Machine Learning for Bitcoin Pricing Classification

This paper follows the development process of various machine learning models, used to provide context on cryptocurrency pricing. Bitcoin is the first decentralized digital currency, built on blockchain technology to facilitate secure transactions without oversight from a central entity. However, Bitcoin prices are historically extremely volatile – driven by fluctuating supply and demand, along with relative novelty – and thus very difficult to predict. The innovative models in this paper *combine* data from various on-chain metrics (publicly-recorded blockchain data); indices, commodities, and Forex markets; and macroeconomic indicators to forecast the magnitude of Bitcoin pricing on a daily basis.

Author: Sohan Bendre

RCM Alternatives Internship Program 2025

Sponsor: Ryan Dillman | **Sponsor:** Tara Begeman

Abstract

This paper follows the development process of various machine learning models, used to provide context on cryptocurrency pricing. Bitcoin is the first decentralized digital currency, built on blockchain technology to facilitate secure transactions without oversight from a central entity. However, Bitcoin prices are historically extremely volatile – driven by fluctuating supply and demand, along with relative novelty – and thus very difficult to predict. The innovative models in this paper *combine* data from various on-chain metrics (publicly-recorded blockchain data); indices, commodities, and Forex markets; and macroeconomic indicators to forecast the magnitude of Bitcoin pricing on a daily basis. The study compares the performance of four different machine learning models implemented through Python libraries scikit-learn and tensorflow: Logistic Regression, Support Vector Machines (SVM), Tree-Based Algorithms, and Long-Short Term Memory (LSTM) recurrent neural networks. Results are quantified based on accuracy score, with logistic regression providing the highest accuracy at 55.94% on out-of-sample daily data from March 2024 to May 2025, and statistical significance across the full period.

Introduction

Bitcoin, the first and most noteworthy cryptocurrency, is increasingly becoming a pivotal financial asset, since its creation in 2009. As a decentralized asset operated independently of traditional financial institutions, Bitcoin has attracted widespread attention from professionals across various fields. Its price behavior, however, is notoriously volatile and driven by a complex network of macroeconomic trends, market sentiment, on-chain activity, and more. This volatility, in conjunction with the absence of intrinsic value or standardized valuation models, has fostered skepticism around Bitcoin's predictability. Yet really understanding Bitcoin's price movements has high significance for portfolio optimization, trading and risk management, and gaining key insights into digital asset behavior. By leveraging some machine learning pipelines across a variety of financial and digital indicators, this project uncovers latent patterns in Bitcoin's price movement.

Data Selection and Framework

Through data collection, the factors were categorized as either on-chain metrics, market data, or macroeconomic indicators.

On-Chain Metrics

- Active Addresses
- Exchange Holdings
- Exchange Netflow
- Fee-Reward Ratio
- Funding Rate
- *** Hashrate**
- *** Mean Coin Age**
- *** Miner Reserves**
- *** MVRV Ratio**
- *** Open Interest**
- Puell Multiple
- SOPR Ratio

Market Data

- *** S&P 500 Close**
- *** Nasdaq Close**
- *** Dow Jones Close**
- *** Gold Close**
- Crude Oil Close
- Nat Gas Close
- *** Silver Close**
- *** Copper Close**
- US Dollar Index
- EUR-USD Exchange Rate
- USD-CNY Exchange Rate

Macroeconomic Indicators

- **** Federal Funds Rate**
- **** Consumer Price Index (CPI)**
- **** 10Y Treasury Yield**
- Real GDP
- M2 Money Supply
- Unemployment Rate

Table 1: Variables marked with * were selected for our models. Variables marked with ** additionally required linear interpolation.

