INVESTMENT ANALYSTS JOURNAL
https://doi.org/10.1080/10293523.2026.2616575

Bitcoin forecasting with machine learning and on-chain information

Hoseung Kang[a], Yeonchan Kang [b], Doojin Ryu [b] and Robert I. Webb [c]

[a]School of Mechanical Engineering, Sungkyunkwan University, Suwon-si, Gyeonggi-do, Republic of Korea; [b]Department of Economics, Sungkyunkwan University, Seoul, Republic of Korea; [c]McIntire School of Commerce, University of Virginia, Charlottesville, USA

ABSTRACT

This study evaluates machine learning models for forecasting daily Bitcoin returns using on-chain, macroeconomic, and market variables from January 2017 to December 2023. We implement a rolling-window framework with window lengths ranging from 365 to 730 days and compare several machine learning models against an autoregressive benchmark. Random Forest and Support Vector Machine achieve the lowest forecasting errors consistently across volatility regimes. Feature importance analysis using permutation importance and SHAP decomposition reveals that on-chain variables account for approximately 50 per cent of total forecasting contribution, with transaction fees and mining-related metrics ranking among the top important variables. Traditional market indicators such as VIX show limited relevance for Bitcoin return forecasting. These findings highlight the distinct informational value of blockchain-native variables for cryptocurrency forecasting.

ARTICLE HISTORY
Received 17 October 2025
Accepted 10 January 2026

KEYWORDS
Bitcoin; cryptocurrency; machine learning; on-chain data; forecasting

JEL CLASSIFICATION
C53; G14; G17

1. Introduction

Conventional financial theories often fail to explain Bitcoin returns (Biais et al., 2023; Kang, Ryu, & Webb, 2025a; Makarov & Schoar, 2020; Son et al., 2023; Vidal-Tomás & Aste, 2025). The problem is not just high volatility. Unlike traditional assets, Bitcoin has no identifiable fundamentals. It lacks earnings, dividends, and discounted cash flows to anchor valuations. Its returns show strong nonlinear dependencies (Liu & Tsyvinski, 2021), which makes linear econometric models less effective. Previous studies attempting to capture these dynamics report mixed results, with trading volume predicting returns in normal markets but failing to forecast volatility (Balcilar et al., 2017). Bitcoin also operates in a relatively isolated network (Ji et al., 2018). It responds to different information sets compared to stocks or bonds. These differences suggest that price formation mechanisms are fundamental. If return dynamics differ, we need modeling approaches that handle nonlinearity and regime shifts. We cannot rely on rigid functional forms. Machine learning algorithms offer exactly this flexibility (Gu et al., 2020; Kim, Park, & Ryu, 2025; Kim, Ryu, & Webb, 2025; Page et al., 2024). Their

CONTACT Doojin Ryu sharpjin@skku.edu

Investment Analysts Journal is co-published by NISC Pty (Ltd) and Informa Limited (trading as Taylor & Francis Group)

expanding use in finance reflects practical advantages in dealing with complex assets (Chun et al., 2025; Kang et al., 2025b). Similar benefits appear in high-dimensional prediction problems. LASSO-based methods successfully isolate significant pricing factors from large variable sets (Bang et al., 2024; Bang & Ryu, 2024). This shows that algorithmic approaches can extract signals that traditional methods miss.

Researchers increasingly combine machine learning with blockchain data to forecast cryptocurrency (Merediz-Solà & Bariviera, 2019). Early studies demonstrate the efficacy of these algorithmic approaches. Akyildirim et al. (2021) report that support vector machines predict cryptocurrency direction with 55 to 65 per cent accuracy. They outperform other methods across multiple digital assets. Sebastião and Godinho (2021) integrate network activity variables. Their models perform well even when the market direction reverses between validation and testing. Recent studies add diverse datasets to these frameworks. Li and Du (2023) analyze transaction patterns for daily price prediction. Wei et al. (2023) construct a framework based on sentiment and volatility leverage. Chi and Hao (2024) examine blockchain activity to forecast both returns and volatility. Omole and Enke (2025) utilize deep learning to merge on-chain metrics with technical indicators. This data is useful because it captures real-time supply, network incentives, and miner behavior. Such information is not available for traditional assets (Kim et al., 2023; Lee & Ryu, 2025a; Siu, 2025). Recent literature supports this view: Kim et al. (2024) document that hash rate fluctuations directly influence network security, with declining computational power exposing the blockchain to elevated attack risks and subsequently affecting investor confidence in the asset. Lee and Ryu (2025b) further demonstrate that miner behavior responds asymmetrically to price movements, particularly during bull markets when fear of missing out intensifies, and miners increase their hash allocation in anticipation of higher future rewards. These feedback loops between on-chain dynamics and market prices suggest that blockchain metrics capture supply-side forces absent in conventional asset classes.

Current literature still faces limitations regarding conventional variable sets. Bouri and Gupta (2021) find that internet search-based uncertainty predicts Bitcoin returns better than newspaper-based policy uncertainty. This implies that traditional media sentiment often misses relevant information channels. Wang et al. (2023) report that macroeconomic factors dominate technical ones. The problem is that previous studies mostly focus on volatility instead of returns. They also exclude on-chain variables. Other research points to different transmission channels. For example, Pham et al. (2025) identify investor contagion as the main driver during exchange cyberattacks. Yet, it remains an open question whether blockchain metrics actually improve return predictions compared to standard variables. We also see a gap in how these models work in practice. Current frameworks rarely handle data alignment or rolling re-estimation in a systematic way. Most researchers focus only on accuracy. They tend to ignore deployment issues like real-time data constraints or model stability across volatility regimes. We aim to fix this disconnect between theory and practice.

