Bitcoin fair value model and attempt at it's application

Samuel Tkáč

February 24, 2026

Contents

Abstract

The question of Bitcoin's fundamental value remains unanswered. Critics argue that it does not possess any intrinsic value; therefore, it should be traded at zero. Conversely, Bitcoin has experienced mainstream adoption through exchange-traded funds (ETFs) and has appeared on the corporate balance sheets of so-called Bitcoin treasuries. This gap between theory and practice is profound and begs further investigation.

The objective of this project is to develop a model for estimating the "fair price" of Bitcoin based on fundamental factors.

1 Fundamental factors

A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a most efficient effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone. [8]

p = probability an honest node finds the next block

q = probability the attacker finds the next block

q_z = probability the attacker will ever catch up from z blocks behind [8]

$$\lambda = z \cdot \frac{q}{p} \qquad q_z = 1 - \sum_{k=0}^{z} \frac{\lambda^k e^{-\lambda}}{k!} \left(1 - \left(\frac{q}{p}\right)^{z-k}\right) \text{[8]}$$

if honest miners have the majority of computational power validating blocks, then the probability of attackers successfully attacking exponentially decreases. This will be our assumption : $p > q$.

1.1 Derivation of model

Main idea

Building on insights in the white paper, a high level of honest hash rate is crucial for the security of the whole network. Miners maintaining high levels of hash rate face costs that need to be covered by their rewards, which is the basic idea of this valuation. Essentially, figure out how much miners need to sell their bitcoin for in order to be profitable.

Floor calculation

We will try to calculate the floor at first. Pick the most efficient machine money can buy, and calculate how many machines of this type need to be running to achieve the network hash rate. Multiply that by their daily consumption and the estimated electricity price. We will distribute these electricity costs across accrued bitcoins. This is the floor conditioned on electricity cost.

Efficiency and margin

Since not everyone can use the most efficient machines, adjustment is necessary. Adjustment will be in the form of an efficiency coefficient that is the estimated machine efficiency divided by the efficiency of the chosen machine.

Machines need proper facilities with personnel to be operational, and they need to make a profit so that miners can acquire better tech eventually. We need to account for this with a margin coefficient.

Continuity

Since price is continuous, valuation should be too. When a new efficient rig drops and halving occurs, there will be jumps in the model that need investigation. The solution is polynomial interpolation $(3x^2 - 2x^3)$, for the machines, we will use 120 days, and for emissions, 500 days. This solution will affect the model in the following ways for machine interpolation, the switch from one machine being most efficient will not be done instantly when it is for sale, but step by step over 120 days. This will likely not affect the model because of the efficiency coefficient adjustment, but the emission interpolation will steadily increase the model over 500 days before halving. We are talking about a possible feature, not a bug, because miners are anticipating changes in subsidy, and this might reflect it.

Data, Sources and Assumptions

- Hash-rate dataset via Blockchain.info API [3]
- Market-price dataset via Blockchain.info API [4]
- Bitcoin miners data from Bitmain [2]
- Miner efficiency from the Cambridge Bitcoin Electricity Consumption Index (CBECI) [6]
- Emission schedule based on halving history [1]
- Electricity cost assumption: $0.05 per kWh
- Margin coefficient assumption : $\frac{5}{2}$
- Fees as part of the subsidy are assumed to be zero during the application, since outside volatile periods, they contribute a minuscule amount relative to block emission. These spikes are an unreliable source of subsidy.

