MA 710 Data Mining

Furthering Network Analysis

of

Ethereum Exchanges’ Transaction Data

Soubhik Chakraborty

Spring 2023

Abstract — The Ethereum Network Data Analysis in the group project used only Exponential Random Graph Models (ERGM) for modeling. To distinguish between heavy edges (high value transactions) Vs light edges, we will explore Valued-ERGM (Pavel N. Krivitsky, n.d.) based potential models, which due to hardware resource constraints couldn't be finished. It is worth looking into it because while vertex attributes driven modeling is very good in explaining the existing dataset, it is not necessarily relevant towards "out of ordinary" anomaly detection and, hence flagging suspicious transactions. A better approach will be to model the overall network shape and interactions using statistical properties of the network itself.

Furthermore, we will see egocentric networks and its potential to do both types of modelling viz. capture per token's transaction blockchain network properties as well as, overall common nodes across all the tokens cumulative pattern of trades, thus money flow.

1. INTRODUCTION/MOTIVATION

Previous Model Recap:

```
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
************************************************************
```

One of the key limitations of the above model is for any *vertex: i,j* edges like *j->k* and *k->l* transaction values as weights cannot be used in ordinary ERGM because most methods are node oriented like “nodematch”, “absdiff”.

2. THE DATA SET

Towards a fair comparison, exact data is used from the previous effort & advance the existing model such that edges are not lost via sampling, instead use the largest component representative of the “highest” important conglomerate of transactions worth investigating.

3. MODEL

Before jumping into model construct, lets briefly look at ergm.control settings.

Note: [Contrastive Divergence]() maximum iteration is set to 10.

```
ergm.ctrl = control.ergm(
    MCMLE.maxit = 3
    ,CD.maxit = 10
    ,MCMC.interval = 50
    ,MCMC.burnin = 40
    # ,MCMC.samplesize = 80
    # ,MCMC.effectiveSize=20
    ,MCMC.effectiveSize.maxruns=200
    ,MCMLE.density.guard=5000
    ,MCMC.runtime.traceplot=T
    ,init.method = "CD"
    ,parallel=cores, parallel.type="PSOCK"
    ,checkpoint="lc1.1.%02d.RData"
    # , resume="lc1.3.03.RData"
  )
```

Lets start with a basic edge value model ‘sum’ that uses edge attribute ‘weight’ and references Poisson distribution. As this itself could not be finished in 8hrs time, further modelling is solely based on documentation descriptions of each methods.

```
  enhanced.1 <- ergm(asNetwork(.mgraph) ~ sum + nonzero + mutual("min")
          + nodefactor("type")
          + nodematch("address")
          ,response="weight", reference=~Poisson, verbose=TRUE
```

```
        , control = ergm.ctrl
        , san = control.san(SAN.maxit=50))
```

Because we have seen geometric weighted edgewise shared partners a good fit, lets change this model to use geometric means instead of simple sum. Notice vertex attributes like name2 and type is totally ignored, which is suitable to our purpose. Also, mutual flow is now geometric mean based i.e., if outflow of money is disproportionately different than inflow or vice versa.

```
enhanced.2 <- ergm(asNetwork(.mgraph) ~ geomean + nonzero + mutual("geomean")
        , response="weight", reference=~Poisson, verbose=TRUE
        , control = ergm.ctrl
        , san = control.san(SAN.maxit=50))
```

Now let’s pick up relevant methods to determine triadic relationships in a more valued manner.

As a token that is being traded amongst multiple stake holders, the valuation of each trade is expected to follow some geometric mean instead of simple sum based arithmetic mean, whereby overall appreciation of a trade is exponential by nature.

```
enhanced.2 <- ergm(asNetwork(.mgraph) ~ geomean + nonzero + mutual("geomean")
         + transitiveweights("geomean","sum","geomean")
         + cyclicalweights("geomean","sum","geomean")
        , response="weight", reference=~Poisson, verbose=TRUE
        , control = lc1.1.ergm.ctrl
        , san = control.san(SAN.maxit=50))
```

Using the equation (13) definition of transitiveweights (Krivitsky, 2012), we can see combine function “sum” is more suitable while value of a “two-path” and final effective pressure is offsetted out between extremes using geometric mean. The modeling is still Poisson based distribution.