Our results provide new insights into model performance. In our tests, Random Forest (RF) and Support Vector Machine (SVM) deliver the strongest performance. They keep forecasting errors low, regardless of the volatility regime or the length of the training window. In contrast, Extreme Gradient Boosting (XGB) and Multi-Layer Perceptron (MLP) underperform. Both models return negative out-of-sample R^2 values, which

points to severe overfitting during training. On-chain variables, however, stand out. In our PFI (Permutation Feature Importance) and SHAP (SHapley Additive exPlanations) tests, transaction fees and mining metrics take the top spots. These metrics proxy for supply-side forces, such as production costs and network security. Rather than merely duplicating macro indicators, they provide complementary signals. In total, blockchain-native inputs generate roughly half of the forecasting power. This share exceeds the combined impact of standard market and macro variables. The data indicates that blockchain sources bring unique information for return prediction. Crucially, these gains persist in both calm and volatile markets. Performance remains steady regardless of volatility levels. Practitioners can therefore maintain a unified model architecture without switching regimes. In contrast, news indicators fail to predict returns accurately. This suggests that mainstream media coverage does not capture the real information flow in crypto markets.

We contribute to the literature in three ways. First, we benchmark on-chain variables against macro and market data using PFI and SHAP. Unlike prior work that examines isolated variables, we measure relative importance inside a unified framework. Our results indicate that mining variables, in particular, hold predictive content missing from financial predictors. Second, we test stability across volatility regimes, defined by the 25th and 75th percentiles. The RF and SVM models perform robustly in all conditions. This simplifies operations. Traders can maintain the same model and just adjust position sizes, rather than switching models. Third, we track the incremental gains of each variable. We use distributional metrics like average improvements and positive shares. This separates reliable predictors from sporadic ones. We also define clear rules for windowing and leakage control. This allows investment teams to run daily re-estimations with confidence, focusing on the stable predictive power of on-chain data.

The remainder of this paper proceeds as follows. Section 2 describes the data sources and methodology. Section 3 presents the empirical results by comparing model performance across volatility regimes and analyzing the incremental contribution of each variable. Finally, Section 4 concludes the paper.

2. Data and methodology

Our study covers the period from January 2017 to December 2023. We focus on forecasting Bitcoin log returns one day in advance. Figure 1 shows the daily return trends during this time.

We select variables based on clear daily timestamps and reliable release schedules. We avoid derived metrics like SOPR or MVRV. Exchange-specific data, such as order book liquidity, is also dropped to prevent look-ahead bias and timing issues. We organize the predictors into three panels. CoinGecko provides on-chain metrics for Panel A. This includes fees, volumes, and mining data such as hash rate and difficulty. For macroeconomic conditions (Panel B), we source treasury yields and spreads from FRED. Market measures in Panel C, including the S&P 500 and the VIX, come from Yahoo Finance.

All variables have a UTC close timestamp. This ensures no future data leaks into the model. Data preparation involves two steps. First, we winsorize data at the 1^{st} and 99^{th} percentiles. This handles outliers without changing the distribution shape. Second, we apply Min-Max scaling. This brings all values into a [–1, 1] range for better training

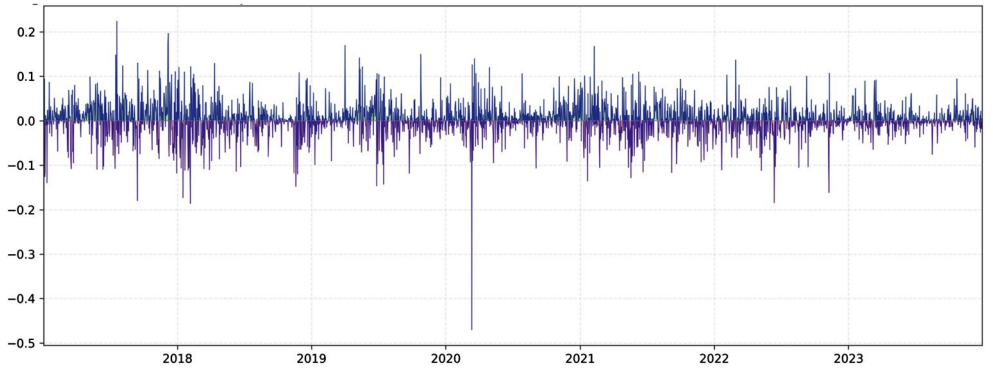

Figure 1. Bitcoin daily returns time series.

Notes: This figure reports the Bitcoin daily log return.

stability. Table 1 lists the descriptive statistics, including mean, standard deviation, and Augmented Dickey-Fuller (ADF) test results.

We compare four machine learning models, RF, XGB, SVM, and MLP, against an Auto Regressive (AR) benchmark. The AR model estimates:

$$y_t = \alpha + \sum_{i=1}^{p} \beta_i y_{t-i} + \epsilon_t \tag{1}$$

where α denotes the intercept, β_i represents the autoregressive coefficient at lag i, p is the lag order selected by the Akaike Information Criterion (AIC), and ϵ_t is the error term on the full T-day window. The AR model provides a baseline capturing linear mean

Table 1. Descriptive statistics.

