# Oracle Counterpoint: Relationships between On-chain and Off-chain Market Data

No Author Given

No Institute Given

**Abstract.** We investigate the theoretical and empirical relationships between activity in on-chain markets and pricing in off-chain cryptocurrency markets (e.g., ETH/USD prices). The motivation is to develop methods for proxying off-chain market data using data and computation that is in principle verifiable on-chain and could provide an alternative approach to blockchain price oracles. We explore relationships in PoW mining, PoS validation, block space markets, network decentralization, usage and monetary velocity, and on-chain liquidity pools and AMMs. We select key features from these markets, which we analyze through graphical models, mutual information, and ensemble machine learning models to explore the degree to which off-chain pricing information can be recovered entirely on-chain. We find that a large amount of pricing information is contained in on-chain data, but that it is generally hard to recover precise prices except on short time scales of retraining the model. We discuss how even a noisy trustless data source such as this can be helpful toward minimizing trust requirements of oracle designs.

## 1 Introduction

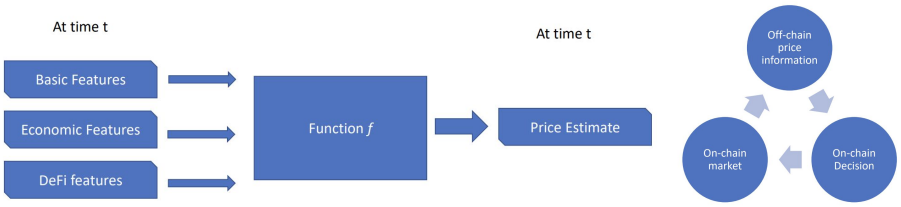
Decentralized finance (DeFi) aims to transfer the role of trusted but risky intermediaries to more robust decentralized structures. A remaining weak link is in reliance on off-chain information, such as prices of reference assets, which need to be imported on-chain through oracles. The issue is that oracle-reported prices cannot be proven on-chain because the price process (usually in USD terms) is not observable there.

Various oracle security models exist, as described in [14], though for the most part, they always involve some sort of trusted party or medianizing of several trusted parties. Even alternatives like referencing time weighted average prices (TWAPs) on decentralized exchanges (DEXs) still essentially involve a trusted party. In particular, to price an asset in USD terms, the standard approach is to use a DEX pair with a USD stablecoin, but this is just equivalent to treating the stablecoin issuer and mechanism as the trusted oracle, and the estimate can be wrong.

In this paper, we explore a new direction in oracle design wherein an *estimate* of an off-chain price can in principle verifiable on-chain. We investigate the theoretical and empirical relationships between activity in on-chain markets and the overall pricing and liquidity in off-chain cryptocurrency markets (e.g.,

$f$BTC, ETH$g$/USD price. The motivation is to develop methods for proxying off-chain market data using data within an on-chain environment.

We formalize this as the task of finding a function $f$ that maps on-chain observable data to close estimates of off-chain prices, as visualized in Figure 1a. Ideally, a good $f$ will also have two further properties: (i) it is difficult/costly to manipulate the output of $f$ through manipulating the inputs, and (ii) outputs of $f$ are provable on-chain. The hypothesis predicating this structure is that off-chain price data (e.g., in USD terms) is incorporated into the behavior of agents in on-chain markets (e.g., mining, block space, and DeFi markets) and that on-chain data thus provides some information that can be recovered about the original off-chain prices, as visualized in Figure 1b.




(a) (b)

Fig. 1: Proposed structure to estimate prices verifiably on-chain.

To understand the problem intuitively, compare with the usual financial price prediction problem, in which we would try to identify several drivers of future price and formulate a model to predict future prices with these drivers as features. The problem we consider is the reverse in some ways. In particular, we hypothesize that the price is a driving factor (probably one of many) behind the behavior of agents in on-chain markets, and we want to recover the current period price from the current state of on-chain market behaviors as features.

