital controls, and the ease of conducting complex financial transactions.

According to a high-level panel of the United Nations, annual money laundering flows were estimated at $1.6 trillion in 2020, which accounts for 2.7% of the global GDP.

Key Take Aways

- The main goal of Anti Money Laundering (AML) efforts is to prevent criminals from concealing the profits of their unlawful activities.
- Criminals use money laundering to make their illicit funds appear as though they were obtained legally.
- AML regulations mandate that financial institutions establish complex customer due diligence plans that assess money laundering risks and detect suspicious transactions.

What is fraud detection?

Detecting fraud can be challenging, given the unlimited and increasing number of fraudulent methods that exist. An organization's ability to detect fraud can be compromised by various factors, such as reorganization, downsizing, migration to new information systems, or cybersecurity breaches. Therefore, it is advisable to use real-time monitoring techniques to detect fraudulent activities. These techniques should encompass the scrutiny of financial transactions, locations, devices used, initiated sessions, and authentication systems.

Fraud detection techniques

- Classify groups, and segment data to search through millions of transactions to find patterns and detect fraud
- Learn suspicious-looking patterns and use those patterns to detect them further through neural networks
- Identify characteristics found in fraud through Pattern recognition

2. THE DATA SET

The network data on the blockchain was extracted from "http://snap.stanford.edu/data/ethereum-exchanges.html". The total dataset is of 8.3 gigabytes with 38.9 million transactions being recorded across 28 tokens. There are 6.08 million distinct to and from addresses out of which 297 entities could be web scrapped from etherscan.io and identified. Each token have very long ledger entries (long chain) for example token 0x6f259637dcd74c767781e37bc6133cd6a68aa161 has 344,426 number of ledger entries recorded so far.

The data contains token_address, from_address, to_address, transaction_hash, value, blocknumber and logindex as main attributes.

Exchange details have address, name and exchange type (cex/dex) i.e. centralized exchange or decentralized exchange.

3. THEORY AND METHODS

Fraud detection Approach -

Network analysis can be a powerful tool for detecting fraud and money laundering. It involves analyzing transactional data in a network graph format to identify suspicious patterns and connections. However, visualizing the entire network graph in one shot can be impossible due to the sheer size of the data. Instead, analysts often zoom in on specific hubs or clusters and examine their connections and transactional behavior in detail. By identifying abnormal activity, such as a high volume of transactions with unverified parties or frequent transfers to offshore accounts, analysts can flag potential cases of fraud and investigate further. Additionally, network analysis can be used to track the flow of funds and uncover hidden relationships between seemingly unrelated entities, providing valuable insights for anti-money laundering efforts.

Network Five Point Summary

- Size

- Density
- Clustering coefficient
- Degree distribution
- Diameter

Modularity: Modularity is a measure of the strength of the division of a network into modules or communities.
Centrality: Centrality is a measure of the importance of a node in a network.
Closeness: Closeness is a measure of how easily a node can reach all other nodes in the network.
Jacquard & Vertex Proximity: Jaccard index measures the similarity between two sets of nodes in a network, while vertex proximity measures the probability of two nodes being connected in a network. These measures can be useful in identifying suspicious transactions between nodes that are not directly connected but have high similarity or probability of being connected.

Sampling

→ Approach A involves the challenge of processing a large amount of data in a blockchain model Ethereum dataset. The dataset contains 38 million data nodes, which cannot be processed on a laptop. Therefore, the first approach is to trim the data and pick the first 5 million data entries.

→ Two choices were considered to analyze the trimmed data's network structure. The first option was to pick the largest component and follow the "highest activity," while the second option was to take a random sample (10%) of 5 million nodes and model the overall network.

→ Approach B aimed to provide proof of the effectiveness of sampling (approach B) by showing that the network density and correlation were retained. The sampling results revealed that the network density and correlation were nearly retained, indicating adequate sampling.

→ Despite the sampling, the ergm model did not converge due to the size and sparsity of the network. Therefore, the final choice was to take approach A and determine the largest component out of the 5 million node network. However, this approach was still computationally demanding, and further trimming was required.

→ Approach A.1 involves picking one token_address and focusing on its chain, while Approach A.2 involves simply picking the top X no of rows, just like the first 5 million rows were selected. These approaches are intended to reduce the computational load and enable the analysis of the network structure of the Ethereum dataset.

Modeling

Large Component:

- The largest component in the Ethereum dataset refers to the subgraph of the entire network that contains the largest number of nodes that are connected to each other. In other words, it is the largest subset of the Ethereum network where each node is connected to at least one other node in the same subset.
- This component is important in network analysis as it often contains the most significant information about the structure and behavior of the network.

ERGM :

- Exponential random graph model (ERGM) is a statistical model used to analyze network data. It models the probability of a network's structure based on the local configurations of the nodes and edges. ERGM can capture various network properties such as degree distribution, clustering, and homophily.
- In the context of Ethereum network analysis, ERGM can be used to model the network structure and identify the significant structural features that contribute to the observed patterns of transactions. For example, it can help identify which nodes are more likely to be involved in high-value transactions or which edges are more likely to be used for money laundering activities.
- However, ERGM can be computationally intensive and may not converge for large and sparse networks. Therefore, researchers often need to use subnetworks or sampling techniques to reduce the computational burden. Additionally, ERGM assumes that the observed network is a random sample from a population of all possible networks with the same node and edge set, which may not be realistic for some network datasets.

4. DATA ANALYSES

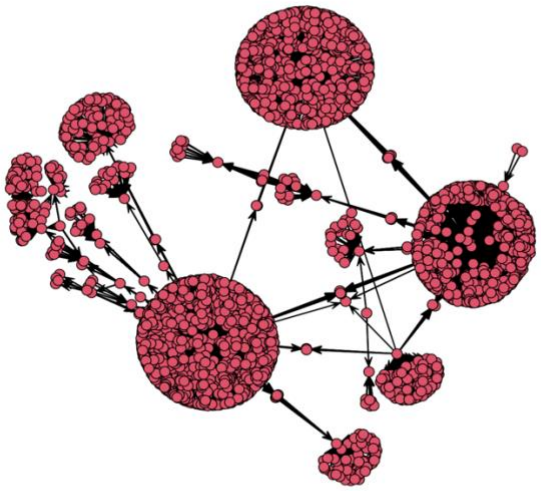
The dataset consists of weighted, directed graphs with partially available node labels, representing token networks that have been extracted from the Ethereum blockchain. Network analysis techniques, such as community detection and centrality analysis, can be used to identify clusters of nodes that are tightly interconnected, and to measure the importance of nodes based on their connectivity patterns.

Sampling the trimmed data:

- The original dataset contains 38 million data nodes, which is too large to process in a laptop, so the decision was made to trim the data to the first 5 million data entries.
- However, plotting a graph of the 5 million node network is still impossible, so there are two choices: A) pick the largest component and follow the "highest activity," or B) randomly sample 10% of the 5 million data entries and try modeling the overall network.
- To test the effectiveness of approach B, a hundred samplings were done, and it was found that the network density was nearly retained, as well as the CLT tendencies of the network size and average "maximum degree" of the sampled network compared to the original network size.
- It was also found that the samples retain the correlation and assortativity of exchange types and exchange name association.
- Despite the effectiveness of the sampling, the ergm model did not converge when either removing edges or removing vertices from the sampled network due to too much computational intensity or flattened series during iteration leading to infinity.
- The second choice was to pick the largest component from within the sample, but the ergm model did not converge due to the small sample size or sparse network.
- The final choice of modeling was to take approach A and determine the largest component out of the 5 million node network.
- However, the computation to determine the "degree" network attribute of the 5 million node network still takes more than half an hour, so further trimming was required to retain the network shape.
- Two approaches were considered for further trimming: A.1) pick one token_address and focus on its chain or A.2) simply pick the top X number of rows, like the first 5 million rows were picked.
- Approach A.1 was deemed too specialized for the current scope of the analysis, so the decision was made to pick the first 50,000 nodes.
- Assuming that the primary dataset's largest component has a network size of 2799, density of 0.0046482, and maximum degree of 15351, the rest of the network statistics will be based on these 50,000 trimmed datasets.