	Description	Mean	Std	Skew	Kurt	ADF
Panel A. On-chain variables						
Block_Count	Daily block count	147.79	16.21	−0.345	4.12	−6.913***
Tx_Fee	Average transaction fee	0.008	1.549	1.138	32.77	−11.39***
Difficulty	Mining difficulty	0.002	0.016	1.207	48.85	−8.508***
Block_Size	Average block size	0.009	0.137	1.175	7.927	−6.893***
Miner_Rev	Miner revenue	0.009	0.129	0.772	5.566	−32.46***
Tx_Volume	Transaction volume	0.066	0.423	2.734	19.51	−7.427***
Hash_Rate	Hash rate	0.010	0.126	0.722	5.215	−19.94***
Panel B. Macroeconomic variables						
S&P_Ret	S&P 500 return	0.000	0.010	−0.612	25.70	−10.48***
Risk_Free	3-month US Treasury change	0.002	0.029	1.949	28.87	−7.038***
OAS	Option Adjusted Spread	4.137	0.974	2.619	12.68	−3.728***
10Year	10-year US treasury yield	0.001	0.045	−0.161	7.94	−37.81***
2Year	2-year US treasury yield	0.001	0.045	−1.137	22.26	−23.65***
Term_Spread	Term spread(10Y-2Y)	−0.001	0.031	0.956	19.08	−9.397***
Panel C. Market variables						
VIX	Volatility index	−0.001	1.668	3.312	53.99	−12.82***
USD_Index	US Dollar index	0.000	0.003	0.153	8.080	−11.27***
Exp_Infla	Expected inflation	0.000	0.027	−0.661	18.01	−14.99***
News_Sent	US news sentiment index	−0.050	0.196	−0.826	4.031	−3.27**

Notes: Panel A presents *On-chain variables* capturing blockchain-specific dynamics; Panel B presents *Macroeconomic variables* reflecting broader financial conditions; Panel C presents *Market variables* capturing uncertainty and investor expectations. *Mean*, *Std*, *Skew*, *Kurt*, and *ADF* denote the sample mean, standard deviation, skewness, kurtosis, and Augmented Dickey-Fuller test statistic, respectively. *** and ** denote significance at the 1% and 5% levels, respectively.

reversion typical of time-series methods. RF builds an ensemble of decision trees trained on bootstrapped samples, selecting random feature subsets at each split and averaging predictions to reduce variance. We tune the number of trees (*n_estimators*) and maximum tree depth (*max_depth*) via grid search. XGB sequentially adds trees that minimize a regularized objective, with each tree correcting errors from previous iterations. We tune the number of boosting rounds (*n_estimators*), learning rate (*learning_rate*), maximum tree depth (*max_depth*), and subsample ratio (*subsample*). Both methods accommodate nonlinear interactions without manual feature construction.

SVM projects features into higher dimensions through a kernel function, then identifies the maximum-margin hyperplane separating prediction errors. We tune the kernel type (linear, radial basis function), regularization parameter C, and kernel coefficient γ. MLP learns hierarchical representations through multiple layers of nonlinear transformations:

$$\hat{y} = f_L(\boldsymbol{W}_L \cdot f_{L-1}(\cdots f_1(\boldsymbol{W}_1\boldsymbol{x} + \boldsymbol{b}_1)\cdots) + \boldsymbol{b}_L), \tag{2}$$

where $\boldsymbol{x}$ denotes the input feature vector, $\boldsymbol{W}_l$ and $\boldsymbol{b}_l$ represent the weight matrix and bias vector at layer l, $f_l(\cdot)$ is the activation function at layer l ($l = 1, 2, 3, \ldots, L$), L is the total number of layers, and $\hat{y}$ is the predicted output. We tune the number and size of hidden layers (*hidden_layer_sizes*), L2 regularization penalty, and maximum training iterations (*max_iter*) with early stopping to prevent overfitting. Table 2 reports the hyperparameter search grids for each machine learning model.

Our forecasting procedure follows a rolling-window evaluation framework to simulate real-time one-day-ahead prediction. Let y_t denote the daily log return on Bitcoin at time t, computed as $\ln(P_t/P_{t-1})$, where P_t is the closing price. All features are time-stamped at market close (UTC) on day t-1 to construct the forecast $\widehat{y_t}$, ensuring no future information enters the training process. For each training window length $T \in \{365, 450, 540, 630, 730\}$ days, we form a rolling window $[t-T+1, t]$. The extended window lengths, ranging from approximately one year to two years, provide sufficient observations for stable parameter estimation in machine learning models while capturing evolving market dynamics.

Machine learning models use a time-ordered 70/30 split within the window exclusively for hyperparameter tuning; the temporal ordering is preserved, and no shuffling is applied. This split serves to regularize hyperparameter search and prevent overfitting

Table 2. Hyperparameters search grid.

Model	Hyperparameter	Search Range
RF	n_estimators	{100, 200, 300}
	max_depth	{3, 5, 7, None}
XGB	n_estimators	{100, 200, 300}
	max_depth	{3, 5, 7}
SVM	kernel	{linear, rbf}
	C	{0.1, 1.0, 10.0}
	gamma	{scale, auto}
MLP	hidden_layer_sizes	{(64), (64, 32), (128, 64)}
	alpha	{0.0001, 0.001, 0.01}

Notes: Grid search is performed on a 30% validation fold within each rolling window. RF, XGB, SVM, and MLP denote the Random Forest, Extreme Gradient Boosting, Support Vector Machine, and Multi-Layer Perceptron, respectively.

during the tuning phase. After selecting optimal hyperparameters via grid search on the validation fold, we refit each model on the entire T-day window and generate the one-step-ahead forecast $\widehat{y_{t+1}}$. The window then rolls forward one day, and the process repeats. This full-window refit preserves sample efficiency while controlling look-ahead bias. The AR benchmark, requiring no hyperparameter tuning, estimates on the full T-day window and directly produces $\widehat{y_{t+1}}$. Figure 2 illustrates the rolling-window method for one-step-ahead forecasting.