Th above models can be tried using Geometric reference as well and observe how mcmc.diagnostics and goodness of fit diagnostics indicate.

4. MULTINET

Picking the top 9 tokens most traded in the largest components of 50,000 ethereum transactions, first lets label the tokens with simplistic characters [A-I] thus below:

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

Figure 1

token_address short labels

	token_address	freq	label	short.token_addr
1	0xc02aaa39b223fe8d0a0e5c4f27ead9083c756cc2	19991	A	0xc02aaa39..083c756cc2
2	0x0d8775f648430679a709e98d2b0cb6250d2887ef	3679	B	0x0d8775f6..250d2887ef
3	0x0000000000085d4780b73119b644ae5ecd22b376	3382	C	0x00000000..5ecd22b376
4	0x9f8f72aa9304c8b593d555f12ef6589cc3a579a2	1680	D	0x9f8f72aa..9cc3a579a2
5	0xb8c77482e45f1f44de1745f52c74426c631bdd52	1614	E	0xb8c77482..6c631bdd52
6	0x89d24a6b4ccb1b6faa2625fe562bdd9a23260359	1368	F	0x89d24a6b..9a23260359
7	0xa15c7ebe1f07caf6bff097d8a589fb8ac49ae5b3	989	G	0xa15c7ebe..8ac49ae5b3
8	0xa0b86991c6218b36c1d19d4a2e9eb0ce3606eb48	727	H	0xa0b86991..ce3606eb48
9	0x6f259637dcd74c767781e37bc6133cd6a68aa161	635	I	0x6f259637..d6a68aa161

Figure 2

Assortativity Measures of different token networks

	type	name2
A	0	0
B	-0.17441517	-0.10991923
C	-0.19521138	-0.17518343
D	0	0
E	-0.97518083	-0.34315559
F	0	0
G	-0.43555901	-0.18109587
H	-0.77905403	-0.27803332
I	-0.57206766	-0.22896018

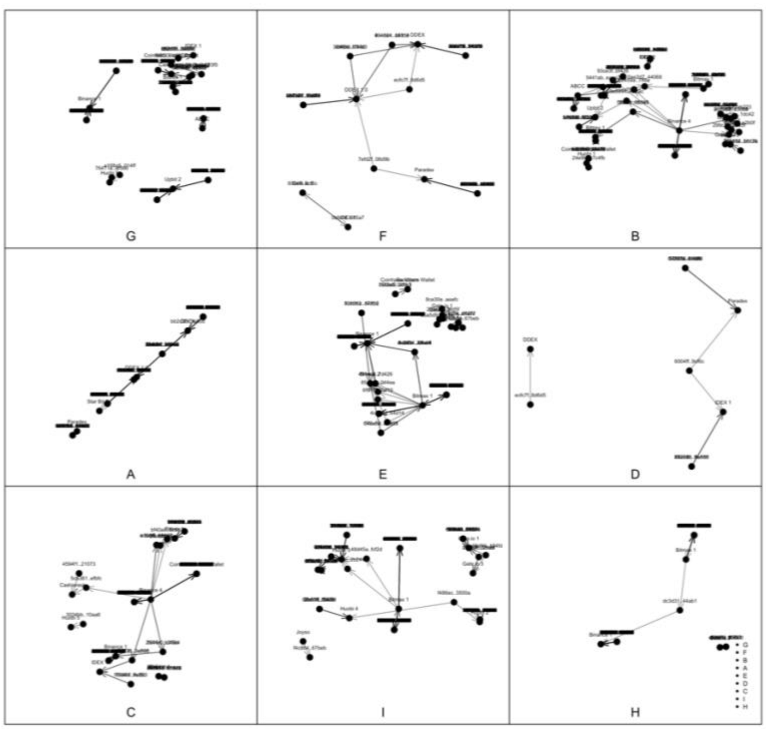
Now having created 9 subgraphs from [A-I], lets look at their isolated token networks' assortativity w.r.t. exchange type and name2.

