Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Signal Refinement in Seasonal Bitcoin Trading: Evidence from a Meta-Labeling Framework

Shantnu Sood[1]

soodshantnu2@gmail.com

[1]Himachal Pradesh University Business School, Himachal Pradesh University

Puneet Bhushan

[1]Himachal Pradesh University Business School, Himachal Pradesh University

Research Article

Keywords: Meta-labeling, Cryptocurrency markets, Financial machine learning, Trading strategy optimization, Market microstructure, Predictive modeling

Posted Date: November 25th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11658745/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Signal Refinement in Seasonal Bitcoin Trading: Evidence from a Meta-Labeling Framework

Shantnu Sood[1]*, Dr. Puneet Bhushan[1]

[1]Himachal Pradesh University Business School, Himachal Pradesh University, Shimla, Himachal Pradesh, India

*** Corresponding Author**

Shantnu Sood

Himachal Pradesh University Business School

Himachal Pradesh University, Shimla, Himachal Pradesh, India

Email: soodshantnu2@gmail.com

ORCID: 0009-0002-8789-8978

Co-author ORCID: 0009-0006-9114-8781

ABSTRACT:

The effectiveness of an overnight seasonal bitcoin trading strategy can be enhanced by a machine learning technique known as meta-labeling. Building on the nightly seasonal pattern found by Vojtko and Javorská (2023), a LightGBM model is used to refine the baseline signal and determine the likelihood that a trade would be profitable or not. Empirical findings demonstrate that using meta-labeling to filter trades improved the strategy's maximum drawdown and sharpe ratio, resulting in a superior risk-adjusted performance than the unfiltered strategy. The model is based on a high-dimensional dataset with 66 features, the majority of which are based on market microstructure variables and technical indicators (including, for example, momentum, volatility, liquidity, and trading intensity). To combat possible overfitting and redundancy, BorutaShap algorithm is implemented for feature selection so that only relevant predictors are used in the model. However, there are still several limitations. The analysis is limited to Bitcoin and a very specific intraday trading horizon,

which could limit the generalisability of the results. Furthermore, the lookback windows for technical indicators remain fixed instead of optimized and the backtesting framework assumes fixed transaction costs and doesn't take into account factors such as market impact or execution frictions. Future work may consider extending the analysis to more assets, calibrating indicator parameters, and analyzing the framework's performance under different market regimes.

Keywords: Meta-labeling; Cryptocurrency markets; Financial machine learning; Trading strategy optimization; Market microstructure; Predictive modeling

1. INTRODUCTION

Cryptocurrencies, in particular, bitcoin, have become an alternative asset class where investors can invest in or speculate on besides traditional asset classes like equity and debt markets. This growing interest is reflected in the sustained growth in trading activity in the cryptocurrency markets (Mourad & Goul, 2025). Bitcoin has provided exceptionally high returns for investors in the last few years particularly the post-covid period. But despite this it has been characterized by high levels of risk, including large drawdowns as well as high levels of volatility, relative to traditional financial markets. For instance, bitcoin experienced drawdowns exceeding 70% during major market corrections (Vojtko & Javorská, 2023).

The growing economic importance of cryptocurrency market has also motivated research into their return structure. Common risk factors in cryptocurrencies document the existence of systematic risk factors that explain cross-sectional variation in returns, suggesting that predictable patterns may exist in this market (Liu et al., 2019). Long et al., (2020) examine and discover that average past same weekday returns positively predict the cross sectional performance of cryptocurrencies.

In addition to cross-sectional factors, recent studies have identified time-based anomalies in Bitcoin returns. Extending the literature on time-based patterns, (Ben Omrane et al., 2023) document that intraday volatility in cryptocurrency markets exhibits day-of-the-week–specific seasonal behaviour. They show that these seasonal volatility patterns are significantly affected by macroeconomic news surprises and monetary policy announcements, particularly from the United States, highlighting the role of external information in shaping intraday cryptocurrency dynamics. Also Vojtko & Javorská (2023) show a statistically significant overnight pattern

and demonstrate that it can be translated into a profitable trading strategy by holding Bitcoin between 21:00 UTC and 23:00 UTC on a daily basis.