Model performance is evaluated using Root Mean Squared Error (RMSE) and out-of-sample R^2 (R^2_{oos}). RMSE measures prediction accuracy by penalizing larger errors more heavily:

$$RMSE = \sqrt{\frac{\left(\Sigma_{t=1}^{N}\left(y_t - \widehat{y}_t\right)^2\right)}{N}} \tag{3}$$

To assess the predictive contribution of exogenous data, we use the relative performance gain using the R^2_{oos} measure (Campbell & Thompson, 2008):

$$R^2_{oos} = 1 - \frac{RMSE^2_{\text{model}}}{RMSE^2_{\text{bench}}}, \tag{4}$$

where $\text{RMSE}_{\text{model}}$ ($\text{RMSE}_{\text{bench}}$) is the *RMSE* after (before) including the variables. We also compute directional accuracy as the proportion of correctly predicted return signs, $\Pr[\text{sign}(\widehat{y_t}) = \text{sign}(y_t)]$, to assess qualitative forecast performance. To evaluate model stability across market conditions, we classify volatility regimes based on realized volatility $RV_t = |y_t|$. Periods with RV_t above (below) the full-sample 75th (25th) percentile are classified as high-volatility (low-volatility) regimes:

$$\text{High: } RV_t > Q_{0.75}; \text{ Low: } RV_t < Q_{0.25}, \tag{5}$$

where $Q_{0.75}$ and $Q_{0.25}$ denote the 75th and 25th percentiles, respectively. These fixed thresholds allow us to assess whether predictive gains remain consistent under different market conditions.

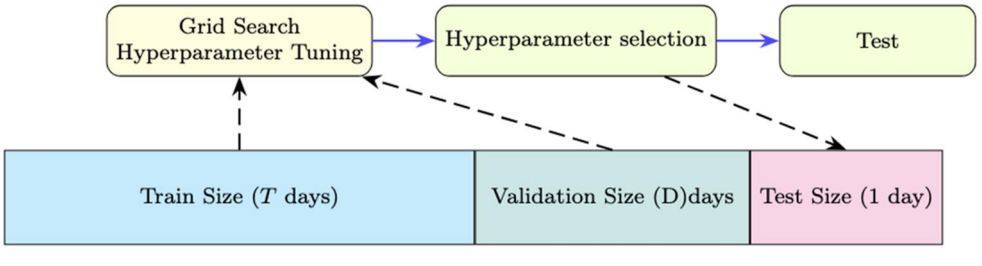

Figure 2. Rolling window method.

Notes: This figure shows the rolling-window evaluation process. Within each T-day window, machine learning models use a time-ordered 70/30 split exclusively for hyperparameter tuning to prevent overfitting during parameter selection. The final model is then refitted on the full T days and issues a single one-day-ahead forecast $\hat{y}_{t+1}$. The window subsequently rolls forward by one day, and the process repeats. The AR benchmark estimates on the full window without hyperparameter tuning and directly predicts y_{t+1}. This procedure ensures strict temporal separation and mirrors real-time deployment in investment practice.

We measure variable importance using Permutation Feature Importance (PFI) and SHAP values (Lundberg and Lee, 2017). PFI measures error growth when we shuffle a specific variable. This process breaks the link to the target but preserves the variable's distribution. We complement this with SHAP values. These decompose forecasts into additive parts, offering both global rankings and observation-level details. This dual approach helps us identify which predictors provide stable signals across different time windows.

3. Empirical analyses

Table 3 compares the models across five training windows. Panel A covers the full sample. Panel B looks at high volatility, while Panel C focuses on low volatility. RF and SVM perform best across all specifications. In Panel A, RF posts an RMSE between 0.033 and 0.035. SVM matches this range. The AR model produces worse results, with an RMSE from 0.042 to 0.046. Both RF and SVM generate positive out-of-sample R^2 in most cases. This confirms they outperform the benchmark. XGB falls in the middle. It achieves an RMSE of 0.037 to 0.038. This is lower than AR but higher than RF and SVM. However, XGB posts negative R^2 values ranging from –0.201 to –0.284. Its forecasts underperform relative to the AR baseline. MLP exhibits significant underperformance. Its RMSE ranges from 0.083 to 0.122, and R^2 drops as low as –10.30.

These patterns remain consistent across market conditions. Panel B isolates high-volatility days (above the 75th percentile). RF and SVM maintain their edge here. Panel C covers quiet periods (below the 25th percentile). RMSE is lower here simply because the returns themselves vary less. But the rankings do not change. RF and SVM maintain an RMSE around 0.024. The AR model sits between 0.024 and 0.025. The takeaway is simple. RF and SVM stay stable whether the market is calm or chaotic.

Table 3. Model performance measured by RMSE and R^2_{oos}.