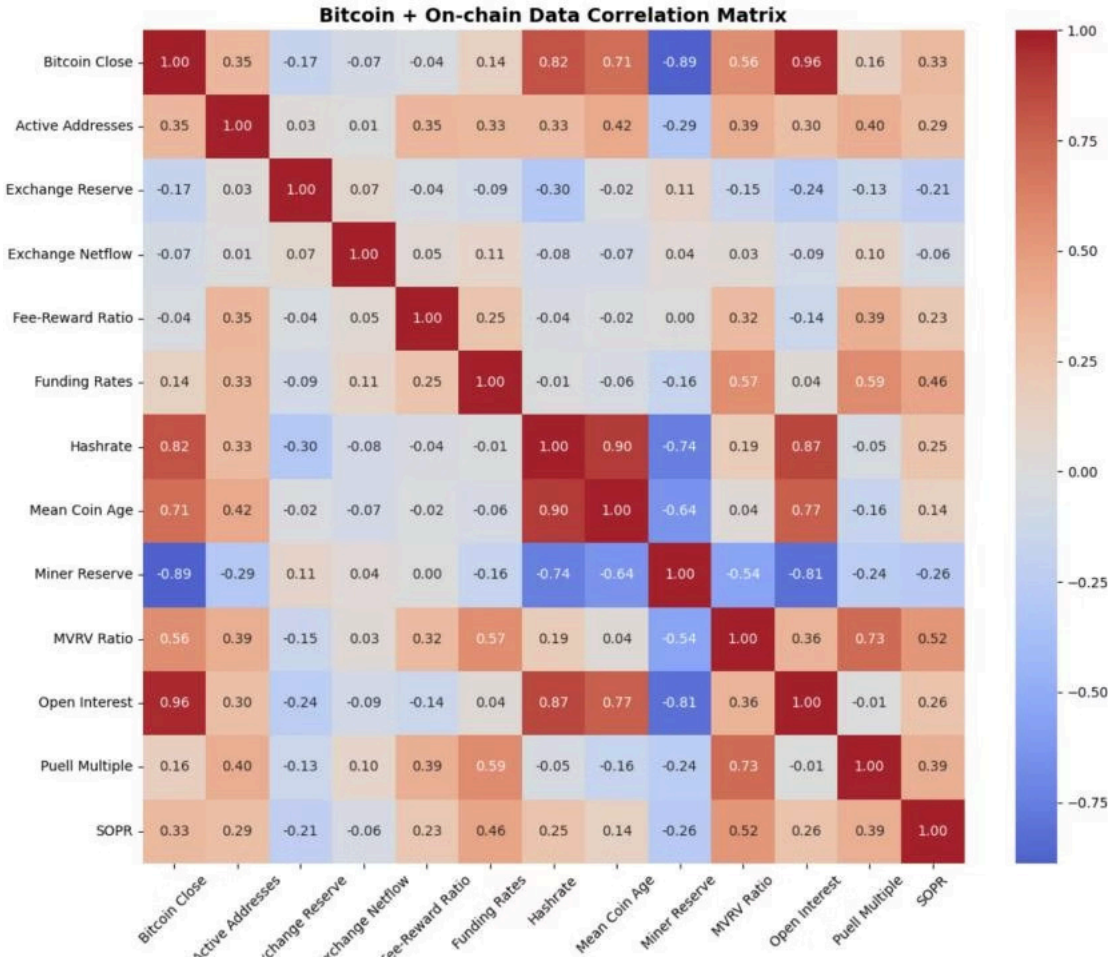
Understanding On-Chain Data

The Importance of Understanding UTXOs

A clear grasp of Unspent Transaction Outputs (UTxOs) is essential for anyone analyzing the Bitcoin blockchain. UTxOs are fundamental components of the Bitcoin transaction model. Each UTxO represents a discrete amount of bitcoin that has not yet been spent, functioning similarly to holding cash in assorted denominations. When a transaction occurs, these outputs are either fully consumed or returned partially as change, effectively creating new UTxOs. This structure enables transparent, verifiable ownership and contributes to network security by preventing issues such as double-spending. UTxOs can only be spent in totality, reinforcing the atomic nature of bitcoin ownership and transfer. They are the essential building blocks for all on-chain metrics, which reflect the state of Bitcoin from many different angles.

On-Chain Data Visualization

This organization and selection of on-chain metrics provide an interpretable feature set for quantitatively modeling Bitcoin price movements. All on-chain data, INCLUDING Bitcoin close prices, were collected from CryptoQuant, a trusted provider of cutting-edge on-chain and market data. Accurate data was available from March 30, 2019 to time of data collection (June 9, 2025).