While such calendar-based strategies may generate positive returns on average, their performance can vary substantially across market conditions due to the evolving efficiency and high volatility of cryptocurrency markets. Consequently, static trading rules may lead to a large number of unprofitable trades and significant drawdowns. This limitation is addressed in this study by applying a meta-labeling framework, introduced by Prado (2018) also called Corrective AI Pik et al. (2025), to identify periods in which the underlying seasonal signal is more likely to be profitable. The model is trained to filter trading signals generated by the primary strategy, with the objective of improving trade selection and enhancing risk-adjusted performance.

On the basis of this the present study adopts the overnight seasonality strategy documented by Vojtko and Javorská (2023) as the baseline trading rule. A machine learning framework is then implemented to enhance this strategy's performance by estimating the conditional probability that a given trade will be profitable (meta-label). Specifically, a LightGBM model is trained using a set of technical indicators and market microstructure variables to identify favorable trading conditions and filter low-quality signals.

Therefore, this study contributes to the existing literature by being the first to examine the application of a meta-labeling framework to a calendar-based Bitcoin trading strategy. The proposed framework incorporates systematic hyperparameter tuning, interpretable feature selection using SHAP-based importance measures, and a set of technical and market microstructure indicators evaluated in a strictly out-of-sample setting. This design ensures both methodological rigor and economic relevance in assessing the incremental value of meta-labeling.

2. REVIEW OF LITERATURE

There is a bunch of evidence that suggests cryptocurrency markets exhibit short-horizon return predictability, challenging the efficient market hypothesis for cryptocurrency markets (Urquhart, 2016; Kristoufek, 2018). Caporale and Plastun (2019) document continuation patterns in intraday crypto prices and short-term momentum effects following large price movements. Similar intraday and calendar regularities have been widely documented in

traditional asset classes, that suggest a predictable return pattern may arise from time-varying liquidity conditions, institutional constraints , or persistent trading behaviour (Breedon and Ranaldo, 2012). More recently, Vojtko and Javorská (2023) have identified overnight seasonality in Bitcoin returns that is pronounced, showing that a strategy based on holding Bitcoin during specific hours generates superior risk-adjusted performance relative to a buy and hold benchmark strategy.

In addition to time-based anomalies, a parallel literature highlights the role of market frictions and liquidity dynamics in cryptocurrency price formation. Makarov and Schoar (2020) document substantial limits to arbitrage and persistent price dispersion across exchanges, indicating that cryptocurrency markets are segmented and subject to trading frictions. More recent studies emphasize the importance of order flow and trading activity in short-term return dynamics, consistent with market microstructure theory (Anastasopoulos et al., 2024; O'Hara et al., 2025). These findings suggest that short-horizon predictability may be driven not only by behavioural effects but also by time-varying liquidity and price impact.

The increasing availability of high-frequency data has also led to a growing application of machine learning methods to cryptocurrency forecasting. Machine learning models are particularly well suited to financial prediction because they can capture nonlinear and time-varying relationships that are difficult to model using traditional econometric approaches (Gu et al., 2018). Empirical evidence shows that momentum-based technical indicators and high-dimensional feature sets can improve out-of-sample predictive performance (Lapitskaya et al., 2024; Hasanli and Dursun, 2025). High-frequency evidence demonstrates that order flow and liquidity conditions contain significant predictive information for short-term price movements (Sirignano and Cont, 2019). While the aforementioned study is conducted using limit order book data from equity markets, the underlying microstructure mechanisms are general and can be approximated using volume-based and liquidity proxies in cryptocurrency market.

Despite these advances, the use of large feature sets introduces a substantial risk of overfitting and model instability. Many machine learning trading strategies exhibit strong in-sample performance but fail to generalize out of sample due to the high noise-to-signal ratio in financial data. Robust feature selection methods can mitigate this problem by removing redundant and uninformative predictors, thereby improving model stability and interpretability (Bailey et al., 2016; López de Prado, 2018). In addition, careful hyperparameter tuning plays a critical role in controlling model complexity and improving generalization performance. Systematic search

procedures such as cross-validation–based optimization and Bayesian tuning frameworks have been shown to provide more reliable model selection and reduce the risk of overfitting in high-dimensional financial prediction problems (Bergstra and Bengio, 2012; Probst et al., 2019).