Plots are all on a y-log scale

2 Model Details

$H_n(t)$	network hashrate at day t	(TH/s)
$H_m(t)$	miner hashrate at day t	(TH/s)
$P_m(t)$	miner power consumption at day t	(kw/d)
$\eta_m(t)$	miner efficiency at day t	(J/TH)
$\eta_n(t)$	estimated network efficiency at day t	(J/TH)
$C(t)$	estimated electricity price at day t	(\$ per kWh)
$E(t)$	daily network emission at day t	(btc)
$F(t)$	fees accrued at day t	(btc)

$$N(t) := \frac{H_n(t)}{H_m(t)} \qquad \beta(t) := \frac{\eta_n(t)}{\eta_m(t)} \qquad \alpha \in \mathbb{R}^+$$

$$M(t) = \frac{N(t) \cdot P_m(t) \cdot C(t) \cdot \beta(t)}{E(t) + F(t)} \cdot \alpha$$

Figure 1: example, $\alpha = 2.5$

In (1) we can observe the model acting as a price mean. Price tends to mean revert when the deviation is large. As we might expect, the model shows overvaluation during bull markets and undervaluation during bear markets.

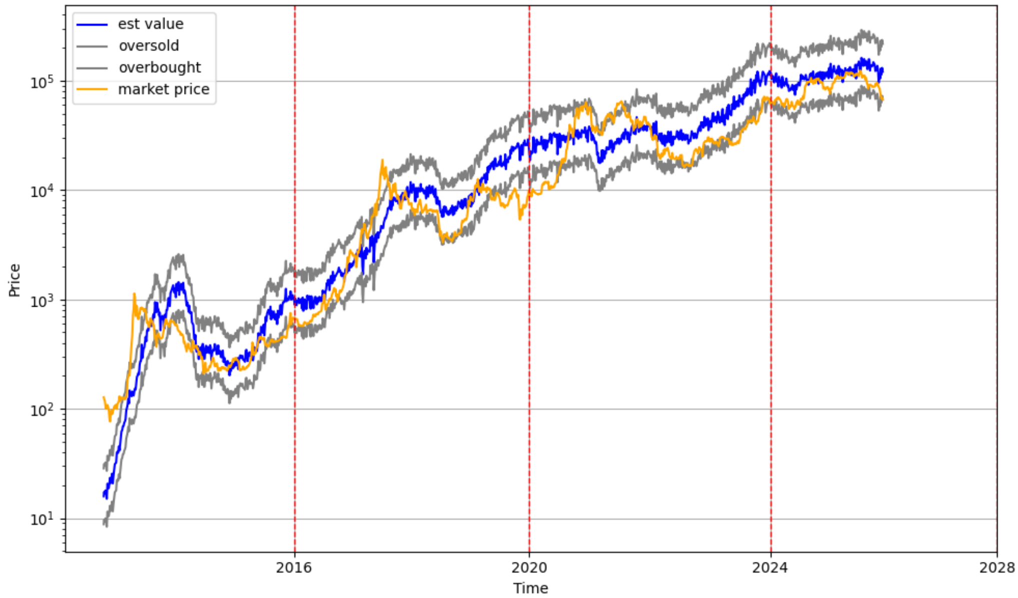

Figure 2: reverting bands, $\zeta = 1.8$

Figure (2) bands that are multiples of the model contain the price:

$$\begin{aligned} \text{Upper band (overbought)} &: M(t) \cdot \zeta^{1} \\ \text{Lower band (oversold)} &: M(t) \cdot \zeta^{-1} \\ &\text{where } \zeta \in (1, \infty) \end{aligned}$$

3 Application

Let's define a bull market as the last visited band being oversold and vice versa. In bear and bull markets, segment further with a method for regime classification and detection. Fair valuation would be used to classify the overall market state, and another method could be used to further distinguish market regimes. This could provide an edge.

We will use a hidden Markov model with the baum-welch algorithm for parameter estimation. We will assume 3 hidden regimes, and the distributions are univariate Gaussian distributions, with the data being daily percentage price changes.

The application involves simple parameter estimation in a bear or bull market and then classification in the following bull or bear market, done using the forward algorithm.