Large Component

- **Due to long chain, pick the largest.component of the trimmed down dataset of 5 million.**



When working with large datasets, it is often necessary to trim down the data in order to perform analysis on a more manageable subset. In the case of the Ethereum dataset provided by Stanford, it was not possible to process all 38 million data nodes on a laptop, so the first 5 million data entries were selected for analysis. However, even this subset was too large to plot as a graph, so two choices were considered: picking the largest component and following the "highest activity," or taking a random sample (10%) of the 5 million data entries and modeling the overall network.
After performing various analyses, it was determined that the final choice of modeling would be to pick the largest component out of the 5 million node network. This allowed for a more manageable dataset to be analyzed while still maintaining an accurate representation of the overall network.

The largest component of the trimmed down dataset had a network size of 2,799, a density of 0.0046482, and a maximum degree of 15,351. These statistics provide valuable insight into the structure and behavior of the Ethereum network.

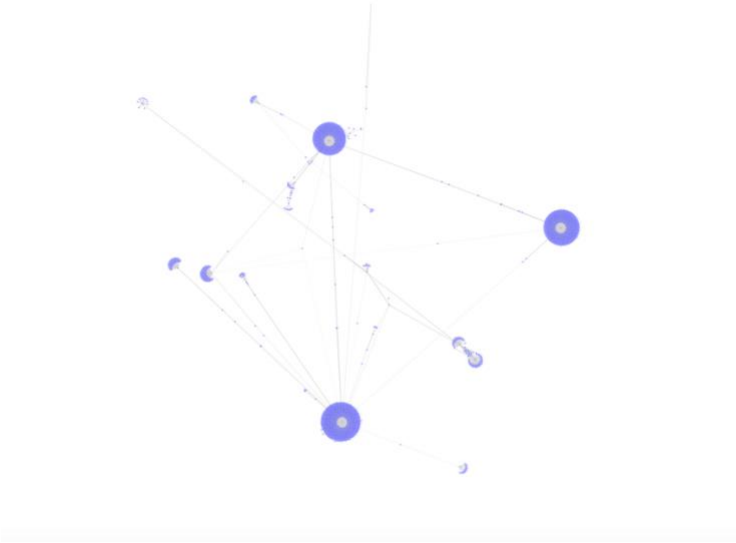
By focusing on the largest component of the trimmed down dataset, it is possible to gain a deeper understanding of the network without being overwhelmed by the sheer size of the original dataset. This approach allows for more targeted analysis and can lead to more accurate insights and conclusions.

- **Inspect overall "highest activity" (largest component) chain of transaction.**

The "highest activity" chain of transactions in the Ethereum dataset can reveal valuable insights about the network's overall behavior. By analyzing the largest connected component of the network, we can identify the most active entities and transactions and understand the flow of tokens across the network.

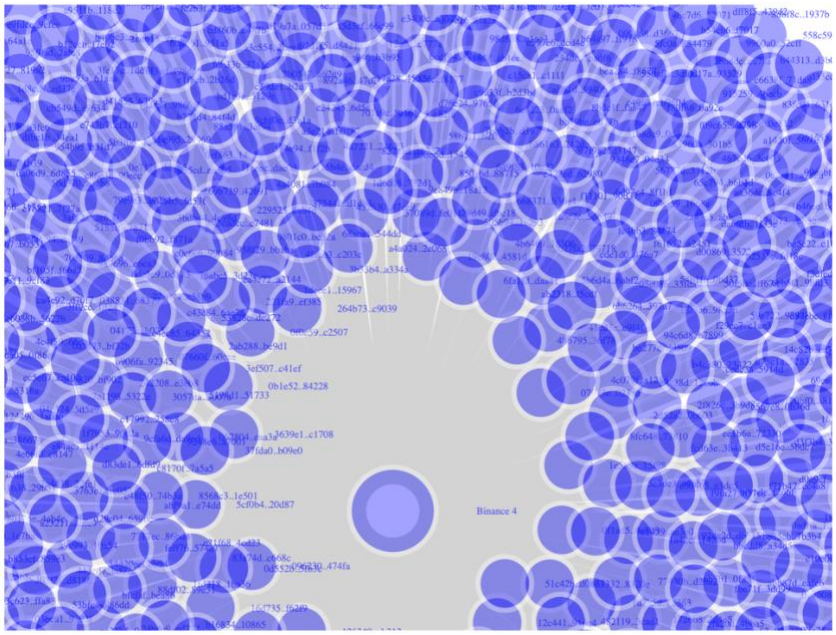
For instance, the "highest activity" chain may reveal which tokens are being frequently exchanged, the most active exchanges or wallets, and the most common transaction types (e.g., transfers, trades, etc.). This information can help us identify the key players in the network and their behavior patterns.

Additionally, studying the "highest activity" chain can help us identify potential anomalies or frauds in the network. For example, if we notice a sudden surge in the transaction volume of a particular wallet or exchange, we can investigate further to determine if the activity is legitimate or suspicious. Inspecting the "highest activity" chain of transactions is a crucial step in network analysis, as it can provide valuable insights into the behavior and dynamics of the network as a whole.



The above graph shows the clusters connected with each other, probably the exchanges but it does not give away any relevant diagnosis hence, zooming in into this network is needed. Zooming in can often reveal more details about the network structure and relationships between nodes. It can help us identify specific clusters or subgraphs within the larger network and provide more insights into the dynamics of the network.
In the case of the Ethereum dataset, zooming in on a specific hub or cluster can reveal patterns of transactions and connections that are not immediately apparent from the larger network graph. This can be useful for identifying key players or entities in the network, understanding the flow of transactions between different nodes, and detecting potential fraud or malicious activity.
By analyzing the largest component of the trimmed down dataset of 5 million transactions, we can get a better understanding of the overall structure of the network and identify any interesting clusters or subgraphs that may warrant further investigation. Hence, we proceed to get a clear and enhanced network graph of one of these hubs/clusters.

- **Zooming in shows one of the hotspots is Binance centralized exchange (cex)**



Binance is one of the largest and most well-known centralized cryptocurrency exchanges in the world. As a centralized exchange, Binance acts as a hub for buying and selling cryptocurrencies, allowing users to trade a wide range of digital assets with other users on the platform.

In network analysis, Binance is an important node or entity to consider because of its significant role in the Ethereum ecosystem. Binance is involved in a large volume of transactions on the Ethereum blockchain, making it a crucial player in the overall network structure. Its high degree centrality, or the number of direct connections it has to other nodes in the network, makes it an important point of analysis for understanding the overall network topology and flow of transactions.
By analyzing Binance's interactions with other nodes in the network, researchers and analysts can gain insights into various aspects of the Ethereum ecosystem, including market trends, liquidity, and even potential fraud and money laundering activities. The centrality of Binance in the Ethereum network also means that any changes or disruptions to its operations can have a significant impact on the overall network structure and transaction flows.

Therefore, understanding the role of Binance and other centralized exchanges in the Ethereum network is essential for conducting thorough and accurate network analysis, and for gaining insights into the overall functioning of the cryptocurrency ecosystem.