Another important limitation of the existing literature is that most machine learning approaches attempt to directly predict asset returns. López de Prado (2018) argues that many such strategies fail because they attempt to jointly determine trade direction and position sizing within a single model, even though these represent distinct decision problems. To address this issue, López de Prado (2018) proposes a meta-labeling framework in which a secondary model evaluates whether a trading signal generated by a primary strategy should be executed. This approach focuses on improving trade selection by filtering low-probability signals and has been shown to enhance risk-adjusted performance in empirical applications (Pik et al., 2025).

While prior research separately documents (i) short-horizon anomalies in Bitcoin returns and (ii) the effectiveness of machine learning–based signal filtering, and their integration remains limited. In particular, the application of meta-labeling to calendar-based cryptocurrency trading strategies has received little attention. This study addresses this gap by applying a machine learning–based meta-labeling framework to the overnight seasonality strategy of Vojtko and Javorská (2023) and evaluating whether data-driven signal filtering based on technical and market-microstructure features improves its robustness and risk-adjusted performance.

3. RESEARCH METHODOLOGY

3.1. Data:

For this study, Bitcoin price data was pulled from Binance. It is one of the biggest cryptocurrency exchanges in terms of liquidity and trading volume which makes it a representative source of price discovery and market activity.

1-minute OHLCV (Open, High, Low, Close, Volume) data for the BTC/USDT pair has been fetched for the period 10.10.2019 to 30.01.2026. To align the data with the time horizon of the trading strategy, the data has been resampled to 2-hour intervals. For each interval, the opening price corresponds to the first transaction price, the closing price to the last transaction price, while the high and low represent the maximum and minimum prices within the interval. Trading volume is aggregated as the total traded quantity.

Raw OHLCV (Open, High, Low, Close, Volume) data were pre-processed and feature-engineered to construct the input variables used in the modeling framework. The preprocessing steps include:

1. Feature Construction, where technical and market microstructure indicators are created using raw price and volume data. A total of 66 features were constructed technical and market microstructure included
2. Data transformation, which involves normalizing variables, creating target variable for classification.

3.1.1. Technical Indicators:

The TA-Lib Python library is used to construct a variety of technical indicators that enhance the predictive value of the basic price and volume data. Several aspects of market behaviour - momentum, trend direction, volatility, and trading activity, are captured by these indicators.

There are a total of 54 technical indicators created which are based on moving averages, candlestick patters, cyclical indicators and volatility. With the use of these characteristics, the model can identify more intricate connections and market trends that are not apparent from raw pricing alone.

3.1.2. Market Microstructure Indicators:

In order to account for trading frictions and liquidity circumstances, the feature set additionally includes a number of market microstructure features that are obtained from price and volume data. These include price impact and illiquidity measurements such as the Amihud illiquidity ratio Amihud (2002), Kyle's lambda Kyle (1985), and Hasbrouck's price impact proxy Hasbrouck (1991), as well as spread estimators based on the Roll (1984) and Corwin-Schultz (2012) techniques. Additionally, trading activity and order-flow pressure are recorded using signed volume, dollar trading volume, and volume-based regime indicators.

Evidence that liquidity and order flow dynamics have a significant impact on cryptocurrency values is the driving force behind the inclusion of microstructure elements. While Anastasopoulos et al. (2024) and O'Hara et al. (2025) demonstrate that

order flow and trading activity are crucial to the development of short-term returns and price discovery, Makarov and Schoar (2020) highlight severe restrictions to arbitrage and persistent liquidity frictions across exchanges. These results imply that adding measures of price impact and liquidity could enhance machine learning models' capacity to capture short-horizon predictability. In total there are 12 market microstructure features.

A detailed description of all the 66 indicators is provided in the Appendix section.