We explore this problem using a combination of economic theory about on-chain markets and data-driven analysis to explore the degree to which off-chain pricing information can be recovered from on-chain data. We find a meaningful price signal is recoverable as well as several strong empirical relationships with on-chain features. While it is not precise enough to use directly as an oracle, we discuss ways in which it could be used as a strustless sense check for oracle-reported prices. We finish by discussing several significant challenges that remain in developing and executing such a tool.

## 2 Methods

We explored relationships in PoW mining, PoS validation, block space markets, network decentralization (e.g., burden on running a full node), usage and monetary velocity, and DeFi liquidity pools and AMMs, including activity on both

Bitcoin, Ethereum, and Celo networks. We obtained raw block and transaction data from Google Cloud Bigquery, Uniswap v1 and v2 data from the Graph, and off-chain USD price data from the Coinbase Pro API. We then derived the following types of on-chain data features:

- *Basic network features* that can be derived from Ethereum block and transaction data directly, covering information related to Ethereum's network utility, ether supply in circulation, transaction cost and the network's computational consumption (i.e. the gas market).
- *Uniswap features* on participation in DEX pools involving ETH and stablecoins (DAI, USDC, USDT). For the most part, we intentionally do not focus on DEX prices, as those measures would equivalently treat the stablecoin issuer as a sort of trusted oracle. We instead mainly focus on a measure of liquidity moving in and out of DEX pools.
- *Economic features* as described in the next subsection.

Data was collected spanning from July 1 2016 to May 1 2022 and was aggregated to the hourly level. We include Bitcoin data along with Ethereum data in the dataset for the sake of exploring relationships as in principle it can also be verified on-chain to varying degrees and discuss the connections further later.

Some further details on data and features are provided in the appendix. Precise methods will be available in a github repo.

### 2.1 Fundamental Economic Features from On-chain Markets

In addition to the above raw on-chain features, we also considered transformations of these features informed by fundamental economic models of on-chain markets, including PoW mining, PoS validation, block space markets, network decentralization costs of running full nodes, usage and monetary velocity, and on-chain liquidity pools (e.g., [7,13,2,3,6,4]). We analyzed the structure of these models to extract features that should economically be connected to price.

For example, [7] models a block space market and finds that the ratio of average demand to capacity $= \frac{}{K}$ plays an important role in linking users' waiting costs to transaction fees pricing. Here is the transaction volume, $K$ is the maximum number of transactions in a block, and is the block adding rate. A function emerges, which we'll call $F(\ )$ that describes the relationship between fee pricing and congestion, which can be translated as

$$\text{tx fees in USD} = (\text{tx fees in ETH}) \quad \text{price}_{ETH} = F(\ ):$$

While $F(\ )$ is nontrivial to work with, various pieces of the results in [7] can be incorporated into useful features for the task of recovering $\text{price}_{ETH}$, including , $^2$, and the empirical finding that $= 0.8$ represents a phase transition in fee market pricing.

We also used the model in [2], which modeled cryptocurrency price based on market fundamentals. A key feature in this model was currency velocity, which

is defined as the ratio of transaction volume to money supply:

$$Velocity = \frac{\text{Transaction Volume}}{\text{Exchange Rate} \quad \text{Supply of Currency}}:$$

Based on this model, we also incorporated the ratio of transaction volume and cryptocurrency supply as an additional feature in our analysis.

We formulate other factors related to mining payoff, computational burden, and congestion as reviewed in the appendix.

### 2.2 Data-driven Feature Analysis

We analyse empirical relationships between features using graphical models and mutual information to study which features are most related to USD prices.

We use Markov random fields, generated through sparse inverse covariance estimation with graphical lasso regularisation over normalized data, to express the conditional dependency (partial correlations) between the time series of on-chain features and off-chain prices. The output of this technique helps to uncover strong empirical dependencies within the data, suggesting features that are strongly related to price and others that replicate similar information as others. We find that the method is often sensitive to the precise dataset used, which we adjust for by smoothing over the outputs of many $k$-fold subsets.