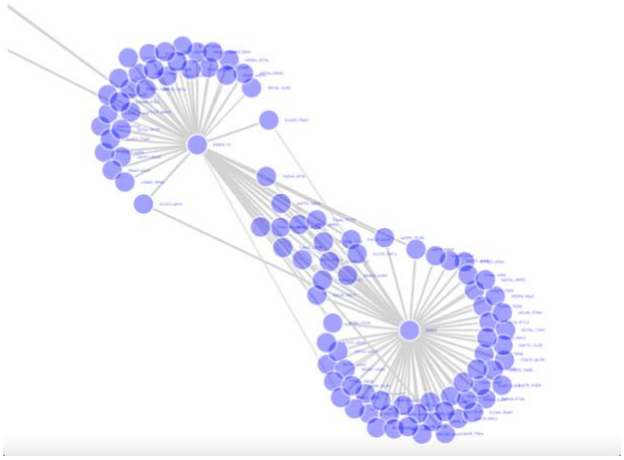
Manual Tracking of Money Laundering

A suspect transaction: Decentralized exchange closed loop transactions:
Decentralized exchanges (DEXs) are a type of cryptocurrency exchange that operates on a decentralized blockchain network, such as Ethereum. Unlike centralized exchanges, which are operated by a single entity, DEXs allow users to trade cryptocurrencies directly with each other without the need for

intermediaries. Closed loop transactions in Ethereum-based DEXs refer to transactions where the assets being exchanged are both Ethereum-based tokens. In other words, the tokens being traded are both on the Ethereum blockchain, and no other cryptocurrencies or fiat currencies are involved in the transaction.

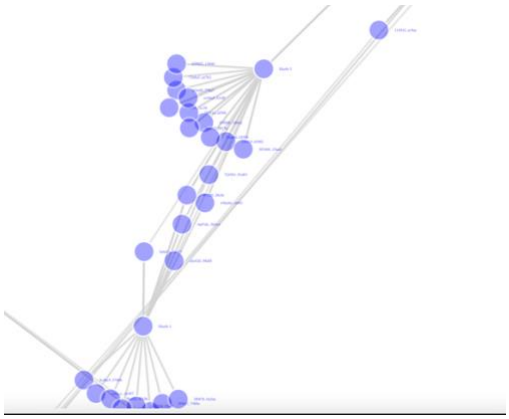
In a closed loop transaction on a DEX, the process typically works as follows:
-A user who wants to trade one Ethereum-based token for another will first need to connect to a DEX using a web3-enabled wallet, such as MetaMask.
-Once connected, the user can search for the specific token they want to trade and enter the amount they wish to exchange.
-The DEX will then search for a matching trade order from another user who wants to trade the opposite token for the one the first user has.
-Once a match is found, the DEX will execute the trade by transferring the tokens between the two users' wallets on the Ethereum blockchain.
-The trade is considered a closed exchanged directly on the Ethereum blockchain without needing loop since both tokens are Ethereum-based and are exchanged directly on the Ethereum blockchain without the need for any other currency or intermediary.
-Closed loop transactions in Ethereum-based DEXs offer several advantages over centralized exchanges, including greater privacy and security, as well as lower fees and faster transaction times. However, they may also have some limitations, such as lower liquidity and potentially higher volatility due to the absence of a central order book.

In Network terminology, network loops (dyadic relation) and homophily of the nodes.



Good Transactions: There is a significant amount of local activity within the fifth largest component of the system, but there is less exchange of money with lower assortativity. This means that there are many transactions happening within the local community or network, but there is limited interaction between different groups or communities within the larger system. In other words, the flow of money or

transactions is more concentrated within certain subgroups or clusters of the system, rather than being distributed evenly throughout the entire system. This can be seen as a form of social or economic segregation, where certain groups are more connected and integrated with each other, while others remain isolated or disconnected. Overall, while there is some degree of local activity and interaction within the system, there are also significant barriers to cross-group exchange or communication, which may have implications for the overall health and resilience of the system.



- **SAMPLING**

Approach A:

Approach A involves the challenge of processing a large amount of data in a blockchain model Ethereum dataset. The dataset contains 38 million data nodes, which cannot be processed on a laptop. Therefore, the first approach is to trim the data and pick the first 5 million data entries.

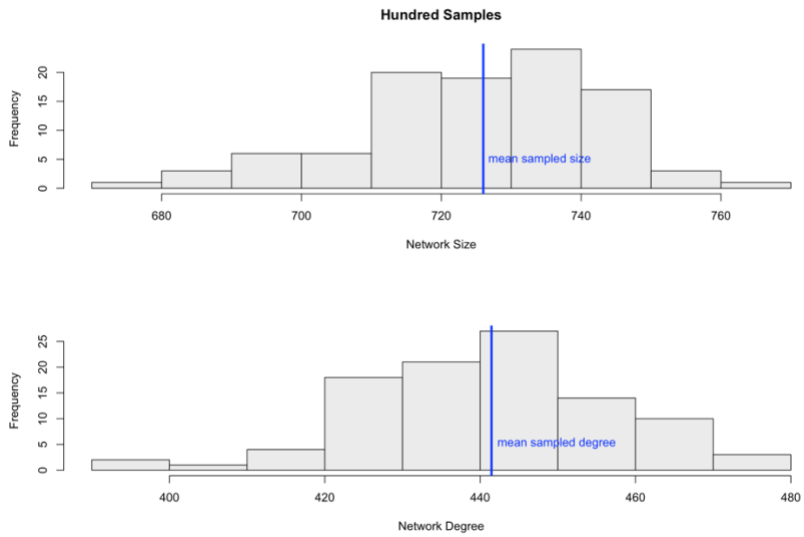
- **Sampling the trimmed data.**
- **38 million data nodes cannot be processed in a laptop (not even open and random sample is possible).**
- **So, picked the first 5 million data entries (trimming).**
- **Plotting 5m node graph is impossible, two choices**

A) Pick the largest component and follow the "highest activity".
B) Random sample (10%) of 5 million, and try modeling the overall network.

Approach B:
Proof of sampling (approach B): Hundred sampling of the 5 million transactions shows we nearly retain the network density (or the most we could attain)

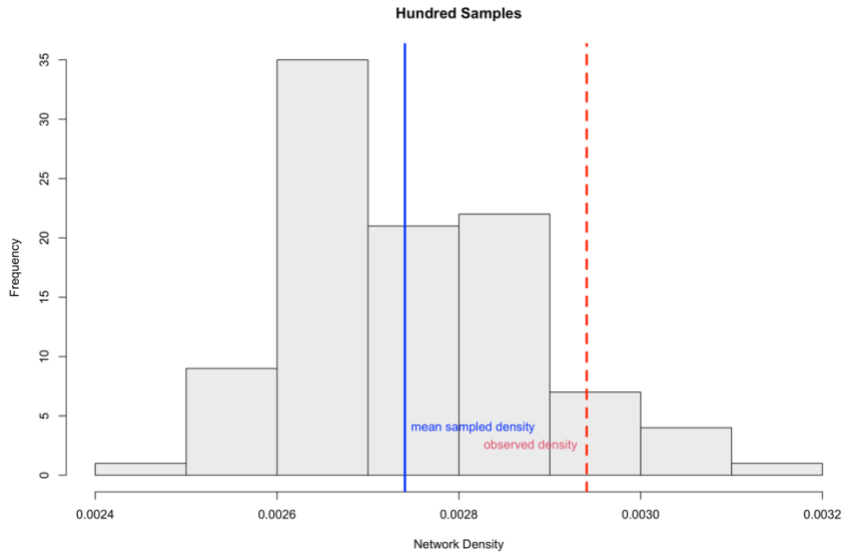
Before we move on to approach A as a final choice, why approach B is random sampling of 5million rows or 120k rows, we could open. Why a random sampling was not a good idea, do we want to think that bootstrapping or bagged model kind of thing, would have been better choice but for that to happen entirely we did more dataset and more complete core. Given this limited dataset, what we could see is difference between 50000 to 120, 000 / 500 00 to 50m and there were lot of distortion in the fact of sheet. So, 10% random sampling, the size frequency is showing by central limit theorem it is having a tendency of around 720 degree of network size and network degree is 440, if you look at the network node given is the exact original network size, and original network degrees which is way off than this., it is definitely not the representation of the same network. (way off w.r.t. to the population)

we nearly retain the network size distribution and degree characteristics (or at least what's the most we could attain)



The CLT tendencies of the network size and average "maximum degree" of the sampled network are retained around 740 and 400 compared to original network size of 14,405 and maximum degree 0.00024098.