Since the Blockchain API provides data every 4 days, that isn't what we need for HMM, so the data for modeling will be used from Yahoo Finance [5]

Baum-Welch

$$\mathcal{N}(x \mid \mu_j, \sigma_j^2) := \frac{1}{\sigma_j\sqrt{2\pi}} e^{\frac{\left(x-\mu_j\right)^2}{2\sigma_j^2}}$$

Forward backward algorithm

$$\begin{array}{cc} \alpha_1(i) = \pi_i \mathcal{N}(x_1 \mid \mu_i, \sigma_i^2) & \alpha_{t+1}(j) = \mathcal{N}(x_{t+1} \mid \mu_j, \sigma_j^2) \sum_{i=1}^{N} \alpha_t(i) A_{ij} \\ \beta_T(i) = 1 & \beta_t(i) = \sum_{j=1}^{N} A_{ij}\mathcal{N}(x_{t+1} \mid \mu_j, \sigma_j^2)\beta_{t+1}(j) \end{array} \quad [7] \tag{1}$$

Expectation

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{k=1}^{N} \alpha_t(k)\beta_t(k)} \qquad \xi_t(i,j) = \frac{\alpha_t(i)A_{ij}\mathcal{N}(x_{t+1} \mid \mu_j, \sigma_j^2)\beta_{t+1}(j)}{\sum_{i,j=1}^{N} \alpha_t(i)A_{ij}\mathcal{N}(x_{t+1} \mid \mu_j, \sigma_j^2)\beta_{t+1}(j)} \quad [7]$$

Maximization

$$\begin{array}{cc} \pi_i = \gamma_1(i) & A_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \\ \mu_i = \frac{\sum_{t=1}^{T} \gamma_t(i) x_t}{\sum_{t=1}^{T} \gamma_t(i)} & \sigma_i^2 = \frac{\sum_{t=1}^{T} \gamma_t(i)(x_t-\mu_i)^2}{\sum_{t=1}^{T} \gamma_t(i)} \end{array} \quad [7]$$

$$\text{regime at time t will be : } \underset{\forall i}{\operatorname{argmax}}\ \alpha_t(i) \quad (1)$$

Figure 3 is trained on the bear market of 2017-2018 and attempts to classify the regimes of the 2021-2022 bear market.

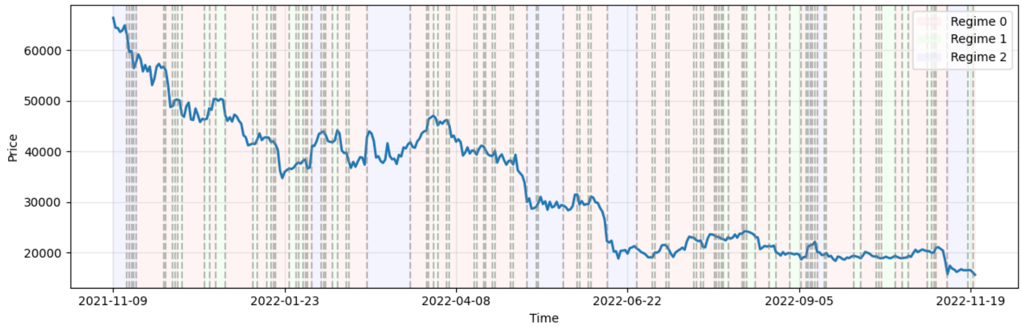

Figure 3: bear market

Figure 4 is trained on the bull market of 2018-2021 and attempts to classify the regimes of the 2022-2026 bull market

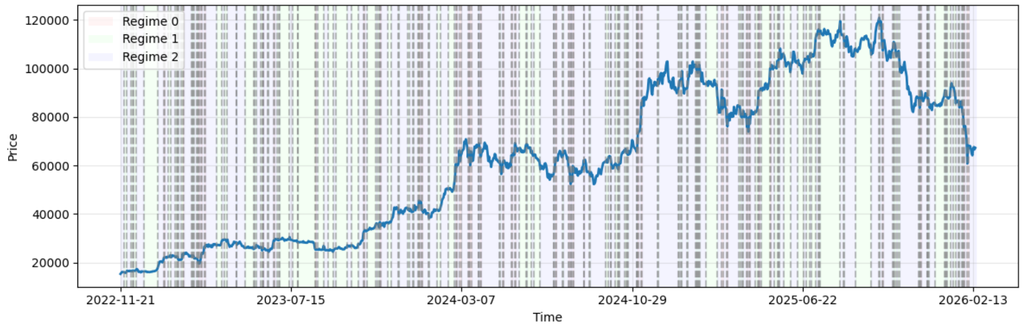

Figure 4: bull market

It is unclear what regime 0, regime 1 and regime 2 correspond to.