We also consider mutual information between features in the dataset, which describes the amount of information measured (in information entropy terms), measured in reduction of uncertainty, obtained about price by observing the on-chain features. In information theory, entropy measures how surprising the typical outcome of a variable is, and hence the 'information value'. This is helpful both in identifying strong relationships and evaluating different smoothing factors considering noisy on-chain signals. In this analysis, we consider smoothed versions of the feature set based on exponential moving averages with memory parameters , i.e., for feature value $b_t$ at time $t$, the smoothed measure is

$$\tilde{b}_t = (1 \quad )b_t + \quad \tilde{b}_{t \ 1}:$$

### 2.3 Modeling O -chain Prices

We apply supervised machine learning methods to explore the degree to which off-chain pricing information can be recovered from information that is entirely on-chain. We apply a few select simple and ensemble supervised machine learning methods on a rolling basis: basic regression, single decision tree, random forest, and gradient boost. The motivation for using tree-based ensemble methods is the non-parametric nature of the dataset and success of similar methods in analyzing other market microstructure settings [5].

We run these models on the data set and evaluate performance using out-of-sample testing data on a rolling basis. The rolling training-testing data split, as depicted in Figure 2, is applied to boost model performance. For a given set of

time series data with time duration of time t + time c = time t+c, where time series before time t were used for model training and time series between time t and time t + c were used for model testing. The benefit of this split is to test how good the model is in proxying ETH USD price for a fix period in the future, with all the information available in the past.


Fig. 2: Rolling training-testing data split

## 3 Results

We focus on Ethereum data analysis under PoW in this section. Analysis of Celo data is included in the appendix as a first look at a PoS system. There is not yet enough historical data to analyze Ethereum PoS but would be a next step.

### 3.1 Feature Analysis

We find that a large amount of off-chain pricing information is contained in on-chain data and that the various features are connected in some strong but complicated ways.

Figure 3 and Appendix Figure 7 show the results of sparse inverse covariance modeling for a selection of the feature set. The graphical structure depicted is the consistent structure over time as smoothed over the outputs of many $k$-fold subsets. The partial correlation matrix shows the graphical structure in matrix form. In the graphical model, the features that are most connected with ETH/USD price include number of active to and from addresses sending transactions, block difficulty, and number of transactions per block. Many other strong relationships are also exhibited among the various other features, potentially indirectly connected to price.

Figure 4 shows the mutual information between ETH/USD prices and other features, meaning the amount of information (reduction of uncertainty) obtained about price by observing each other variable individually. We find that across the top 10 features, a large amount of information about off-chain price is contained in on-chain data. We also find that the mutual information decreases with ,

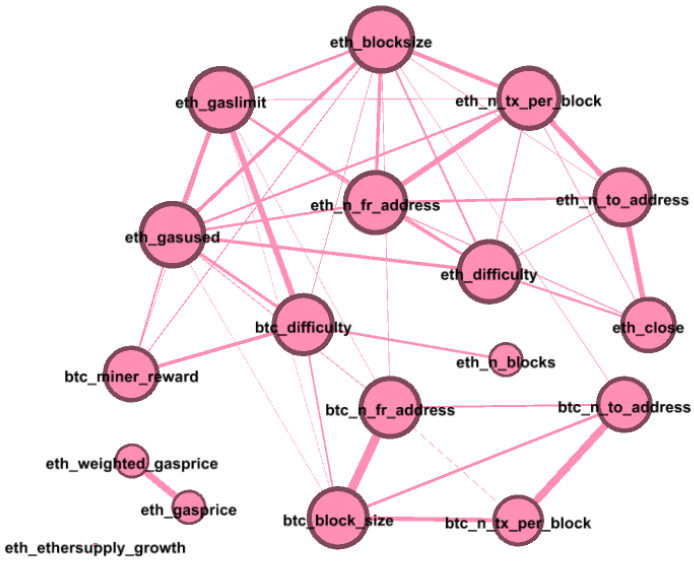



Fig. 3: Graphical network visualization.