Density : Similarly, density is symbolic, So the sample density is also way down and lesser dense and lesser dense means, fewer connections-fewer edge and state it is very difficult to detect a pattern.
The red line is density from 120k transactions representative towards sample density being way off from the population.



Correlation with other networks around:

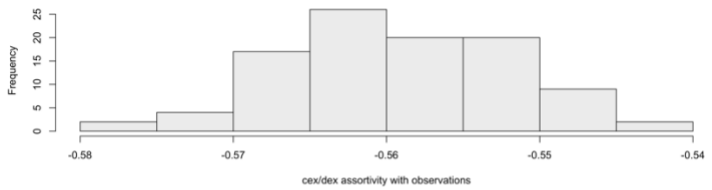
The correlation with other networks the correlation matrix that is showing across 100 samples if you could do a correlation plot of the network they are showing a central tenancies, what is the likelihood of making edges and that also is showing tenancy where it is 0.56 which is around 54% of likelihood bith for type of exchanges like centralized exchanges and decentralized exchanges, hence, we can say that Samples retain the correlation.

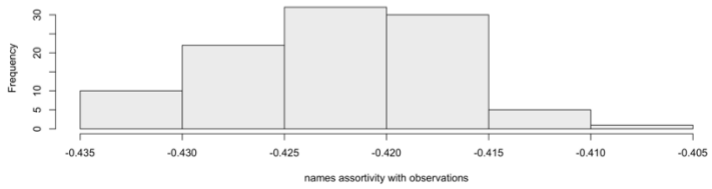
Assortativity :

Assortativity refers to the tendency of nodes in a network to be connected to nodes with similar attributes.In the case of the Ethereum exchanges network, the assortativity of exchange types and exchange names means that exchanges of the same type and name tend to be connected to each other.

Assortativity of exchange types and exchange name association is retained. Say by if we just look at the exchange names they are showing 50% chances of being here or not there which is not a very good probability in terms of what we have seen so far, that why we finally decided to take the approach to one out of Approach A where we have taken the largest component then we can take that OK that is the follow the money kind of better approach wherever there is bog movement of money or big volatility happening it is a good idea to look at volume transaction.

So far, we got an idea that the sampling does not retain the original network chrematistics.



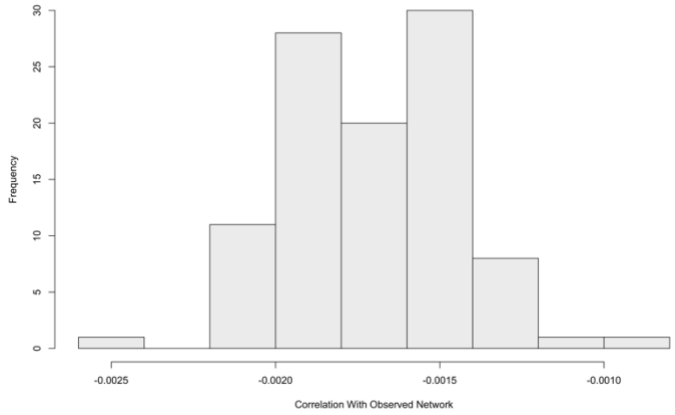


Assortativity :

Assortativity refers to the tendency of nodes in a network to be connected to nodes with similar attributes.In the case of the Ethereum exchanges network, the assortativity of exchange types and exchange names means that exchanges of the same type and name tend to be connected to each other.

Assortativity of exchange types and exchange name association is retained. Say by if we just look at the exchange names they are showing 50% chances of being here or not there which is not a very good probability in terms of what we have seen so far, that why we finally decided to take the approach to

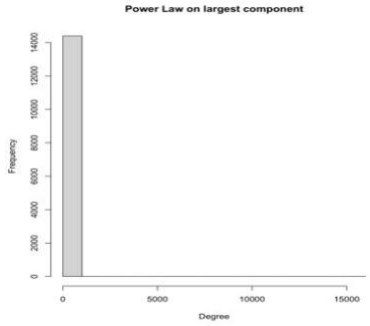
one out of Approach A where we have taken the largest component then we can take that OK that is the follow the money kind of better approach wherever there is bog movement of money or big volatility happening it is a good idea to look at volume transaction.



So far, we got an idea that the sampling does not retain the original network chrematistics.

- **EDA before modeling:**

Power Law:

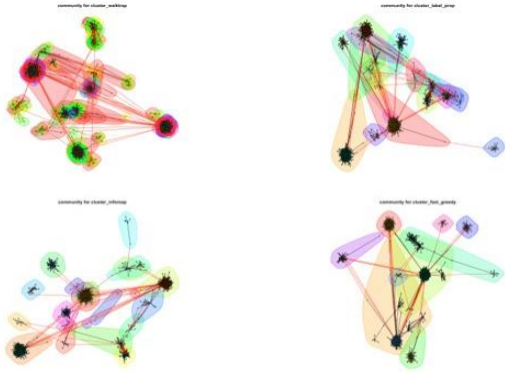


	continuous	alpha	xmin	logLik	KS.stat	KS.p
1	0	2.681	1	-7518.992	0.018	0.007

From the above plots, Alpha (α) is the exponent that describes the slope of the Power Law curve. In other words, it quantifies how quickly the frequency of connections decreases as the number of connections increases.
A smaller value of α means that the decrease is faster, indicating that fewer nodes have a large number of connections.
Minx, on the other hand, is the minimum number of connections required for a node to be included in the Power Law distribution. In some cases, there may be nodes with very few connections that do not follow the Power Law pattern. By setting a minimum threshold for the number of connections, we can

exclude these nodes and focus only on those that exhibit the Power Law distribution. This can help to improve the accuracy and reliability of our analysis.
Power law suggests everyone is at equality, which is a good thing otherwise a polarised interaction means some are doing "out of ordinary" activity.

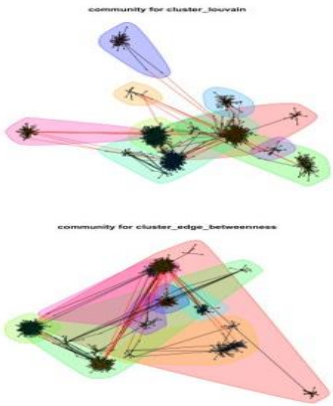
Modularity:



	modularity
cluster_walktrap	0.3082354
cluster_label_prop	0.3084468
cluster_infomap	0.3084454
cluster_fast_greedy	0.3085777
cluster_louvain	0.3085777
cluster_edge_betweenness	0.3085764

Modularity is a measure of how well a network is divided into modules or communities based on the connectivity patterns of the nodes. It ranges from -1 to 1, with higher values indicating a stronger division of the network into modules. A modularity value of 0.3 indicates that the network has a moderate level of community structure, with nodes being more connected within their communities than between them. It suggests that the network can be divided into meaningful groups based on the patterns of exchange among the nodes. However, the strength of the community structure is not as strong as in networks with higher modularity values.