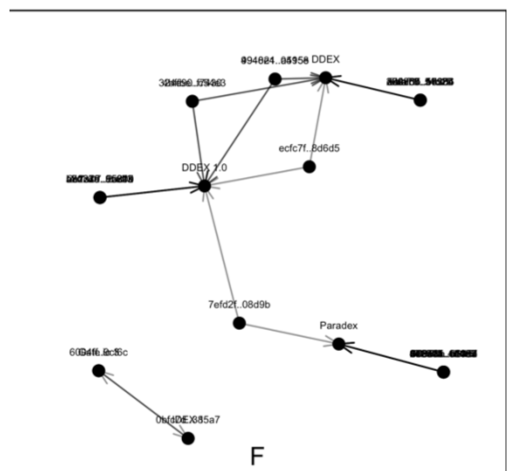
Summary of the multi network

	n	m	dir	nc	slc	dens	cc	apl	dia
*_flat_*	2465	2738	1	2	2441	0.00045079	2.298e-05	7.57651372512269e+22	1.99808837788e+26
A	104	119	1	2	96	0.01110904	0	1875511796611613184	1.0666111e+20
B	561	556	1	9	461	0.0017698	3.57e-05	1.57750076048983e+22	8.29642e+23
C	671	671	1	4	663	0.00149254	1.719e-05	1.26159484356307e+22	2e+24
D	11	9	1	2	9	0.08181818	0	361348076969821312	1.03e+18
E	579	582	1	6	565	0.00173907	0.00011229	1.31039264248002e+21	1.1e+24
F	30	31	1	3	26	0.03563218	0	1.44118699037466e+21	1.02583704455879e+22
G	311	302	1	9	136	0.00313246	0	1.14689393791219e+25	2.82272337672228e+27
H	210	210	1	2	205	0.00478469	0	11204150110.3534	1.3e+12
I	253	258	1	4	222	0.00404668	0.00014704	8.83309200509925e+21	1.69999e+23

Also, lets look at the summary statistics of the same multinetwork. **Token F** seems to be quite dense while having just 30 n/w size.

Plotting the multinet shows top row middle column Graph - (F) token is showing **convergence of funds into single exchange DDEX**.





Now lets calculate Jeffrey degree and measure Pearson correlation degree among these networks.

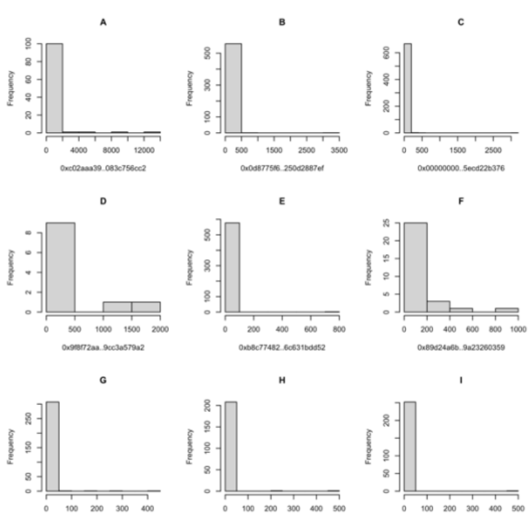
Jeffrey Degree

	G	F	B	A	E	D	C	I	H
G	0	0.00026058	0.00088254	1.012e-05	3.859e-05	0.00026058	0.00726842	8.005e-05	0.00065078
F	0.00026058	0	1.273e-05	0.11588482	1.195e-05	0.08705593	8.9e-06	1.565e-05	9.114e-05
B	0.00088254	1.273e-05	0	0.01344919	1e-08	1.273e-05	5.266e-05	1.5e-07	0.00296331
A	1.012e-05	0.11588482	0.01344919	0	0.00025176	0.00084434	0.01519251	0.00023617	9.56e-05
E	3.859e-05	1.195e-05	1e-08	0.00025176	0	1.195e-05	2.3e-07	2.5e-07	3.708e-05
D	0.00026058	0.08705593	1.273e-05	0.00084434	1.195e-05	0	8.9e-06	1.565e-05	9.114e-05
C	0.00726842	8.9e-06	5.266e-05	0.01519251	2.3e-07	8.9e-06	0	9.5e-07	0.00384357
I	8.005e-05	1.565e-05	1.5e-07	0.00023617	2.5e-07	1.565e-05	9.5e-07	0	3.125e-05
H	0.00065078	9.114e-05	0.00296331	9.56e-05	3.708e-05	9.114e-05	0.00384357	3.125e-05	0