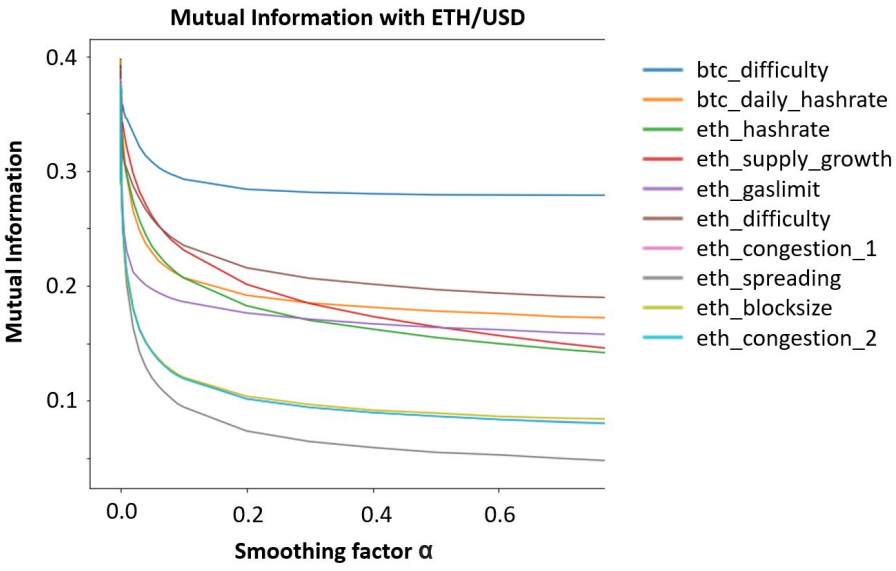



Fig. 4: Mutual information of price data and features, with smoothing .

the exponential moving average memory factor for smoothing, indicating that the smoothed data is generally less informative than the most up-to-date data.

We also analyze the full feature set, including the transformed economic factors and Uniswap pool liquidity factors. Perhaps unsurprisingly, since the transformed features contain the same underlying information, they do not exhibit stronger relationships than the raw features. More surprising is that the Uniswap pool factors also did not present strong relationships with price. We then arrived at the above version of the analysis excluding Uniswap factors enabling us to use the entire data history (as Uniswap was launched later than the start of the dataset).

### 3.2 Recovering O -chain Prices from On-chain Data

Random forest and gradient boost both outperformed the other two simpler ML algorithms. We selected Random Forest as the candidate model in the end as it is in principle simpler to be implemented on-chain compared to the gradient boost model (theoretically, a random forest model could be implemented as one big mapping table in a smart contract).

We tested the model performance over different lengths of period - the length of time duration between time t and time t+c. As would be expected with nonstationary time series, we observed that the longer the time duration that a single trained model is used for price estimation, the less accurate is price estimation. The degree to which time between retrainings affects accuracy is informative, however.

Figure 5 shows the random forest model performances, Estimated vs Actual ETH/USD price, for 1-day ahead, 1-week ahead and 1-month ahead of retrainings. While none of the models provide high accuracy of recovering ETH prices, they do demonstrate that a good signal of the general price level can be recovered, particularly in the 1-day and somewhat in the 1-week retraining cases.

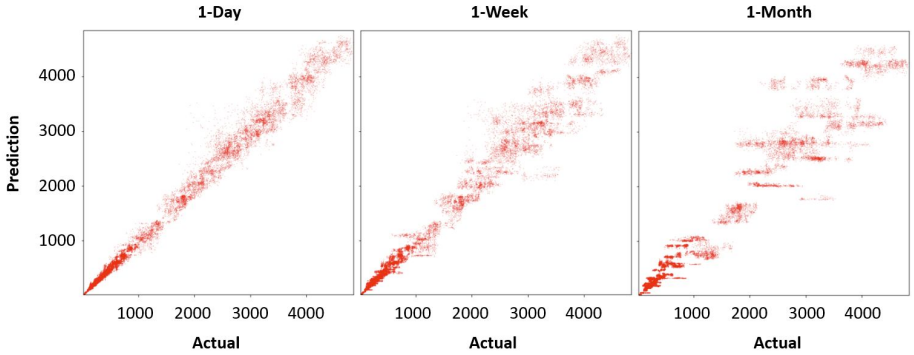



Fig. 5: Recovered price vs actual for random forest with given retraining periods.