Centrality:



Rand Index comparing various centralities

	cluster_walktrap	cluster_label_prop	cluster_infomap	cluster_fast_greedy	cluster_louvain	cluster_edge_betweenness
cluster_walktrap						
cluster_label_prop	0.09917					
cluster_infomap	0.10003	0.99466				
cluster_fast_greedy	0.09018	0.94736	0.94439			
cluster_louvain	0.09018	0.94736	0.94439	1		
cluster_edge_betweenness	0.08981	0.94338	0.94464	0.98543	0.98543	

Centrality is a measure of the importance of a node in a network. There are different types of centrality measures, including degree centrality, betweenness centrality, closeness centrality, and eigenvector centrality.
Betweenness centrality is a measure of a node's importance in a network based on the number of shortest paths that pass through that node. It is calculated by identifying all of the shortest paths between pairs of nodes in the network and then calculating the fraction of those paths that pass through each individual

node. Nodes with high betweenness centrality are important for maintaining communication and information flow between different parts of the network, as they act as key bridges or intermediaries between different groups of nodes. In other words, nodes with high betweenness centrality are strategically positioned in the network and can have a significant impact on the flow of information and resources through the network.
The Rand Index is a measure of the similarity between two clusterings, with a value of 1 indicating perfect agreement between two clustering methods. In this case, we're comparing different centrality measures in the Ethereum dataset using the Rand Index.
The results of the Rand Index comparison show that there is generally good agreement between the centrality measures used in the analysis, except for the walktrap method.

- **MODELING**

Largest component is extracted and isolated from rest of the network for further modeling of various exchanges' traffic flow across multiple instruments (tokens).

1. What is inside largest component?

Introduction to model:

Top 8 Exchanges

	Exchange	Num Txns
1		2776
2	Huobi	4
3	Binance	2
4	Bitmax	2
5	DDEX	2
6	Gate	2
7	ABCC	1
8	Bittrex	1

Exchange Types

	Exchange Type	Num Txns
1		2776
2	cex	17
3	dex	6

Component summary

	size	density	diameter	cl.coeff	txns
1	2799	0.0046482	9.5147495923e+25	0.0080961	36403

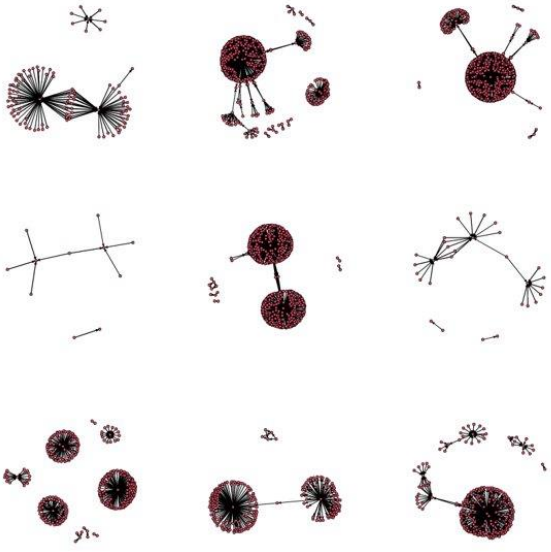
Top 20 Tokens

	Tokens	Num Txns
1	0xc02aaa39b223fe8d0a0e5c4f27ead9083c756cc2	19991
2	0x0d8775f648430679a709e98d2b0cb6250d2887ef	3679
3	0x0000000000085d4780b73119b644ae5ecd22b376	3382
4	0x9f8f72aa9304c8b593d555f12ef6589cc3a579a2	1680
5	0xb8c77482e45f1f44de1745f52c74426c631bdd52	1614
6	0x89d24a6b4ccb1b6faa2625fe562bdd9a23260359	1368
7	0xa15c7ebe1f07caf6bff097d8a589fb8ac49ae5b3	989
8	0xa0b86991c6218b36c1d19d4a2e9eb0ce3606eb48	727
9	0x6f259637dcd74c767781e37bc6133cd6a68aa161	635
10	0xe41d2489571d322189246dafa5ebde1f4699f498	446
11	0xd26114cd6ee289accf82350c8d8487fedb8a0c07	363
12	0x8971f9fd7196e5cee2c1032b50f656855af7dd26	356
13	0xf629cbd94d3791c9250152bd8dfbdf380e2a3b9c	251
14	0x8e870d67f660d95d5be530380d0ec0bd388289e1	237
15	0xdd974d5c2e2928dea5f71b9825b8b646686bd200	220
16	0x0f5d2fb29fb7d3cfee444a200298f468908cc942	200
17	0xb64ef51c888972c908cfacf59b47c1afbc0ab8ac	80
18	0x1f573d6fb3f13d689ff844b4ce37794d79a7ff1c	78
19	0x514910771af9ca656af840dff83e8264ecf986ca	66
20	0x4dc3643dbc642b72c158e7f3d2ff232df61cb6ce	41

As we can see the largest component covers around 20 distinct tokens out of 28, captures 36,400 transactions out of 50,000 across 2800 entities. Exchange type wise centralized exchanges are 17 and decentralized are 6 in this sub-network.

Isolation The Top 9 Transaction:

Filtering out all the edges (token_address is an edge attribute) equal to the most heavily traded top 9 tokens and plotting a network diagram we see that top 3 tokens show some splitted groups of transactions.



Fourth and Sixth token from the above list of 20, we see strange activity. Zooming further into the fourth network for token 0x9f8f72…579a2 having 1680 transactions converging to two nodes looks something worth looking into.

Expanding the fourth plot below, we DDEX is involved, and each edge is a heavy flow of funds converging towards Paradex or DDEX (it is a directed graph Figure A shows that and Figure B shows entity names where as Figure C shows number of transactions happening around the edge DDEX and bb2c35)

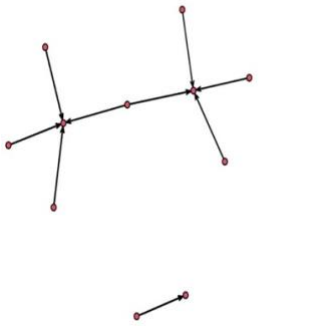


Figure A

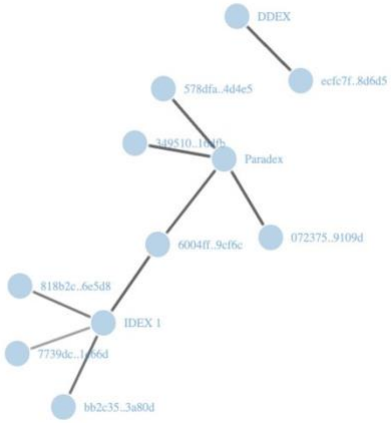


Figure B

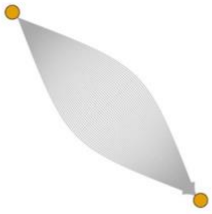


Figure C

Future Friends:

Common Neighbors

	from	to	value
1	DDEX	DDEX 1.0	22
2	Gate.io 3	Gate.io 1	8
3	Binance 4	Bitmax 1	7

Preferrential Attachments

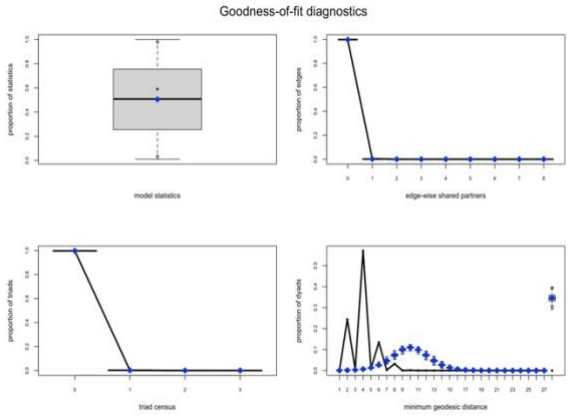
	from	to	value
1	Binance 1	Binance 4	715546
2	Binance 4	Bitmax 1	610207
3	Binance 1	Bitmax 1	484822

- Likelihood of becoming friends i.e., these exchanges as we saw previously showed affinity for newer transactions,
- DDEX was our suspicious transaction because of its isolation from others, whereas Binance being large exchange, expected to have relatively more transactions and shows preferential trading.
- However, due to latest investigation of April'2023, Binance might as well be elevated to a suspicious epicenter when analyzed with all 38Mill transactions.