Many interesting relationships between different factors can be observed here, but the top row is of main concern, which shows correlation specifically between Bitcoin close price and each on-chain factor. From this figure, Hashrate (cor 0.82), Mean Coin Age (cor 0.71), Miner Reserves (cor -0.89), and Open Interest (cor 0.96) are the variables with highest correlation magnitudes. This provides a comprehensive view of linear relationships between on-chain data, which informs feature selection.

RCM Alternatives

On-Chain Factors Used in Predictive Modeling

The following on-chain metrics were leveraged as explanatory variables in machine learning models to forecast directional changes in Bitcoin's price, backed by higher correlations.

Miner Reserve

Definition: Miner reserve denotes the total bitcoin held within wallets associated with mining entities.

Insight: This value reflects the portion of newly-mined coins yet to be liquidated. High miner reserves may indicate an expectation for higher future prices, while a decline (miners selling coins) could foreshadow selling pressure and potential downward price movement.

Hashrate

Definition: Hashrate is the average computational power employed by all miners, measured as hashes per second (commonly in exahashes).

Insight: Higher hashrate enhances network security and is often interpreted as miner confidence in future profitability. Significant changes in hashrate can signal shifts in mining economics or sentiment.

Mean Coin Age (MCA)

Definition: Mean Coin Age measures the average duration (weighted by value) that UTXOs remain unspent.

Computation: MCA is the mean of products between each UTXO value and its days since creation.

Insight: An increasing MCA suggests coins are being held rather than spent, indicating accumulation. Conversely, a sharp decline in MCA signals older coins are being moved, which is often correlated with heightened market activity or potential selloffs.

Open Interest

Definition: Open Interest is the aggregate number of open positions (both long and short) in derivative markets.

Insight: Rising open interest typically aligns with increased liquidity, volatility, and speculative capital entering the market. Sudden declines may precede long/short squeezes, amplifying price swings.

Brief Overview of Excluded Factors

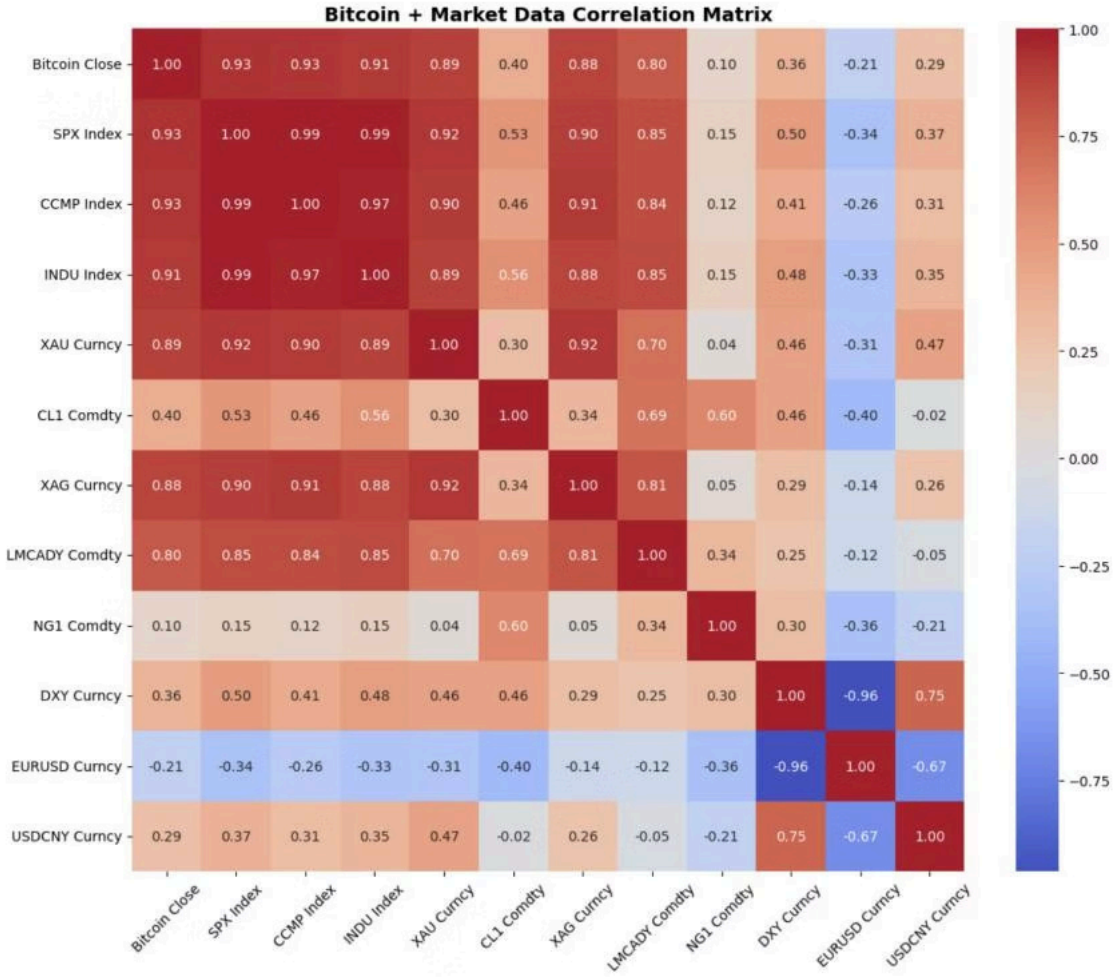
The following on-chain or derivative metrics, while informative, were excluded from most final models due to redundancy, lower predictive power, or diminished relevance:

Exchange Reserve	Tracks the total amount of bitcoin held on exchanges. Higher reserves often signal increased selling risk, but can be confounded by inter-exchange transfers.
Puell Multiple	Ratio of daily bitcoin issuance value (USD terms) to its 365-day moving average. Historically used to identify miner-driven market cycles.
Fee-Reward Ratio	The fee component of block rewards as a percentage of total block rewards, reflecting transaction demand and network congestion.
SOPR (Spent Output Profit Ratio)	Measures realized profit by comparing output values at spend time to those at creation. SOPR > 1 indicates coins are moved at a profit, useful for assessing market sentiment.
Active Addresses	The unique count of active sending and receiving addresses within a timeframe, often considered as a proxy for network usage and adoption.
Exchange Netflow	The net difference between bitcoin entering and leaving exchange addresses. Large net inflows are frequently associated with selling interest.
Funding Rates	Periodic payments exchanged between long and short traders to align perpetual futures prices with spot prices. These rates can reflect trading bias or momentum extremes.
MVRV Ratio (Market Value to Realized Value)	Compares total market capitalization to realized capitalization, helping to contextualize periods of over- or under-valuation in the market. ****Was used for support vector machines**

Market Data and Macroeconomic Indicators

Market Data

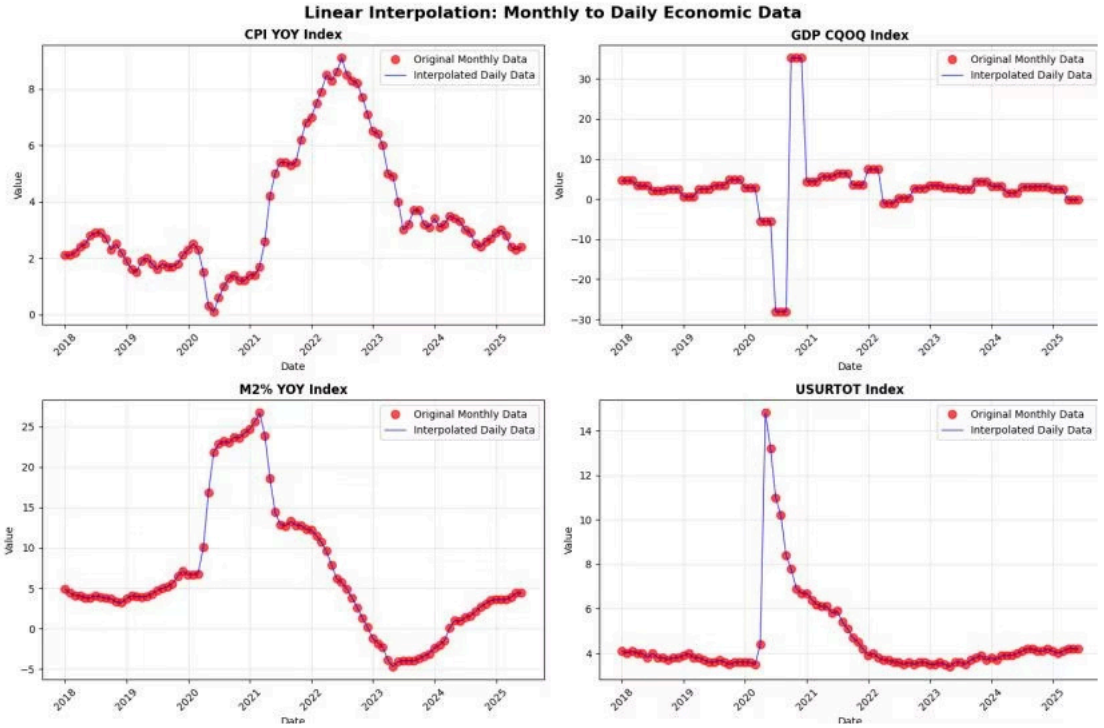
While on-chain data has been thoroughly analyzed in relation to Bitcoin pricing, also of great interest are the co-movements between cryptocurrencies and major markets / indices. Will Bitcoin prices be influenced by fluctuations in the S&P 500? Why have so many people been juxtaposing Bitcoin and gold? This study aims to answer these questions, and provide further context on latent relationships between Bitcoin and other indices, commodities, and forex markets. As found in Table 1, a variety of these factors have been selected, and data from Bloomberg has been compiled from January 02, 2017 to time of data collection (June 9, 2025).



From the figure, it's clear that strong co-movement exists between Bitcoin and the three major indices: S&P 500, Nasdaq, and Dow Jones. Additionally, high correlation is found between Bitcoin and commodities such as gold, silver, and copper. However, few notable results are found in relation to forex markets. While high correlations are promising, it remains to be seen if any of these markets can serve as leading indicators for Bitcoin fluctuations in an effective model.