The deviation between estimated price and actual price is bigger for higher ETH prices. This is a combination of both having less data in the dataset for these prices and the fact that the same relative error scales with the absolute price, and so deviations measured absolutely are expected to be greater.

We run the models on the full feature set, including transformed economic factors and Uniswap pool factors. The economic factors provide little new information vs the raw features, perhaps a consequence of the flexibility of the tree models. Uniswap pool factors similarly do not improve accuracy. The final analysis excludes Unsiwap factors enabling the entire data history to be used.

## 4 Discussion

We find that a general, but noisy, signal of off-chain prices can be extracted from the on-chain feature set, although it remains difficult to extract precise prices from the noise. It is possible to improve the accuracy of the model by including features of DEX pricing of ETH/stablecoin pairs, as would be expected from [1]. However, this is antithetical to the approach, as this would implicitly rely on the assumption that 1 stablecoin = 1 USD. Such models would face the significant further issues of detecting stablecoin depeg events (such as happened in USDC in March 2023) given that data is sparse for such events.

While this approach could likely not be used as a direct price oracle, the information from the recovered price signal could still be useful as a sense check to inform when other oracle-reported prices may be suspect. This function would be potentially very useful in application as the most profitable oracle manipulations to date have been large manipulations that may be caught by such methods. Oracle systems of this style have indeed been proposed [10], although using other measures than the price signal we uncover. Such a method could also serve to better align the incentives of an oracle provider to report correct prices with the knowledge that their quality of their feed is being graded against the signal in on-chain information. Models such as [8,9] could model this analytically, interchanging the oracle provider with the governors in those models.

Several challenges remain for implementing and running such a mechanism in practice. One is accessing all the data within the EVM. Some of the data is in principle possible to access but may be too computationally intense under current systems. For instance, proving information about transactions or bridging BTC data might require running light clients on-chain. For BTC data, this can mostly be ignored as it wasn't critical for the predictive models, but there was a lot of information in Ethereum transaction statistics. It's worth noting that some features such as gas prices are easier to access now with EIP 1559. Another challenge is in evaluating how manipulable the features are should a bad actor want to affect the price estimation. In principle, resilient measures seem possible, though may also be computationally burdensome to produce.

An implementation would also have to handle the rolling nature of retrainings required to accurately recover price data. THe implementation would need a trust minimized way to update a smart contract implementation with new

trainings. In principle this is also possible, such as by implementing the training program in fixed point to run deterministically and implementing a way to prove the correctness of a training on-chain. However, this would be daunting from the technical side as well as likely costly to run in most environments. The burden could possibly be eased by running it 'optimistically' by incorporating a challenge period and fraud proofs, though it's unclear if this would be enough of an improvement. Another viable way is for a trusted trainer to regularly update calibrations on-chain subject to on-chain spot checks and not full proofs.