Base Model:

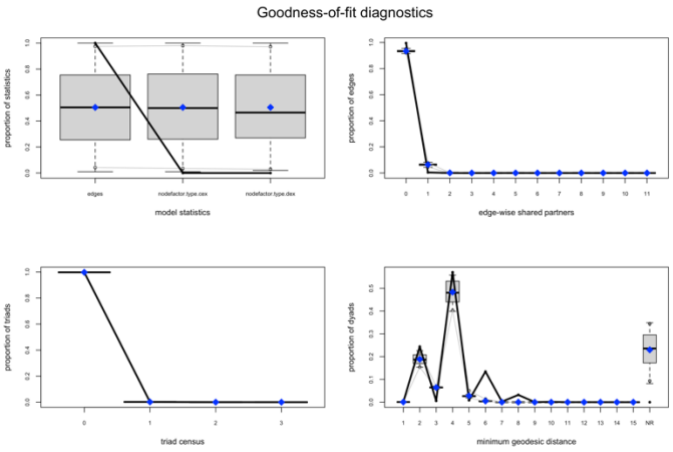
Base Model 1:

```
Call:
ergm(formula = train.nw.inw ~ edges)

Maximum Likelihood Results:

      Estimate Std. Error MCMC % z value Pr(>|z|)
edges -7.21808    0.01868      0  -386.5   <1e-04 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     Null Deviance: 5428453  on 3915801  degrees of freedom
 Residual Deviance:   47157  on 3915800  degrees of freedom

AIC: 47159  BIC: 47173  (Smaller is better. MC Std. Err. = 0)
------------------------------------------------------------
probability
      edges
0.0007326726
```



- **First Model:** Edge probability is equal to density of the network.

```
Call:
ergm(formula = train.nw.inw ~ edges + nodefactor("type"))

Maximum Likelihood Results:

                    Estimate Std. Error MCMC % z value Pr(>|z|)
edges               -9.80102    0.06592      0 -148.67   <1e-04 ***
nodefactor.type.cex  6.86236    0.06868      0   99.92   <1e-04 ***
nodefactor.type.dex  4.26954    0.11552      0   36.96   <1e-04 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     Null Deviance: 5428453  on 3915801  degrees of freedom
 Residual Deviance:   24427  on 3915798  degrees of freedom

AIC: 24433  BIC: 24472  (Smaller is better. MC Std. Err. = 0)
------------------------------------------------------------
probability                                  99 and 98 % probability
              edges nodefactor.type.cex nodefactor.type.dex
       5.539217e-05        9.989547e-01        9.862048e-01
```



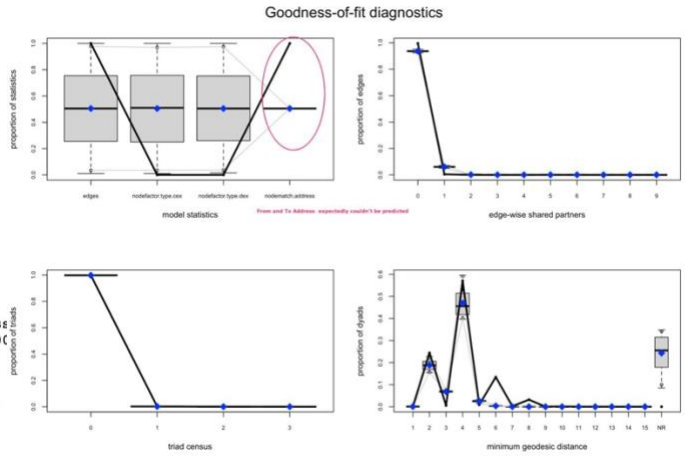
- **Second Model:** Exchange type plays a significant role in classifying transactions however the model prediction is way off from the observations

- **Base Model 2:**

```
Maximum Likelihood Results:

                  Estimate Std. Error MCMC % z value Pr(>|z|)
edges             -9.80102    0.06592      0 -148.67   <1e-04 ***
nodefactor.type.cex  6.86236    0.06868      0   99.92   <1e-04 ***
nodefactor.type.dex  4.26954    0.11552      0   36.96   <1e-04 ***
nodematch.address      -Inf    0.00000      0    -Inf   <1e-04 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     Null Deviance: 5428453  on 3915801  degrees of freedom
 Residual Deviance:   24427  on 3915797  degrees of freedom

AIC: 24433  BIC: 24472  (Smaller is better. MC Std. Err. = 0)

 Warning: The following terms have infinite coefficient estimates:
  nodematch.address
------------------------------------------------------------
probability
            edges nodefactor.type.cex nodefactor.type.dex   nodematch.address
      5.539217e-05        9.989547e-01        9.862048e-01        0.000000e+0(


Call:
ergm(formula = train.nw.inw ~ edges + nodefactor("type") + nodematch("name2"))

Maximum Likelihood Results:
```

Probability changed very quickly and flipped in presence of name

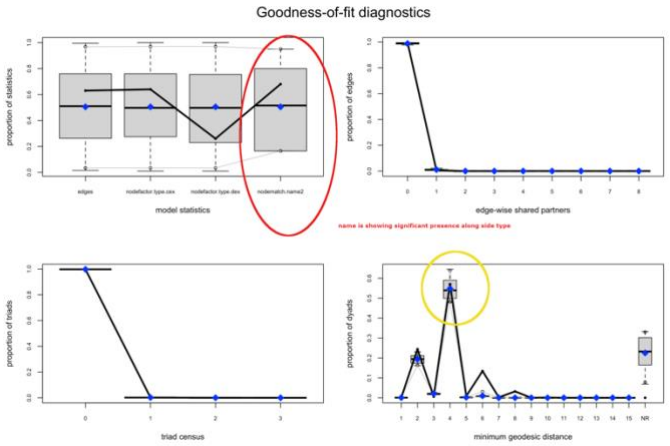
```
                  Estimate Std. Error MCMC % z value Pr(>|z|)
edges             -2.2761     0.4561      0  -4.991   <1e-04 ***
nodefactor.type.cex  -0.5218     0.4549      0  -1.147    0.251
nodefactor.type.dex  -2.4086     0.4615      0  -5.219   <1e-04 ***
nodematch.name2     -12.8879     1.0990      0 -11.727   <1e-04 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     Null Deviance: 5428453  on 3915801  degrees of freedom
 Residual Deviance:   22563  on 3915797  degrees of freedom

AIC: 22571  BIC: 22624  (Smaller is better. MC Std. Err. = 0)
------------------------------------------------------------
probability
            edges nodefactor.type.cex nodefactor.type.dex     nodematch.name2
      9.312236e-02        3.724277e-01        8.251967e-02        2.528372e-06
```



- **Third Model**: Address showing significance indicator, but coefficient is infinity, thus its insignificant. This is a good indicator that self-trading is not a frequent occurrence.

- **Fourth Model:** Name is showing statistically significant and goodness of fit showing close estimation to the observation. Geodesic distance is showing degree 4 is highly proximate. The exchange type probability flipped and changed very quickly showing high variability.