Macroeconomic Indicators

Finally, it may prove influential to include certain macroeconomic indicators in pricing models, as factors such as inflation, interest rates, or even geopolitical tensions could change how investors approach Bitcoin and cryptocurrency markets as a whole. Many macroeconomic indicators are collected quarterly, so after accessing data from Bloomberg, linear interpolation was performed to standardize data on a daily basis.



These four features, along with Federal Funds Target Rate (FDTR Index) and US 10 Year Treasury Yield (USGG10yr Index), comprised the macroeconomic indicators evaluated in the study. Linear interpolation was not required for the additional two features.

Feature Engineering and Model Setup

Target Variable

The primary goal of the predictive models is to determine highest possible accuracy in forecasting **price direction** of Bitcoin close. This classification will be done as follows: for day d (the day from which our factors come), the target variable is 1 if the close price on day $d+1$ is higher than on day d. Otherwise, if the close price on day $d+1$ is less than day d, the target variable is 0. This can be represented mathematically as:

$$T_d = \max(0, \text{sign}(P_{d+1} - P_d)) \quad (1)$$

This effectively reflects the study's use of day d's values to forecast behavior on day $d+1$, ensuring there is no target leakage.

Factor Transformations

To align with the goal of capturing Bitcoin price direction, data must be transformed to indicate the dynamics of each on-chain metric, market, index, or macroeconomic indicator. Therefore, a multitude of transformations were applied to various factors, prior to input into machine learning models. Here is a succinct description:

- **On-Chain Data:** one-day percent change, three-day percent change, seven-day moving average, binary flags
- **Market Data:** daily percent returns (one-day percent change)
- **Economic Indicators:** one-day delta, raw form

Numerous combinations of these transformations were applied, in order to achieve highest testing set accuracy. The main intuition is simply that the study aims to capture dynamic relationships; this is best accomplished through the above transformations, which were individually evaluated and selected for each factor.

Boruta Feature Selection

Correlation matrices capture linear relationships between variables. However, the study also aims to evaluate non-linear relationships that may provide context towards Bitcoin's price movements: the attempt to select most impactful non-linear factors is done through Boruta feature selection.