## References


1. Angeris, G., Chitra, T.: Improved price oracles: Constant function market makers. In: Proceedings of ACM Advances in Financial Technologies. p. 80–91 (2020)
2. Athey, S., Parashkevov, I., Sarukkai, V., Xia, J.: Bitcoin pricing, adoption, and usage. Working Paper No. 3469 (17) (2016)
3. Buterin, V.: Blockchain Resource Pricing pp. 1–32 (2018), `https://ethresear.ch/uploads/default/original/2X/1/197884012ada193318b67c4b777441e4a1830f49.pdf`
4. Easley, D., O'Hara, M., Basu, S.: From Mining to Markets: The Evolution of Bitcoin Transaction Fees (2018). https://doi.org/10.1007/s10551-015-2769-z.For
5. Easley, D., López de Prado, M., O'Hara, M., Zhang, Z.: Microstructure in the machine age. The Review of Financial Studies **34**(7), 3316–3363 (2021)
6. Fanti, G., Kogan, L.: Economics of Proof-of-Stake Payment Systems (2019)
7. Huberman, G., Leshno, J.D., Moallemi, C.: An Economic Analysis of the Bitcoin Payment System*. SSRN Electronic Journal pp. 1–60 (2019)
8. Huo, L., Klages-Mundt, A., Minca, A., Münter, F.C., Wind, M.R.: Decentralized governance of stablecoins with closed form valuation. In: MARBLE (2022)
9. Klages-Mundt, A., Harz, D., Gudgeon, L., Liu, J.Y., Minca, A.: Stablecoins 2.0: Economic foundations and risk-based models. In: ACM AFT. pp. 59–79 (2020)
10. Klages-Mundt, A., Schuldenzucker, S.: Design of the gyroscope consolidated price feed and circuit breaker system (2022)
11. Kroll, J.a., Davey, I.C., Felten, E.W.: The Economics of Bitcoin Mining, or Bitcoin in the Presence of Adversaries. WEIS 2013 (Weis), 1–21 (2013). https://doi.org/June 11-12, 2013
12. Nicolas, H.: The Economics of Bitcoin Transaction Fees. SSRN Electronic Journal (2014). https://doi.org/10.2139/ssrn.2400519
13. Prat, J., Benjamin, W.: An Equilibrium Model of the Market for Bitcoin Mining. CesIfo Working Papers (January), 26 pages (2017)
14. Werner, S.M., Perez, D., Gudgeon, L., Klages-Mundt, A., Harz, D., Knottenbelt, W.J.: SOK: Decentralized finance (defi). In: ACM AFT (2022)


## A More Details on Dataset Features

Figure 6 and Table 1 provide more information on the feature set used.

| | CCY | Source | Starting From | Frequency |
|---|---|---|---|---|
| On-Chain | BTC block data | Google BigQuery | Jan 2016 | Hourly |
| | ETH block data | | | |
| | CELO block data | Celo Graph (block, celoTransfers) | Apr 2020 | |
| | cGLD transaction data | | | |
| | cUSD transaction data | | | |
| | Uniswap liquidity and balance data | The Graph | Aug 2020 | |
| Off-Chain | BTC price and volume data | Coinbase API | Jan 2016 | |
| | ETH price and volume data | | | |
| | Celo price and volume data | | Sep 2020 | |

Fig. 6: Overview of dataset.

[table of features, including the economic ones, refer to online appendix that will be provided with more details of underlying economic models]

| **Feature type** | **Feature (high level description)** |
|---|---|
| Network | Number of blocks |
| | Number of transactions |
| | % change in accumulated ETH supply |
| | Avg gas limit |
| | Avg gas used |
| | Avg gas price |
| | Hash rate |
| Uniswap | Liquidity in ETH/stablecoin pools |
| | Trade volume in ETH |
| Economic | Mining pay-off factors |
| | Computational burden measures |
| | Congestion factors |
| | Social cost factors |
| | Spreading factor |

Table 1: Data features.

Online documentation in the project github repo will provide further details of the underlying economic models and calculation of the economic factors (as well as calculation of other factors from the raw data). A brief overview is as follows along with citations for the relevant models that influenced the choice of these features.