	AR (baseline)		RF		XGB		SVM		MLP	
Window	RMSE	R^2_{oos}	RMSE	R^2_{oos}	RMSE	R^2_{oos}	RMSE	R^2_{oos}	RMSE	R^2_{oos}
Panel A: Overall periods										
365	0.046	–	0.035	0.006	0.038	–0.201	0.035	–0.010	0.119	–8.853
450	0.044	–	0.034	0.003	0.038	–0.284	0.033	0.017	0.122	–10.30
540	0.043	–	0.033	–0.002	0.038	–0.262	0.033	0.014	0.094	–5.716
630	0.042	–	0.034	0.002	0.038	–0.256	0.033	0.015	0.095	–5.811
730	0.042	–	0.033	0.004	0.037	–0.232	0.033	0.011	0.083	–4.368
Panel B: High-volatility periods (above *75th* percentile)										
365	0.060	–	0.043	0.008	0.047	–0.169	0.043	–0.006	0.137	–7.336
450	0.058	–	0.041	0.001	0.046	–0.232	0.041	0.021	0.138	–8.158
540	0.055	–	0.041	–0.005	0.046	–0.230	0.041	0.018	0.109	–4.712
630	0.055	–	0.041	–0.001	0.046	–0.225	0.041	0.016	0.108	–4.590
730	0.055	–	0.041	0.004	0.045	–0.223	0.041	0.011	0.097	–3.527
Panel C: Low-volatility periods (below *25th* percentile)										
365	0.025	–	0.024	0.006	0.028	–0.297	0.024	–0.017	0.098	–14.84
450	0.024	–	0.024	0.013	0.029	–0.426	0.024	0.014	0.105	–17.79
540	0.024	–	0.024	0.006	0.028	–0.378	0.024	0.007	0.076	–9.19
630	0.024	–	0.024	0.007	0.027	–0.342	0.024	0.011	0.080	–10.25
730	0.024	–	0.024	–0.001	0.027	–0.281	0.024	0.007	0.069	–7.25

Notes: *Panel A*, *Panel B*, and *Panel C* present *RMSE* and R^2_{oos} for the overall sample period, high-volatility periods, and low-volatility periods, respectively. Window denotes the training window length in days. Lower RMSE and higher R^2_{oos} indicate better predictive accuracy.

Why do XGB and MLP fail? The answer lies in the bias-variance tradeoff. These algorithms aim for low training error by using flexible structures. But in finance, this flexibility backfires. The signal is weak compared to the noise. XGB adds trees one by one to fix previous errors. If the data is noisy, this process ends up learning the noise instead of the signal. MLP has the same problem. Its huge parameter space leads it to memorize data rather than generalize, especially with small samples. RF and SVM handle this better. RF averages predictions from many trees, which smooths out the noise. SVM uses margins to ignore small errors. These structural differences explain why RF and SVM work while the others break down.

Table 4 details the PFI and SHAP values for our two best models, RF and SVM. In the RF model, PFI analysis shows that on-chain data dominates. *Tx_Fee* takes the first spot, followed by *USD_Index* and *Miner_Rev*. If we count the top ten variables, five come from the blockchain (*Tx_Fee*, *Miner_Rev*, *Block_Count*, *Hash_Rate*, *Difficulty*). This clustering confirms that on-chain data offers unique signals missing from traditional indicators. SVM, however, relies on different inputs. The prominence of *Exp_Infla* puts it in first place, followed by *Block_Size*, with *Block_Count* in third. This difference matters. It shows that different algorithms extract signal from different combinations of variables. This supports the use of multiple models instead of relying on just one.

Figure 3 helps clarify these PFI rankings. In RF, *Tx_Fee* is the top predictor. This makes sense. Fees go up when the network is busy. They act as a proxy for immediate market demand. SVM is different. It relies heavily on *Exp_Infla*. This suggests it focuses more on macro conditions than RF does. *Block_Size* also scores high. Since it tracks the size of new blocks, it serves as a gauge for network throughput.

The picture changes when we look at SHAP values in Figure 4. For RF, the ranking shifts. *News_Sent* jumps to the top, *OAS* takes second, and *Tx_Fee* falls to third. The reason is simple. The two methods measure importance in different ways. PFI tests how much error grows when we shuffle a variable. It isolates the variable's individual

Table 4. PFI and SHAP values by model.

		RF		SVM	
Variable	Category	PFI	SHAP	PFI	SHAP
Tx_Fee	On-chain	1	3	6	12
USD_Index	Market	2	6	7	9
Miner_Rev	On-chain	3	7	10	13
OAS	Macro	4	2	15	5
News_Sent	Market	5	1	14	7
Block_Count	On-chain	6	5	3	14
Term_Spread	Macro	7	4	17	16
S&P_Ret	Macro	8	8	4	6
Hash_Rate	On-chain	9	10	5	11
Difficulty	On-chain	10	14	11	17
Risk_Free	Macro	11	13	13	8
Block_Size	On-chain	12	12	2	1
VIX	Market	13	9	16	10
Tx_Volume	On-chain	14	11	9	2
Exp_Infla	Market	15	15	1	4
10Year	Macro	16	16	8	15
2Year	Macro	17	17	12	3

Notes: This table reports the variable importance ranking of RF and SVM with two methods: *PFI* and *SHAP*. *Category* indicates variable grouping: *On-chain* (blockchain metrics), *Macro* (macroeconomic indicators), and *Market* (market sentiment and uncertainty measures).