Essentially, Boruta is a wrapper algorithm around a Random Forest, a machine learning classification method that will be explored later in this paper. Boruta runs many trials to test each individual feature, but additionally adds *shadow features*, which are randomized versions of each feature that essentially compete against their original versions. Only features *more influential* than their random vectors are awarded points during each trial; finally, the Boruta model will score each feature on a binomial distribution to determine significance.

Separate Boruta algorithms, each with a total of 50 trials, were run for on-chain factors and market factors.

The random forest accuracies are not too promising – this indicates the relationships between the study's factors and Bitcoin close may not be non-linear, in the vein captured by decision trees.

However, the Boruta simulations are still run to check for feature importance.

Observing the results above, very few features would be labeled as "significant" by the Boruta binomial distribution. But, considering the lower accuracy of the random forests in the previous table, the Boruta results must be regarded with a grain of salt. Another weakness in the study is the relatively small pool of factors from which Boruta can choose from: as an extremely precise algorithm, likely hundreds of on-chain / market factors would be required for Boruta to select a handful passing the significance test.

Overall, the study combines interpretations of correlation matrices and Boruta significance tests to select features, which can be found in Table 1. Now, let's use them to build predictive machine learning models.

Machine Learning Models and Results

Now that comprehensive data transformations and feature engineering methods have been performed, it is time to select models and evaluate their results. Since this is a classification problem, the study will use four prominent machine learning methods:

Logistic Regression (LR)
simple linear model that estimates probabilities using the sigmoid function for classification

Support Vector Machine (SVM)
an algorithm that calculates the optimal hyperplane in n-dimensional space, which separates data into two categories of maximal width

Tree-Based Algorithms
using a tree-based setup, these models divide feature space into hierarchical structures to make predictions, capturing abstract nonlinear patterns

Long Short Term Memory (LSTM)
a type of recurrent neural network (RNN) that can recognize nonlinear long-term dependencies in time-series data

For all models, the train-test split was 80%-20%. This means that the training period was from April 10, 2019 to March 7, 2024. The testing period, on which the accuracy score is measured, was from March 8, 2024 to May 29, 2025. This comes out to about 1282 training days, and 320 testing days. Before all models were run, a standard scaler was applied to the various data.

1. Logistic Regression

Since the goal is to predict a binary variable, logistic regression is the perfect supervised machine learning algorithm to act as a baseline for the study. Essentially, logistic regression is built over a linear model, applying the sigmoid function to output a value between 0 and 1.

$$\sigma(z) = \frac{1}{1+e^{-z}} \quad (2)$$

The sigmoid function is a classic activation function in machine learning. After applying the sigmoid, the threshold value to separate categories is usually 0.5.

But how is z calculated? The value is initialized as a linear combination of inputs, which were previously selected using feature engineering.

$$z = w^T \cdot X + b \quad (3)$$

These features are standardized and represented in matrix X. Meanwhile, vectors w and b serve as a set of weights and biases, which are used to transform the input matrix into output z.

However, the weights and biases need to be optimized, otherwise the model will perform terribly. To evaluate the current performance of weights w and biases b, a function called **binary cross entropy**, or **log loss**, is employed.

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \cdot \log(p_i) + (1-y_i) \cdot \log(1-p_i)] \quad (4)$$

Here, N is the number of observations, p_i is the probability outputted by the sigmoid function for day i, and y_i is the actual binary classification target variable on day i. The goal of logistic regression is to minimize average binary cross entropy, by recomputing w and b accordingly. To achieve this, let's apply **gradient descent**:

$$w_{k+1} = w_k - \eta \cdot \frac{\partial L}{\partial w} \quad (5)$$

$$b_{k+1} = b_k - \eta \cdot \frac{\partial L}{\partial b} \quad (6)$$

To calculate the weights and biases for iteration $k+1$, we use the above formulas, relying on their previous values. The learning rate, η, is usually set to 0.01. The derivatives dL/dw and dL/db are calculated by applying the chain rule to binary cross entropy in terms of w and b.

During logistic regression, weights w and biases b are iteratively updated using gradient descent until some optimization threshold. Python's scikit-learn library, used for this study, stops iteration when the gradient reaches below a convergence condition.