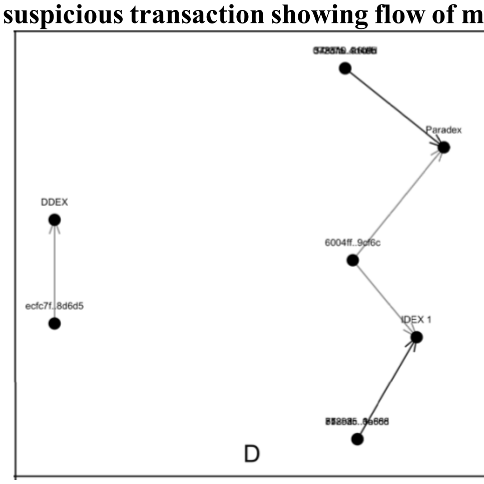
We can see token G, E, C,B and A are highly correlated overall transaction history.

Pearson Degree

	G	F	B	A	E	D	C	I	H
G	1	NaN	0.87470359	NaN	0.999936	1	0.99191835	0.55441595	1
F	NaN	1	0.61237244	0.87705447	NaN	0.17524942	0.08935989	NaN	NaN
B	0.87470359	0.61237244	1	0.99988468	0.73585534	0.40824829	0.9954855	0.69026073	0.91248566
A	NaN	0.87705447	0.99988468	1	NaN	-0.13910801	0.55440062	NaN	NaN
E	0.999936	NaN	0.73585534	NaN	1	NaN	0.9986636	0.9994707	0.94898768
D	1	0.17524942	0.40824829	-0.13910801	NaN	1	0.57396402	NaN	NaN
C	0.99191835	0.08935989	0.9954855	0.55440062	0.9986636	0.57396402	1	NaN	1
I	0.55441595	NaN	0.69026073	NaN	0.9994707	NaN	NaN	1	0.99968445
H	1	NaN	0.91248566	NaN	0.94898768	NaN	1	0.99968445	1



Plotting the degree histogram, we see that D and F have some isolated varying degree, otherwise all other token transaction history is uniform and not showing much disparity in power_law. Therefore D becomes another **suspicious transaction showing flow of money between Paradex and IDEX**.



Sorted by XRelevance

	degree_ml	neighborhood_ml	xneighborhood_ml	xrelevance_ml
Paradex	21	12	5	0.41666667
IDEX 1	17	10	2	0.2
DDEX 1.0	59	55	8	0.14545455
Gate.io 3	10	8	1	0.125
DDEX	79	67	1	0.01492537
Star Bit Ex	1	1	NA	NA
Bitmax 1	570	550	NA	NA
Joyso	9	7	NA	NA
ABCC	68	61	NA	NA
Binance 4	990	949	NA	NA
Gate.io 1	16	13	NA	NA
Binance 1	685	587	NA	NA
Huobi 5	20	19	NA	NA
Huobi 2	17	17	NA	NA
Faa.st	2	1	NA	NA
Switchain	5	2	NA	NA
Huobi 1	12	12	NA	NA
Upbit 2	80	78	NA	NA
Huobi 4	6	6	NA	NA
Coinhako: Warm Wallet	4	4	NA	NA
Bittrex 1	46	43	NA	NA
Cashierest	20	19	NA	NA
Bitmax 2	9	7	NA	NA

Looking at the exclusive relevance metric which essentially demonstrate how these token networks' vertices (entities) are highly localised to the exchanges i.e. not present outside its own network layer.

5. CONCLUSIONS AND FUTURE WORK

We saw that Valued-ergm has a better promising model and multinetwork statistics already started showing signs of suspicious transactions when we see in parts of the whole and whole divided into parts.

In future, following things is worth trying:

- Try out bootstapped ergms (btergm) and egocentric ermg (egoERGM) on ego.subgraphs
- Lookout for R-GNN and R-GCN implementations in R.

REFERENCES

Bibliography

Krivitsky, P. N. (2012). *Exponential-family random graph models for valued networks*. Retrieved from National Library of Medicine: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3964598/

Pavel N. Krivitsky, C. T. (n.d.). *ERGMs for Valued Networks with Applications to Count Data*. Retrieved from cran-r: https://cran.r-project.org/web/packages/ergm.count/vignettes/valued.html