- Mining payoff factor 1: $(R(\text{blockReward} + \text{blockFees}))^{1}$ [11,13]
  R = block rate (/s), eth_n_blocks = # blocks in the last hour
- Previous high hash rate / current hash rate
- previous high $(R(\text{blockReward} + \text{blockFees}))^{1}$=current
- Excess block space (block limit - gas used)

- Social value: D(W) is the social value of the level of decentralization = D(W) = - log(W) =) D(W) = - log(gas_used) for ethereum, = - log(bytes); gas used as the measure of the weight of a block (W) [3]

- Social cost: Marginal cost = 1/gas_used or 1/bytes [3]

- Computational burden on nodes: use block_size as bandwidth =) block_size $\log^2$(block_size) [3]

- Congestion factors: rho = gas used/gas limit, and $\text{rho}^2$; (in economic model, rho is defined as average number of transaction per block / number of transactions per block) [7]

- Congestion factor: Indicator_*f*rho > *xg*, heuristic use x = 0.8 [7]

- Congestion pricing term 1: F(rho) / tx_fees_eth, where F describes relationship between USD tx fees and congestion [7]

  Heuristic: use F = congestion factor 1 or 2 above

- Congestion pricing term 2: max number of transactions in a block / fees in block [12]

- Congestion pricing term 3: max number of transactions squared in a block / fees in block [12]

- Spreading factor: number of unique output addresses / number of unique input addresses [2]

## B Further figures on Ethereum Analysis

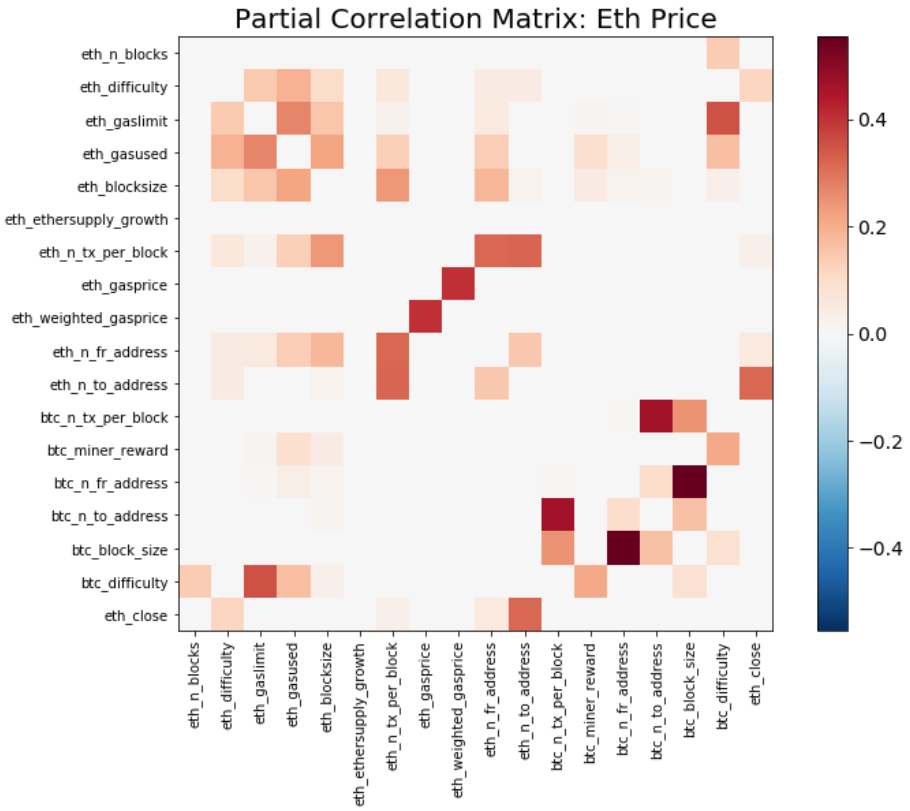



Fig. 7: Partial correlation matrix from sparse inverse covariance estimation.

## C Analysis of Celo PoS Data

In addition to Ethereum data, we also analyse data on the Celo PoS network. This analysis involves some further features involving PoS systems as well as Celo's dual token model. This additionally serves as a first look at the analysis of a PoS system with historical data spanning longer than a year. In comparison, a similar analysis of Ethereum's new PoS system does not yet have enough history at the current time to perform a good analysis.