3.2.Scaling and Normalization:

As tree methods are invariant to monotonic transformations, feature standardisation is not necessary for the tree-based prediction model (LightGBM) (Hastie et al., 2009; López de Prado, 2018).

3.3.Baseline Trading Strategy:

The empirical analysis is based on the overnight seasonality pattern documented by Vojtko and Javorská (2023). The baseline strategy involves taking a long position in Bitcoin at the open of 21:00 UTC and closing the position at the open of 23:00 UTC.

Formally, the strategy return for day t is defined as:

$$r_t = \frac{Open\ 23{:}00}{Open\ 21{:}00} - 1$$

This unconditional strategy serves as the benchmark for evaluating the effectiveness of machine learning–based trade filtering.

3.4.Meta-Labeling Framework:

In accordance with López de Prado (2018), the study employs a meta-labeling approach to enhance trade selection and risk-adjusted returns. The machine learning model is trained to decide whether to execute the trade signal produced by the baseline approach, rather than making direct predictions about returns.
The meta-label is constructed as a binary outcome indicating whether the baseline strategy return was profitable:

$$y_t = \begin{cases} 1, & if\ r_t > 0 \\ 0, & otherwise \end{cases}$$

As a result, given the knowledge at the moment of decision-making, the model learns the conditional probability that a transaction would result in a positive return. Only when the anticipated probability surpasses a predetermined level are trades conducted.

3.5.Feature Selection:

Feature selection is performed using the BorutaShap algorithm, which combines the Boruta feature selection method (Kursa & Rudnicki, 2010) with SHapley Additive exPlanations (SHAP)–based feature importance (Lundberg & Lee, 2017). Boruta is a wrapper based feature selection method designed to identify features that contain useful information for the prediction task.

The Boruta procedure operates by creating randomized versions of the original predictors, referred to as shadow features, through random permutation of their values. A tree-based model is then trained on the dataset containing both original and shadow features. The importance of each original feature is compared with the maximum importance achieved by the shadow features. Features that constantly exhibit higher importance than the shadow benchmark are considered informative and retained, while the remaining features are discarded as uninformative.

In the BorutaShap framework, feature importance is evaluated using SHAP values instead of traditional impurity-based measures such as mean decrease in impurity. SHAP values provide a theoretically consistent measure of the contribution of each feature to the model's prediction based on cooperative game theory. The Shapley value for feature $i$is defined as

$$\phi_i = \sum_{S \subseteq M \setminus \{i\}} \frac{|S|!(m-|S|-1)!}{m!} [f(S \cup \{i\}) - f(S)],$$

where Mdenotes the set of all features with cardinality m, S represents a subset of features that does not include feature i, and $f(S)$is the model prediction obtained using the feature subset S. The value ϕ_i measures the average marginal contribution of feature i across all possible feature combinations.

For tree-based models such as LightGBM, SHAP values are computed efficiently using the TreeSHAP algorithm (Lundberg et al., 2020). By combining the Boruta selection mechanism with SHAP-based importance, the BorutaShap approach enables the identification of the most informative predictors while removing redundant or noisy variables.

3.6.Model Building:

LightGBM, proposed by Ke et al. (2017), has been widely used due to its efficiency in handling large-scale data, which makes it suitable for use in classification, prediction, and regression tasks. It is an ensemble learning technique that is based on decision trees and contains two key innovations: Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB).

The objective of LightGBM is to estimate a function $f^*(x)$that achieves the minimum expected loss on the training dataset $X = \{(x_i, y_i)\}_{\{i=1\}}^{n}$:

$$f = argminE_{X,y}\big[L\big(y, f(x)\big)\big]$$

To achieve this, LightGBM integrates T regression trees in the form of:

$$f_T(X) = \sum_{t=1}^{T} f_t(X)$$

Each regression tree can be expressed as $w_q(x)$, where $q \in \{1,2, \dots, J\}$denotes the set of decision rules defining the tree structure, and $w$represents the vector of weights associated with the leaf nodes. LightGBM adopts an additive boosting approach, in which the model is refined iteratively by reducing the loss function at each step:

$$t = \sum_{i=1}^{n} L(y_i\,, F_{t-1}(x_i) + f_t(x_i))$$

LightGBM employs Newton's method to approximate the objective function efficiently. By simplifying the expression and omitting constant components, it reduces to the following form:

$$\sum_{i=1}^{n} (g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i))$$

The quantities $g_i$and $h_i$correspond to the first- and second-order derivatives of the loss function, respectively. Given a fixed tree structure $q(x)$, the optimal weight assigned to each leaf node j, denoted by w_j^*, along with the corresponding objective value T^*, can be determined analytically:

$$w_j^* = \frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}$$

Here, I_j denotes the set of training instances that fall into leaf j, and λrepresents the regularization parameter used to control model complexity. Furthermore, the scoring function T^*, which quantifies the quality of a given tree structure q, is derived accordingly.

$$T^* = \frac{1}{2} \sum_{j=1}^{J} \frac{\left(\sum_{i \in I_j} g_i\right)^2}{\sum_{i \in I_j} h_i + \lambda}$$

To improve the tree structure, LightGBM optimises the split by maximising the gain G, defined as:

$$G = \frac{1}{2} \left(\frac{\left(\sum_{i \in I_L} g_i\right)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{\left(\sum_{i \in I_R} g_i\right)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda} \right)$$

The sets $I_L$and $I_R$correspond to the observations assigned to the left and right child nodes of a split, respectively. In contrast to conventional GBDT methods that expand trees in a level-wise (horizontal) manner, LightGBM adopts a leaf-wise (vertical) growth approach. This strategy enables the model to scale more effectively to large datasets with high-dimensional feature spaces, improving computational efficiency.

3.7.Hyperparameter Tuning:

Hyperparameter tuning is performed using Bayesian optimization implemented through the Optuna framework (Akiba et al., 2019). The optimization procedure employs the Tree-structured Parzen Estimator (TPE), which sequentially explores the hyperparameter space by learning from previously evaluated configurations. Model performance in each trial is

assessed using walk-forward cross-validation with purging and embargo, and the parameter set that maximizes the average Matthews Correlation Coefficient (MCC) is selected.

3.8.Model Training and Evaluation:

The Bitcoin dataset is divided into three sequential subsets to preserve the temporal structure of the data: a training set (22.10.2019 to 29.07.2023), a validation set (30.07.2023 to 29.10.2024), and a test set (30.10.2024 to 30.01.2026). This time-series based split ensures that future observations do not influence the training of the machine learning model., thereby preventing look-ahead bias. Along with this data transformation is estimated using only the training data and subsequently applied to the validation and test sets so that out-of-sample information does not influence model construction.

During the training phase, model development is performed using time-series cross-validation. Specifically, the training data is divided into multiple folds that are sequential in nature, and a purging and embargo procedure is applied to eliminate any potential information leakage that can arise due to overlapping observations, following the framework proposed by López de Prado (2018). Matthews Correlation Coefficient (MCC) is used to assess model performance during cross-validation. The cross validation score is calculated as the mean MCC across all the folds, including the standard deviation of the same, to obtain the stability of model performance with time.

Hyperparameter tuning is done using this cross validation framework, and the configuration of parameters that gives the maximum of this average MCC is selected. The tuned model is tested on the validation set to check the model performance on unseen data.

Finally, the model is re-estimated with the training and validation data sets combined in order to make the best use of the information available. The final model performance is then evaluated on the test set which is the true out-of-sample period and is only used once for the final evaluation of predictive performance.

3.9.Evaluation Metric:

The model performance is evaluated not using the Accuracy, F1 score, Precision or Recall but rather the Matthews Correlation Coefficient (MCC). In contrast to accuracy and F1 score that

can give too optimistic judgment in case of class imbalance, MCC is informative even in the case where there is skewness in the distribution of classes.

In the case of meta-labeling, where the number of profitable and unprofitable trades may not be equal, MCC provides a better scale of the model's ability to distinguish correctly between positive and negative outcomes. Recent evidence indicates MCC is a more robust and informative evaluation metric that accuracy and F1 score for binary classification tasks, especially in the imbalanced class condition (Chicco & Jurman, 2020).