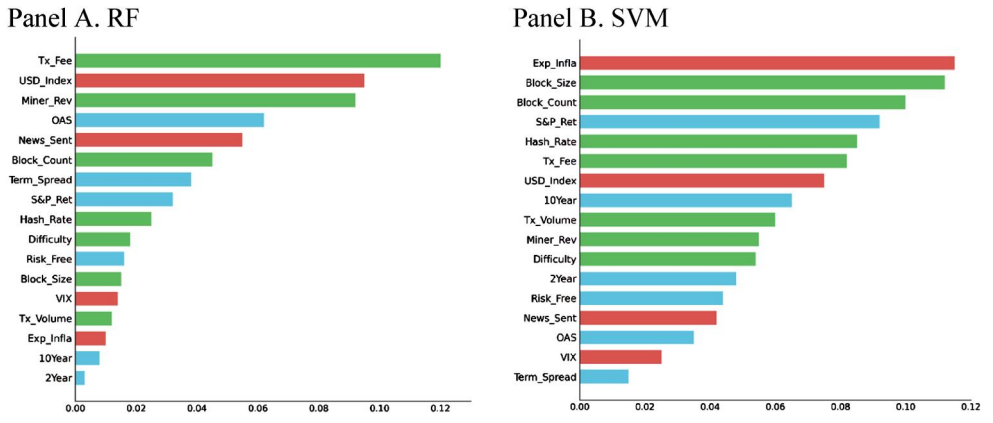

Figure 3. PFI by model.

Notes: This figure reports PFI for RF (Panel A) and SVM (Panel B). The *y*-axis represents the increase in prediction error when each variable is randomly permuted. Bars are colored by variable category: *On-chain* (green), *Macro* (blue), and *Market* (red).

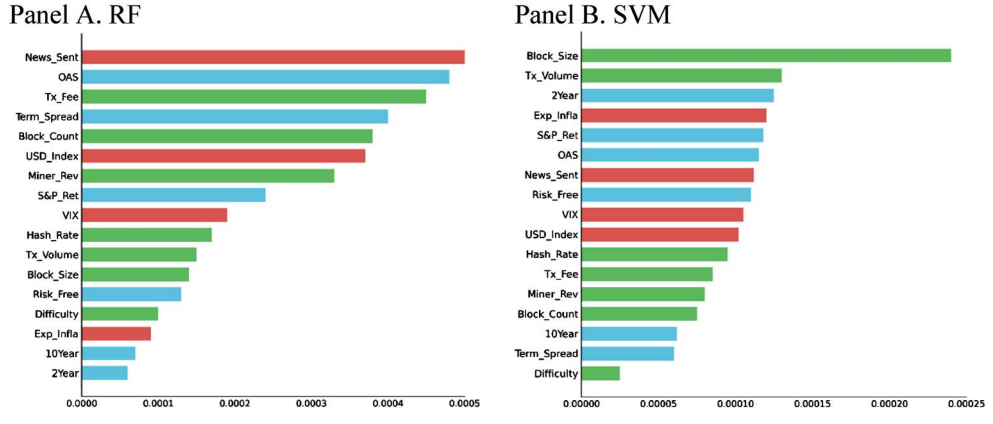

Figure 4. SHAP importance by model.

Notes: This figure reports mean absolute SHAP values for RF (Panel A) and SVM (Panel B). The *y*-axis represents the average magnitude of each variable's contribution to individual forecasting. Bars are colored by variable category: *On-chain* (green), *Macro* (blue), and *Market* (red). Higher values indicate greater contribution to forecasting performance.

impact. SHAP breaks the forecast down into additive parts. The high SHAP score for *News_Sent* implies that sentiment matters, but mostly when it interacts with other variables. It does not have strong power on its own.

Figure 5 groups the scores by category. In the PFI test, on-chain variables do the heavy lifting. They make up about half of the total importance, 49.5% in RF and 51.1% in SVM. SHAP numbers are similar, though a bit lower (around 41%). Macro indicators cover about a quarter of the total. Market variables fill the gap. The results are consistent. They support our main point: blockchain data offers signals that standard financial metrics miss.

Blockchain economics explains why these variables work. Transaction fees balance the demand for block space against the limited supply. When users bid up fees, it signals high

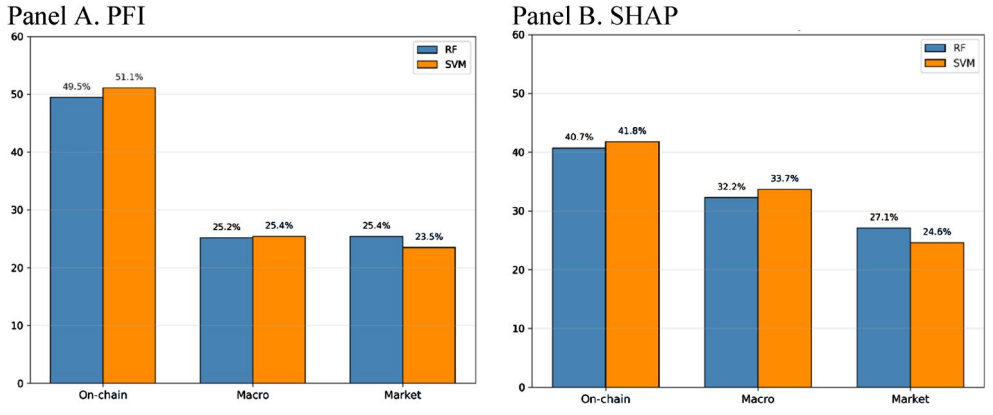

Figure 5. Feature importance by category.

Notes: This figure reports the aggregate contribution of each variable category based on PFI (Panel A) and SHAP (Panel B). The *x*-axis represents variable categories: *On-chain* (blockchain-specific metrics), *Macro* (macroeconomic indicators), and *Market* (market sentiment and uncertainty measures). The *y*-axis represents each category's percentage share of total importance within each model.

demand. This often happens right before price moves. *Miner_Rev* works differently. It measures how much miners earn. When revenue changes, it affects mining profitability. This shifts hash rates and network security. Forecasting power likely flows through this channel. Consider traditional assets. Stocks react to earnings. Bitcoin reacts to hash rates and difficulty. These are fundamentally different mechanics. They capture information that conventional indicators miss.