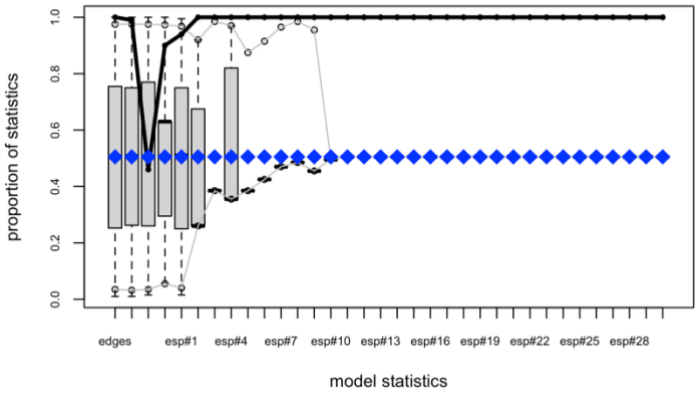
  Overall, this is a good starting model to enhance further towards feature agnostic relationship identification.



Model Improvement - 1:

Geometrically Weighted Edge Shared Partners (GWESP) is used to identify triadic relationships with better control and decay parameters governing new edge contribution to the existing relations amongst the vertexes.

```
Model: 3 -- gwesp.tri   triadic relations exists
Call:
ergm(formula = asNetwork(.mgraph) ~ edges + nodefactor("type") +
    nodematch("name2") + gwesp(decay = 0.01, fixed = F), control = e0.ctrl)

Monte Carlo Maximum Likelihood Results:

                    Estimate Std. Error MCMC % z value Pr(>|z|)
edges                -4.2651     0.4893      1  -8.717   <1e-04 ***
nodefactor.type.cex   0.7581     0.4911      1   1.543   0.1227
nodefactor.type.dex  -1.1098     0.5016      1  -2.213   0.0269 *
nodematch.name2     -11.1338     0.9367      0 -11.886   <1e-04 ***
gwesp                -0.2231         NA     NA      NA       NA
gwesp.decay          52.1058     0.9370      1  55.608   <1e-04 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     Null Deviance: 10856906  on 7831602  degrees of freedom
 Residual Deviance:    26685  on 7831596  degrees of freedom

AIC: 26697  BIC: 26780  (Smaller is better. MC Std. Err. = 18.79)
-------------------------------------------------------------
probability
            edges nodefactor.type.cex nodefactor.type.dex     nodematch.name2
      1.385632e-02        6.809357e-01        2.479004e-01        1.460951e-05
             gwesp         gwesp.decay
      4.444519e-01        1.000000e+00
*************************************************************
```



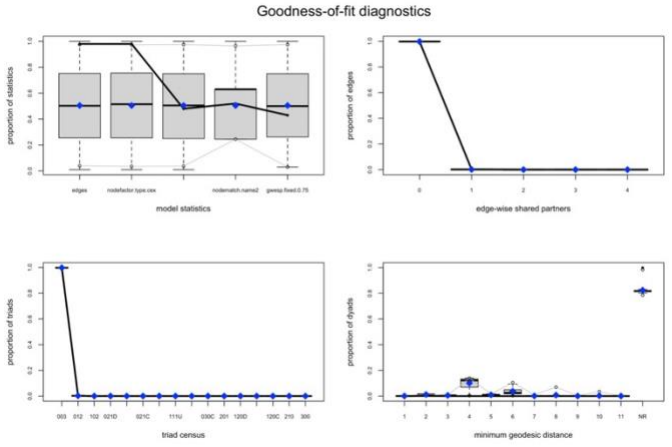
esp#7 is showing both in the g.o.f and mcmc.diagnostics summary that gwesp.decay optimal parameter is around .75 to 0.90. Given the dataset size, .9 did not converge but gwesp=.75 seems to be a good fit.

```
Model: 3 -- gwesp.tri
Sample statistics summary:

Iterations = 883200:2828800
Thinning interval = 12800
Number of chains = 8
Sample size per chain = 153

1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:

                          Mean     SD Naive SE Time-series SE
edges                -9.7377 47.359  1.35366        6.86722
nodefactor.type.cex  -6.7500 45.389  1.29735        6.82931
nodefactor.type.dex  -0.6503 11.221  0.32074        0.89431
nodematch.name2       1.1879  1.503  0.04296        0.04675
gwesp                 0.0000  0.000  0.00000        0.00000
gwesp.decay           2.7958  1.434  0.04098        0.16036

2. Quantiles for each variable:

                       2.5%     25%     50%   75%  97.5%
edges              -115.0000 -40.000 -8.000 23.00 75.425
nodefactor.type.cex -114.0000 -33.250 -5.000 25.00 72.000
nodefactor.type.dex  -23.0000  -8.000  0.000  7.00 22.000
nodematch.name2       -1.0000   0.000  1.000  2.00  4.000
gwesp                  0.0000   0.000  0.000  0.00  0.000
gwesp.decay            0.4462   1.785  2.677  3.57  6.247


Are sample statistics significantly different from observed?
                edges nodefactor.type.cex nodefactor.type.dex nodematch.name2 gwesp
diff.      -9.7377451          -6.7500000          -0.6503268    1.187908e+00     0
test stat. -1.6290020          -1.0945105          -0.7773907    2.607192e+01   NaN
P-val.      0.1033126           0.2737312           0.4369283   7.593248e-150   NaN
            gwesp.decay       (Omni)
diff.      2.795844e+00           NA
test stat. 1.713510e+01 7.143501e+02
P-val.     8.120795e-66 7.957428e-41
```

```
Model: 3 -- gwesp.tri
Call:
ergm(formula = asNetwork(.mgraph) ~ edges + nodefactor("type") +
    nodematch("name2") + gwesp(decay = 0.75, fixed = T), control = e0.ctrl)

Monte Carlo Maximum Likelihood Results:

                    Estimate Std. Error MCMC % z value Pr(>|z|)
edges                -5.9939     0.6411      1  -9.349  < 1e-04 ***
nodefactor.type.cex   2.4875     0.6428      1   3.870 0.000109 ***
nodefactor.type.dex   0.6188     0.6445      1   0.960 0.336999
nodematch.name2      -9.7799     1.2505      0  -7.821  < 1e-04 ***
gwesp.fixed.0.75     -1.2082     0.3584      0  -3.371 0.000749 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     Null Deviance: 10856906  on 7831602  degrees of freedom
 Residual Deviance:    26613  on 7831597  degrees of freedom

AIC: 26623  BIC: 26693  (Smaller is better. MC Std. Err. = 2.044)
-------------------------------------------------------------
probability
            edges nodefactor.type.cex nodefactor.type.dex     nodematch.name2
      0.0024876922        0.9232615320        0.6499543450        0.0000565742
  gwesp.fixed.0.75
      0.2300182354
*************************************************************
```

Improved – 1 Model: With model Gwesp=.75 we see in the left that the model is a good fit, with probability of 23% and better AIC/BIC.

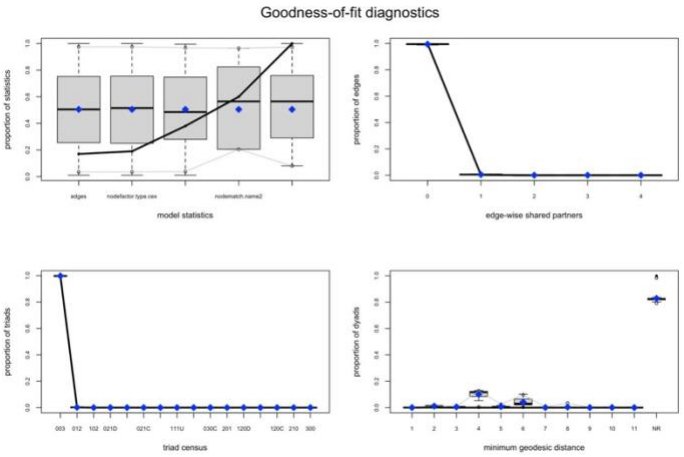
The goodness of fit is showing close estimation to the observed exchange name/type transactions seen so far in the largest component network.