The Matthews Correlation Coefficient is defined as:

$$MCC = \frac{TP\ X\ TN - FP\ X\ FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Where TP = True Positives, TN = True Negatives, FP = False Positives, FN. = False Negatives.

MCC values range from −1 to 1, where:

A value of 1 represents perfect prediction, 0 corresponds to performance equivalent to random guessing, and −1 reflects complete disagreement between predicted and actual outcomes.

3.10. Backtesting:

A backtesting simulation is conducted to evaluate the economic performance of the proposed meta-labeled strategy in a realistic trading environment. For each observation in the out-of-sample test period, the required technical and market microstructure features are constructed using only the information available up to the decision time. The feature set includes only those variables identified as informative by the BorutaShap feature selection procedure.

The resulting features are then provided as inputs to the trained model, which generates a binary prediction indicating whether the baseline trading signal should be executed. Specifically, a prediction of 1 implies taking a long position in Bitcoin at the open of 21:00 UTC and closing the position at the open of 23:00 UTC, while a prediction of 0 indicates that the trade is skipped.

Strategy returns are computed only for the periods in which the model signals trade execution. This produces a filtered trading strategy in which the model selectively participates in trades generated by the baseline rule.

The performance of the meta-labeled strategy is then compared with the unconditional baseline strategy, which takes a position every day between 21:00 UTC and 23:00 UTC. This comparison allows an assessment of whether machine learning–based trade filtering improves risk-adjusted performance relative to the original seasonal strategy.

Let r_t denote the realized return of the strategy at time t, and T denote the total number of observations:

$$Cumulative\ Return = \prod_{t=1}^{T}(1 + r_t) - 1$$

$$CAGR = \left(\prod_{t=1}^{t}(1 + r_t)\right)^{\frac{N}{T}} - 1,$$

where Tdenotes the number of observed trading periods and $N$is the number of periods per year.

$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p}$$

Where R_p = average return of the strategy, R_f = risk-free rate (assumed to be zero) and σ_p = standard deviation of the strategy's returns.

The largest decline in cumulative returns from a peak to the subsequent low is measured by maximum drawdown (MDD). It displays the strategy's worst loss during the course of the test period. The MDD formula is:

$$MDD = \frac{Peak\ Value - Trough\ Value}{Peak\ Value}$$

This metric evaluates downside risk by capturing the largest observed loss over the trading period. Smaller drawdowns indicate a more stable portfolio.

4. RESULTS AND DISCUSSION:

Feature Validation:

The BorutaShap algorithm, that combines Boruta's feature selection methodology with SHAP-based feature importance measures, yields the values shown in Table 1. To prevent look-ahead bias, only the training data was used for feature selection. The chosen features were then applied consistently to the validation and out-of-sample test periods. In particular, Stochastic RSI has the highest SHAP importance, indicating the importance of short term overbought and oversold conditions. Volatility-based measures such as Volatility Ratio and Volatility Scaled Return (60 period) reflect regime change and risk-adjusted return dynamics. Trend persistence is measured by Average Directional Movement Index Rating, Dollar Volume and Williams %R add liquidity and further momentum information. Overall, the selected features indicate that the meta-labeling model is predominantly based on short-term momentum signals, volatility regime dynamics and market liquidity conditions.

Table 1: Selected Features Identified by the BorutaShap Algorithm

Feature	SHAP Importance	Interpretation
Stochastic RSI	2.878095	Short-term momentum/ overbought-oversold conditions
Volatility Ratio	0.973146	Short vs. long term volatility regime
Volatility Scaled Return (60 period)	0.703043	Risk-adjusted return intensity
Average Directional Movement Index Rating (14 period)	0.616933	Trend Strength persistence
Dollar Volume	0.470165	Market liquidity and trading activity
Williams %R (14 period)	0.453938	Momentum and reversal pressure

Hyperparameter Tuning:

Bayesian hyperparameter tuning using Optuna was done. Three important parameters of LightGBM have been considered in the search space: minimum number of observations per leaf (from 5 to 100 observations), maximum tree depth (from 3 to 25), number of boosting iterations (from 5 to 200 iterations). The average Matthews Correlation Coefficient based on walk forward cross-validation with embargo was used to select the model.