Traditional market indicators contribute surprisingly little. The VIX ranks 13th in RF and 16th in SVM under PFI, despite being a common uncertainty gauge. This makes sense given the market structure. Bitcoin operates in an ecosystem separate from traditional equities. It has distinct liquidity providers and volatility drivers. A spike in the VIX during equity stress does not necessarily affect Bitcoin. News sentiment gives mixed signals. It ranks 5th in RF but 14th in SVM under PFI. This divergence likely stems from our data sources. Crypto information propagates through Twitter, Reddit, and Discord, not Reuters. A measure built from legacy media might track the wrong channels. However, the fact that *News_Sent* performs better under SHAP suggests it still carries a signal through interactions. Future research should focus on social media sentiment, provided we can handle manipulation and spam.

4. Conclusion

We test machine learning models on Bitcoin returns from 2017 to 2023. We use rolling windows for the analysis. RF and SVM perform the best. They outperform the AR benchmarks in every volatility regime. In contrast, XGB and MLP fail. Their out-of-sample R^2 is negative. Why the difference? It comes down to design. RF uses bagging. SVM uses margin maximization. These features act as built-in regularization. They prevent the models from overfitting when noise overwhelms the signal. XGB and MLP prioritize reducing training error. In noisy financial data, this approach fails because the signal is weak.

The feature importance results are clear. On-chain variables provide about half the predictive power in both PFI and SHAP tests. In RF, transaction fees rank first. They capture real-time network demand. Miner revenue and block count also matter. They reflect the supply side of mining. These metrics offer data that traditional assets lack. The VIX performs poorly, ranking in the bottom half for both models. This confirms that Bitcoin moves independently of standard equity drivers. News sentiment is inconsistent. This is likely because our measure uses traditional media, not the crypto-native platforms where the real discussions happen.

Our study has limits. We investigate only the Bitcoin return dynamics. These results might not apply to coins with different consensus rules or market structures. Also, our sentiment measure misses Twitter, Reddit, and Telegram. That is where crypto news propagates. We also exclude transaction costs. Real trading profits would look different. Future research needs to test this on other major coins and add social media data. It should also simulate realistic trading with costs. Even with these gaps, we make two points. First, properly regularized machine learning can extract signals from noisy data. Second, blockchain information outperforms traditional indicators for Bitcoin. For portfolio managers, the lesson is simple. When forecasting returns, prioritize on-chain metrics over conventional financial variables.

Disclosure statement

No potential conflict of interest was reported by the authors.

Acknowledgement

We thank Christo Auret (Editor-in-Chief), Daniel Page (Editor), and the three anonymous referees for their helpful comments, which substantially improved the manuscript. This paper was supported by SKKU Academic Research Support Program (Samsung Research Fund), Sungkyunkwan University, 2024. This study is an outcome of the SKKU research project "Interdisciplinary research on financial markets using machine learning, psychology, and complex systems: International collaboration" (PI: Prof. Doojin Ryu).

ORCID iDs

Yeonchan Kang http://orcid.org/0009-0004-5988-299X
Doojin Ryu http://orcid.org/0000-0002-0059-4887
Robert I. Webb http://orcid.org/0000-0003-1714-5778

References

Akyildirim, E., Goncu, A., and Sensoy, A. (2021). Prediction of cryptocurrency returns using machine learning. *Annals of Operations Research*, *297*(1), 3–36. https://doi.org/10.1007/s10479-020-03575-y

Balcilar, M., Bouri, E., Gupta, R., and Roubaud, D. (2017). Can volume predict Bitcoin returns and volatility? A quantiles-based approach. *Economic Modelling*, *64*, 74–81. https://doi.org/10.1016/j.econmod.2017.03.019

Bang, J., and Ryu, D. (2024). ESG factors and the cross-section of expected stock returns: A LASSO-based approach. *Finance Research Letters*, *65*, 105482. https://doi.org/10.1016/j.frl.2024.105482

Bang, J., Kang, Y., and Ryu, D. (2024). Potential pricing factors in the Korean market. *Finance Research Letters*, *67*, 105946. https://doi.org/10.1016/j.frl.2024.105946

Biais, B., Bisière, C., Bouvard, M., Casamatta, C., and Menkveld, A.J. (2023). Equilibrium Bitcoin pricing. *Journal of Finance*, *78*(2), 557–598. https://doi.org/10.1111/jofi.13206

Bouri, E., and Gupta, R. (2021). Predicting Bitcoin returns: Comparing the roles of newspaper-and internet search-based measures of uncertainty. *Finance Research Letters*, *38*, 101398. https://doi.org/10.1016/j.frl.2019.101398

Campbell, J.Y., and Thompson, S.B. (2008). Predicting excess stock returns out of sample: Can anything beat the historical average? *Review of Financial Studies*, *21*(4), 1509–1531. https://doi.org/10.1093/rfs/hhm055

Chi, Y., and Hao, W. (2024). Return-forecasting and volatility-forecasting power of on-chain activities in the cryptocurrency market. *arXiv preprint arXiv:2411.06327*. https://doi.org/10.48550/arXiv.2411.06327

Chun, D., Cho, H., and Ryu, D. (2025). Volatility forecasting and volatility-timing strategies: A machine learning approach. *Research in International Business and Finance*, *75*, 102723. https://doi.org/10.1016/j.ribaf.2024.102723

Gu, S., Kelly, B., and Xiu, D. (2020). Empirical asset pricing via machine learning. *Review of Financial Studies*, *33*(5), 2223–2273. https://doi.org/10.1093/rfs/hhaa009