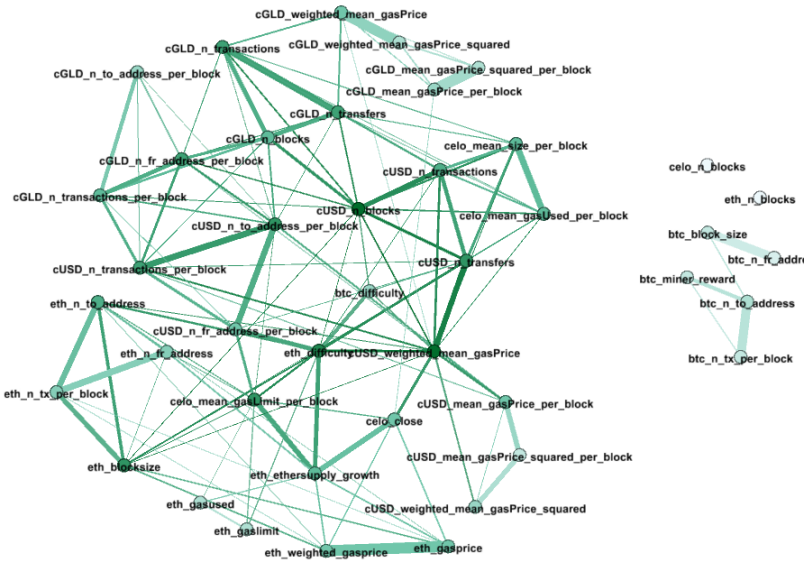



Fig. 8: Graphical network visualization from sparse inverse covariance estimation.

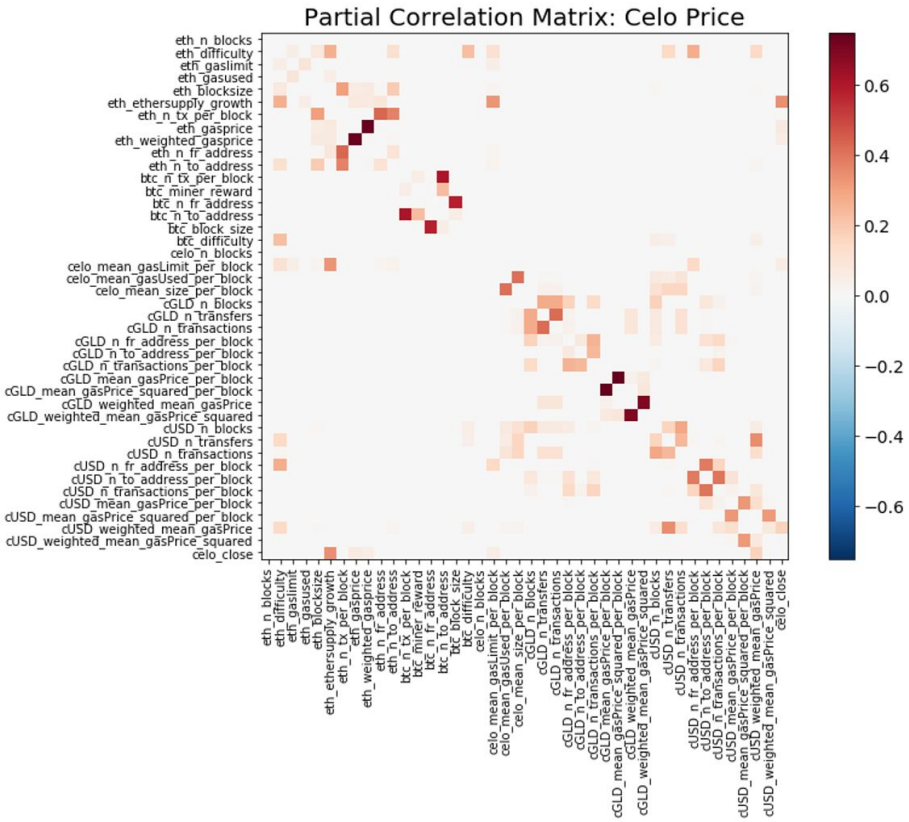



Fig. 9: Partial correlation matrix from sparse inverse covariance estimation.