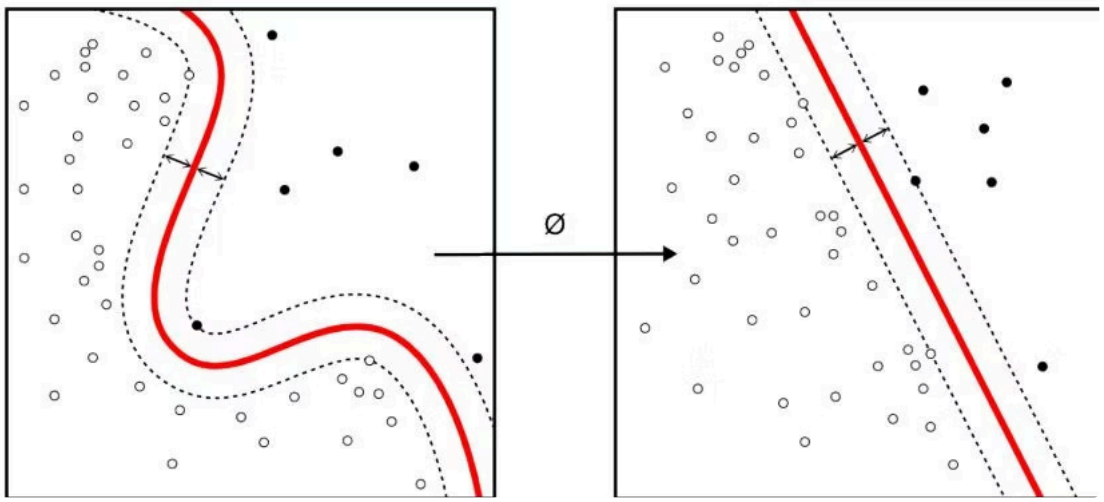
Logistic Regression Results

After running the logistic regression and toggling different factors with various transformations, the **highest accuracy score achieved was 55.94%**, a considerable improvement from the random baseline. This indicates that significant predictive linear relationships exist between Bitcoin and the various market indices and on-chain metrics.

Additionally, the logistic regression F1 score was 0.6536, indicating meaningful ability to identify directional trends beyond a random baseline.

2. Support Vector Machine

Moving on from logistic regression, another great supervised machine learning method for binary classification is the support vector machine. This algorithm attempts to find an optimal hyperplane in n-dimensional space (where n is the number of factors) that separates data points into two categories for classification. It does so by calculating the maximum margin between the hyperplane and closest data points on either side, called **support vectors**. While humans obviously cannot visualize n-dimensional space, support vector machines can use a **kernel function** to map the hyperplane to higher-dimensional space. This kernel may be linear or non-linear, as shown below.



All support vector machines, regardless of kernel function, aim to minimize the **regularized loss function**

$$\min_f C\sum_{i=1}^{n}\mathcal{L}(f(x_i), y_i) + \Omega(w) \quad (12)$$

which essentially finds the boundary $f(X)$ that classifies data as correctly as possible, while maximizing the boundary between support vectors. The hinge loss $\mathcal{L}$ penalizes points that are incorrectly categorized compared to actual binary value y_i, while $\Omega(w)$ keeps the decision boundary as wide as possible. The formulas for $\mathcal{L}$ and $\Omega(w)$ are less significant for the purposes of this explanation. More importantly for this study, parameter C determines the balance between these goals: larger C prioritizes fewer classification errors above possible overfitting, while smaller C prioritizes a simpler, wider boundary.

This section will test two types of support vector machines: one with **linear kernels**, and one with an **RBF (radial basis function) kernel**. While a linear support vector machine simply attempts to maximize the distance between support vectors, the RBF kernel measures similarity between data points, allowing for curved boundaries.

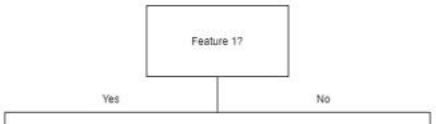
Support Vector Machine Results

Support vector machines were run for both linear and RBF kernels, with parameter C being set to 1, 5, and 10.