Model Improvement - 2:

Another way of finding triadic relationships is by looking at the vertex pairs instead of edges, which is done by Geometrically Weighted Dyadwise Shared Partner Distribution (DGWDSP). A static GWDSP is a combination of GWESP and GWNSP. Choosing 'D' version of GWDSP allows us to specify bi-directional relationship assessment i.e., Reciprocated Two-path ("RTP") defined as: vertex k is an RTP shared partner of ordered pair (i,j)(i,j) iff i↔k↔ji↔k↔j.

```
Model: 5 -- dgwdsp
Call:
ergm(formula = asNetwork(.mgraph) ~ edges + nodefactor("type") +
    nodematch("name2") + dgwdsp(decay = 0.75, fixed = T, type = "RTP"),
    control = e0.ctrl)

Monte Carlo Maximum Likelihood Results:

                     Estimate Std. Error MCMC % z value Pr(>|z|)
edges                 -3.1287     0.4198      0  -7.452  < 1e-04 ***
nodefactor.type.cex   -0.3787     0.4182      0  -0.905  0.36524
nodefactor.type.dex   -2.2497     0.4427      0  -5.082  < 1e-04 ***
nodematch.name2      -12.7875     1.0953      0 -11.675  < 1e-04 ***
gwdsp.RTP.fixed.0.75  -0.7398     0.2867      0  -2.580  0.00987 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     Null Deviance: 10856906  on 7831602  degrees of freedom
 Residual Deviance:    26590  on 7831597  degrees of freedom

AIC: 26600  BIC: 26669  (Smaller is better. MC Std. Err. = 1.619)
-------------------------------------------------------------
probability
            edges  nodefactor.type.cex  nodefactor.type.dex       nodematch.name2
     4.193701e-02         4.064473e-01         9.537326e-02         2.795487e-06
gwdsp.RTP.fixed.0.75
     3.230384e-01
*************************************************************
```



The model decay parameter is found like GWESP and fixed to .75 showing good fit covering both edgewise shared partners and non-edgewise shared partners.

Model Improvement - 3:

```
Model: 7 -- name2factor
Call:
ergm(formula = asNetwork(.mgraph) ~ edges + nodefactor("name2") +
    gwesp(decay = 0.01, fixed = F) + gwdsp(decay = 0.01, fixed = F),
    control = e0.ctrl)

Monte Carlo Maximum Likelihood Results:

                          Estimate Std. Error MCMC % z value Pr(>|z|)
edges                    -11.3743     0.1613     12 -70.524   <1e-04 ***
nodefactor.name2.ABCC      7.7162     0.2176      9  35.465   <1e-04 ***
nodefactor.name2.Binance  10.5730     0.1671     12  63.273   <1e-04 ***
nodefactor.name2.Bitmax    9.3547     0.1725     11  54.237   <1e-04 ***
nodefactor.name2.Bittrex   7.1221     0.2338     11  30.457   <1e-04 ***
nodefactor.name2.Cashierest  6.0531     0.2680      6  22.589   <1e-04 ***
nodefactor.name2.Coinhako  3.6254     0.7970     13   4.549   <1e-04 ***
nodefactor.name2.DDEX      7.1364     0.2097     19  34.024   <1e-04 ***
nodefactor.name2.Faa       0.2781     1.3516     10   0.206    0.837
nodefactor.name2.Gate      5.2518     0.3678     11  14.279   <1e-04 ***
nodefactor.name2.Huobi     5.5944     0.2585     15  21.643   <1e-04 ***
nodefactor.name2.IDEX      5.2272     0.4940      6  10.580   <1e-04 ***
nodefactor.name2.Joyso     4.2888     0.6211     11   6.905   <1e-04 ***
nodefactor.name2.Paradex   5.3873     0.4240      5  12.707   <1e-04 ***
nodefactor.name2.Star      0.7506     1.1817      9   0.635    0.525
nodefactor.name2.Switchain  1.6944     1.2650      9   1.339    0.180
nodefactor.name2.Upbit     7.9923     0.2183      8  36.616   <1e-04 ***
gwesp                     -2.6722     0.4731      2  -5.649   <1e-04 ***
gwesp.decay               26.2966         NA     NA      NA       NA
gwdsp                     -0.0130         NA     NA      NA       NA
gwdsp.decay               40.5571     0.9281     48  43.698   <1e-04 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     Null Deviance: 10856906  on 7831602  degrees of freedom
 Residual Deviance:    17899  on 7831581  degrees of freedom

AIC: 17941  BIC: 18232  (Smaller is better. MC Std. Err. = 27.13)
-------------------------------------------------------------
probability
                  edges          nodefactor.name2.ABCC      nodefactor.name2.Binance
           0.0000114872                   0.9995546435                   0.9999744031
  nodefactor.name2.Bitmax     nodefactor.name2.Bittrex nodefactor.name2.Cashierest
           0.9999134544                   0.9991935607                   0.9976549891
nodefactor.name2.Coinhako        nodefactor.name2.DDEX          nodefactor.name2.Faa
           0.9740534307                   0.9992050283                   0.5690672001
    nodefactor.name2.Gate       nodefactor.name2.Huobi         nodefactor.name2.IDEX
           0.9947891002                   0.9962951309                   0.9946604053
   nodefactor.name2.Joyso     nodefactor.name2.Paradex         nodefactor.name2.Star
           0.9864648512                   0.9954465662                   0.6793121719
nodefactor.name2.Switchain       nodefactor.name2.Upbit                      gwesp
           0.8448046828                   0.9996620470                   0.0646323340
            gwesp.decay                          gwdsp                    gwdsp.decay
           1.0000000000                   0.4967509837                   1.0000000000
*************************************************************
```

Looking at the final factorization of name2 and dropping address with gwesp and gwdsp, we see both triadic relationship (edgewise and dyadic vertex wise) exists amongst each exchanges i.e. there are entities which only trade amongst themselves via certain exchange like Binance and never trade outside. This kind of closed loop transactions are crucial to capture with the probabilities listed in the bottom. 99% probability exchanges can be first monitored and inspected like Binance in April 1st 2023 (a month ago) is under investigation by government.

Predictability: If the model flags new transaction through this exchange as unexpected, it will be certainly worth looking at as among thousands of daily transaction a particular edge (to and from address) the model thinks its unusual.

5. CONCLUSIONS AND FUTURE WORK

Following future enhancements are worth pursuing for this ergm models and network data.

- Egocentric Networks: Pick top few token addresses and use the entire chain as multi layered networks
- Valued ERGM: Edge weights are not considered so far, ergm-count package is capable of honoring different weights to different edges while determining triadic relationships. This is in particular interest of following high value transactions in "sum" between two vertex, irrespective of their individual edge weights as micro transactions.
- R-GNN and R-GCN: Relational- Graph Neural Networks and Relational – Graph Convolutional Networks preserving more than immediate vertex relationship.

REFERENCES

[1] T.W.Liao, Clustering of time series data: A survey, Pattern Recog., vol.38, no. 11, pp. 1857-1874, Nov. 2005.

[2] B.D.Fulcher, and N.S.Jones, Highly comparative feature-based time-series classification, IEEE Transactions on Knowledge and Data Engineering., vol 26, no. 12, pp. 3026-3037, Dec 2014.