Ji, Q., Bouri, E., Gupta, R., and Roubaud, D. (2018). Network causality structures among Bitcoin and other financial assets: A directed acyclic graph approach. *Quarterly Review of Economics and Finance*, 70, 203-213. https://doi.org/10.1016/j.qref.2018.05.016

Kang, D., Ryu, D., and Webb, R.I. (2025a). Bitcoin as a financial asset: A survey. *Financial Innovation*, *11*, 101. https://doi.org/10.1186/s40854-025-00773-0

Kang, Y., Ryu, D., and Webb, R.I. (2025b). How well do machine learning models in finance work?. *Financial Innovation*, *11*(1), 128. https://doi.org/10.1186/s40854-025-00870-0

Kim, D., Ryu, D., and Webb, R.I. (2023). Determination of equilibrium transaction fees in the Bitcoin network: A rank-order contest. *International Review of Financial Analysis*, *86*, 102487. https://doi.org/10.1016/j.irfa.2023.102487

Kim, D., Ryu, D., and Webb, R.I. (2024). Does a higher hashrate strengthen Bitcoin network security? *Financial Innovation*, *10*(1), 79. https://doi.org/10.1186/s40854-023-00599-8

Kim, A., Ryu, D., and Webb, R.I. (2025). Forecasting oil futures markets using machine learning and seasonal trend decomposition. *Investment Analysts Journal*, *54*(2), 205–218. https://doi.org/10.1080/10293523.2024.2405294

Kim, T.-Y., Park, E., and Ryu, D. (2025). Determinants of housing rental prices in Seoul: Applying explainable AI. *Spatial Economic Analysis*, *20*(2), 312–332. https://doi.org/10.1080/17421772.2024.2418906

Li, X., and Du, L. (2023). Bitcoin daily price prediction through understanding blockchain transaction pattern with machine learning methods. *Journal of Combinatorial Optimization*, *45*(1), 4. https://doi.org/10.1007/s10878-022-00949-9

Lee, G., and Ryu, D. (2025a). Are base layer blockchains establishing a new sector? Evidence from a connectedness approach. *Research in International Business and Finance*, *73*(Part B), 102654. https://doi.org/10.1016/j.ribaf.2024.102654

Lee, G., and Ryu, D. (2025b). Fear of missing out and cryptocurrency miners: Evidence from Dogecoin and Litecoin. *Journal of Behavioral and Experimental Finance*, *46*, 101059. https://doi.org/10.1016/j.jbef.2025.101059

Liu, Y., and Tsyvinski, A. (2021). Risks and returns of cryptocurrency. *Review of Financial Studies*, *34*(6), 2689–2727. https://doi.org/10.1093/rfs/hhaa113

Lundberg, S.M., and Lee, S.I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, *30*. https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html

Makarov, I., and Schoar, A. (2020). Trading and arbitrage in cryptocurrency markets. *Journal of Financial Economics*, *135*(2), 293–319. https://doi.org/10.1016/j.jfineco.2019.07.001

Merediz-Solà, I., and Bariviera, A.F. (2019). A bibliometric analysis of bitcoin scientific production. *Research in International Business and Finance*, *50*, 294–305. https://doi.org/10.1016/j.ribaf.2019.06.008

Omole, O., and Enke, D. (2025). Using machine and deep learning models, on-chain data, and technical analysis for predicting bitcoin price direction and magnitude. *Engineering Applications of Artificial Intelligence*, *154*, 111086. https://doi.org/10.1016/j.engappai.2025.111086

Page, D., McClelland, D., and Auret, C. (2024). Machine learning style rotation–evidence from the Johannesburg Stock Exchange. *Cogent Economics & Finance*, *12*(1), 2402893. https://doi.org/10.1080/23322039.2024.2402893

Pham, D.T.N., Chung, C.Y., and Ryu, D. (2025). Contagion from crypto exchange hacks: Wealth effect or portfolio rebalancing? *Investment Analysts Journal*, *54*(3), 488-507. https://doi.org/10.1080/10293523.2025.2517973

Sebastião, H., and Godinho, P. (2021). Forecasting and trading cryptocurrencies with machine learning under changing market conditions. *Financial Innovation*, *7*(1), 3. https://doi.org/10.1186/s40854-020-00217-x

Siu, T.K. (2025). Market consistent valuation for Bitcoin options with long memory in conditional volatility and conditional non-normality. *Journal of Futures Markets, 45*(8), 917–945. https://doi.org/10.1002/fut.22597

Son, J., Ryu, D., and Webb, R.I. (2023). Central bank digital currency: Payment choices and commercial bank profitability. *International Review of Financial Analysis*, *90*, 102874. https://doi.org/10.1016/j.irfa.2023.102874

Vidal-Tomás, D., and Aste, T. (2025). Integration or separation? Examining the dynamic relationship between crypto and traditional finance. *Finance Research Letters*, *86*(Part G), 108927. https://doi.org/10.1016/j.frl.2025.108927

Wang, J., Ma, F., Bouri, E., and Guo, Y. (2023). Which factors drive Bitcoin volatility: Macroeconomic, technical, or both? *Journal of Forecasting*, *42*(4), 970–988. https://doi.org/10.1002/for.2930

Wei, M., Sermpinis, G., and Stasinakis, C. (2023). Forecasting and trading Bitcoin with machine learning techniques and a hybrid volatility/sentiment leverage. *Journal of Forecasting*, *42*(4), 852–871. https://doi.org/10.1002/for.2922