4 Conclusion

Subjectivity problem

The network hash rate, subsidy, and the most efficient machine are all unambiguous metrics, but metrics like average miner efficiency, electricity price, and margin coefficient are ambiguous. This means that the model could give you any valuation you want, conditioned on the data. For example, the widely used assumption $0.05 per kWh is wrong, but because of the unavailability of the dataset, it is a necessary evil.

Pros

This model values bitcoin via a supply-side proxy through miner expenses: "price is fair if those who uphold the network can flourish". As emission will reduce because of halving, fees will become the bedrock of miner subsidies. The bitcoin security budget problem might intensify if lost emission aren't at least replaced by fees. The price needs to double every halving, but that might be unsustainable. Sustainability might come from bitcoin's usage since the price may remain constant with the network having the same security budget if halving emission loss gets replaced by fees. Reduction in the security budget would endanger miners, which will lead to a decrease in the hash rate. This model might correctly decrease bitcoin value based on reduced security.

Cons

Bitcoin could become centralized even if miners do not face a security budget problem, but somehow monopolize mining. If bitcoin were to reprice from a decentralized network to a centralized network, this model might claim a devalued state despite our broken assumption: $p > q$. The model might not estimate this in any way because deterioration might not be caused by a decreasing hash rate in this case.

Potential improvements to model

A possible Solution to the valuation in a scenario where the network becomes centralized without a security budget problem might be a demand side model based on Metcalfe's law, Zipf's law, Reed's law, etc. Repricing from a decentralized network to a centralized network would likely leave evidence in user data, and thus the demand-side model might be indicative in this scenario. In the pros section, there is a mention of fees replacing the subsidy, and fees are collected when there are users, so the demand-side model might even be overall better than the supply-side since miners will eventually depend on fees as their subsidy.

Application

The result of the hidden Markov model application was not interpretable, probably because one feature, such as percentage daily change, might not explain a complex asset like bitcoin. The assumption of gaussian distributions didn't help either, since returns have fatter tails than a Gaussian distribution. Application could be improved by using a multivariate student-t model in case of a huge feature set, cleaning noise with PCA before using baum-welch. Since the implementation of the algorithm was done from scratch and since I am not a programmer, the solution to underflow might affect the algorithm.

References

[1] Bitcoin halving dates dataset. https://charts.bitbo.io/halving-dates/, 2026. Accessed: Feb 2026.

[2] Bitcoin miners data (bitmain). https://shop.bitmain.com/?coin=antminer, 2026. Accessed: Feb 2026.

[3] Blockchain.info hash-rate dataset. https://www.blockchain.com/explorer/charts/hash-rate?utm_source=chatgpt.com, 2026. Accessed: Feb 2026.

[4] Blockchain.info market-price dataset. https://www.blockchain.com/explorer/charts/hash-rate?utm_source=chatgpt.com, 2026. Accessed: Feb 2026.

[5] Btc-usd historical data. https://finance.yahoo.com/quote/BTC-USD/, 2026. Accessed: Feb 2026.

[6] Cbeci miner efficiency dataset. https://ccaf.io/cbnsi/cbeci, 2026. Accessed: Feb 2026.

[7] Guy Leonard Kouemou. *History and Theoretical Basics of Hidden Markov Models*. IntechOpen, 2010. Accessed: 2025-03-08.

[8] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf, 2008. Accessed: 2025-11-03.