Interestingly enough, these results show the most effective SVM achieves **54.69% accuracy**, with a linear kernel that prioritizes establishing the simplest hyperplane (since $C = 1$). Again, this suggests that the relationships between the various features and the target variable are quite linear. The associated F1 score is 0.6588, showing a promising balance between precision and recall.

3. Tree-Based Algorithms

The **decision tree** is the core tree-based model, which uses a series of nodes that represent partitions in which each data point could fall. Through a series of yes/no questions (usually inequalities for quantitative data), an optimal flow is created to maximize classification accuracy.



Conclusion and Future Recommendations

Findings and Commentary

This original research paper represents the first ever machine learning study conducted on Bitcoin pricing that combines *both* on-chain metrics and market conditions for predictive analysis.

Often, professional traders aim for directional accuracy of 55%-60% in their trades, and considering the volatility of cryptocurrencies, this is an ambitious range. The **logistic regression model achieved greatest accuracy across the testing period at 55.94%**, quite an impressive marker for a linear-based model. Further research would be required to engineer a profitable trading strategy from these baseline predictions. While more complicated models like long short term memory produced decreased accuracy, this can likely be attributed to the smaller size of the dataset. However, running all these machine learning models in parallel acts as a strong blueprint to understand the preliminary nature of a volatile asset, and its degree of linearity (or non-linearity).

This analysis seems to show that **Bitcoin is primarily acting like a stock index, such as the SPX, Nasdaq or Dow; and secondarily, a metal commodity**. The classification of Bitcoin as most like a stock index makes this a more transparent product for mainstream investors and portfolio managers, who previously avoided the asset class due to ambiguity. While a mainstream investor may still want to avoid spot bitcoin to avoid uncertainty and cumbersome custody on physical asset custody; there are traditional financial instruments, such as Bitcoin Futures (launched in 2017) and Bitcoin Future ETFs (launched in 2021) as well as options on these products that eliminate the issue of direct spot investment. A portfolio manager can allocate to Bitcoin Futures and/or Bitcoin ETFs much like investing in the ES futures and/or SPY ETF. While more on-chain data would need to be pulled and implemented into a logistic regression and/or long short term memory recurrent neural networks in future, our findings indicate diminished importance compared to Bitcoin's relationship with established markets. We would specifically like to highlight Bitcoin's notable relationship with copper, a definite point of further investigation.

Future Research

Most obviously, expanding the dataset from this study (specifically in regards to on-chain data) while implementing the same methodologies could produce further promising results. With a larger dataset and more factors, both Boruta feature selection and long short term memory (LSTM) will be augmented, increasing predictive power. Then, most significantly, **trading strategies** can be developed based on the signals from this machine learning study, which can ideally provide an edge as Bitcoin markets become further optimized. Backtesting can be done on Bitcoin historical data, with the end goal being implementation of various ML-influenced trading strategies.

Another interesting avenue to explore would be intraday Bitcoin predictive pricing, which would involve looking at fluctuations in extremely small periods of time. This could lead to high frequency trading in cryptocurrency markets, which is still a novel and extremely risky undertaking. However, an increasing number of firms are investigating the integration of machine learning into short-term prediction and trading strategies.

Finally, another area of influence in Bitcoin pricing could be electricity demand and generation, which relates to ASIC usage, pricing, and other elements of Bitcoin's sophisticated mining processes. Through analysis of the infrastructure, better understanding of pivotal factors to enhance machine learning models could be discovered.

References

1. Oluwadamilare Omole, David Enke. *Deep Learning for Bitcoin Price Direction Prediction: Models and Trading Strategies Empirically Compared*. 2024. https://doi.org/10.1186/s40854-024-00643-1
2. Azaz Hassan Khan et al. *A performance comparison of machine learning models for stock market prediction with novel investment strategy*. 2023. https://doi.org/10.1371/journal.pone.0286362
3. Franco Valencia et al. *Price Movement Prediction of Cryptocurrencies Using Sentiment Analysis and Machine Learning*. 2019. https://doi.org/10.3390/e21060589
4. scikit-learn. *User Guide*. 2025. https://scikit-learn.org/stable/user_guide.html
5. GeeksforGeeks. *Machine Learning*. 2025. https://www.geeksforgeeks.org/machine-learning/machine-learning/