Table 2: Tuned Hyperparameter Values For LightGBM

Parameter	Search Range	Optimal Value
Number of iterations	**5 – 200**	**57**
Maximum tree depth	**3 – 25**	**3**
Minimum child samples	**5 – 100**	**90**
Number of trials	**–**	**100**
Best CV MCC	**–**	**0.139**

Table 3: Classification Performance (Matthews Correlation Coefficient)

Model	Training (CV) MCC	Validation MCC	Test MCC
Dummy Classifier (Prior)	0	0	0
LightGBM (Meta-label model)	0.113	0.091	0.096

A no skill benchmark was created using a Dummy Classifier set to "prior" strategy. Under this approach, the model does not make use of any feature information, but simply generates predictions based on the class distribution present in the training data. Specifically, it sets the weights to the frequencies of empirical classes and always predicts the majority class. As a result, the classifier does not capture any underlying patterns or relationships in the data, it purely acts as a baseline to random or uninformed decision making under class imbalance.

In consistence with this the dummy model resulted in a Matthews Correlation Coefficient (MCC) of 0.000 on the train, validation, and test data thereby confirming no predictive power. In contrast, the proposed LightGBM model gives positive MCC values for all the datasets. The performance of the model decreases from 0.113 to 0.091 in training period and validation period (i.e. 19.5% decrease in performance) indicating there is some loss in in-sample fit with validation dataset. However, in the test period the MCC becomes 0.096 (an increase of about 5.5% compared to the validation set), indicating stable out of sample performance. Overall, the test MCC is still around 15% lower than the training MCC, indicating that the level of generalization error is moderate, but not extreme (overfitting is not evident).

Table 4: Baseline Vs Meta-Labeled Strategy Performance

Strategy	CAGR	Sharpe Ratio	Max Drawdown
Baseline Strategy	33.33%	2.40	-7.40%
Meta-Label Strategy	35.58%	3.07	-5.35%

The results of the economic performance of the meta-labeled method and the baseline seasonal strategy over the out of sample period of testing are presented in Table 4. With a maximum drawdown of -7.40% and with a Sharpe ratio of 2.40, the baseline approach has a compound annual growth rate (CAGR) of 33.33%.

Meta-Labeling framework helps increase performance in all risk-return metrics. The CAGR rises to 35.58%. More significantly a notable improvement of a risk-adjusted return is indicated by the Sharpe ratio which goes from 2.40 to 3.07 with a significant improvement. Furthermore, the max drawdown goes from -7.40% to -5.35%, which means greater control over downside risk.

Shown in Fig. 1 below is a comparison between the equity curves for both the baseline and the meta-labeled strategies. The results indicate that compared to the baseline strategy, the meta-labeled strategy has a smoother equity curve, indicating a higher level of stability and lower level of volatility



Fig. 1: Equity Curves (30.10.2024 to 30.01.2026)

These results indicate that the model is effective in filtering low-quality trading signals, leading to higher risk-adjusted performance and more stable equity growth compared to the unconditional seasonal strategy.

5. CONCLUSION

This paper evaluates whether the performance of a calendar-based Bitcoin trading strategy could be improved using a machine learning-based meta-labeling strategy. A LightGBM model was applied to predicting the conditional probability that a trade would be profitable with the overnight time series seasonality from Vojtko and Javorská (2023) as the baseline signal. The findings show that the meta-labeling approach outperforms the unconditional baseline strategy in terms of the trade selection, leading to better Sharpe ratios and annualised returns with less maximum drawdown. These results indicate that data-driven signal filtering could potentially improve the robustness and risk-adjusted performance of simple calendar anomalies in cryptocurrency markets.

A high-dimensional feature collection of around 66 features that are mainly technical features and market microstructure features to capture momentum, volatility, liquidity and trading activities were used to build the model. The BorutaShap algorithm was applied to the feature selection in order to reduce noise while keeping the most useful predictors.

Some limitations should also be mentioned. The analysis is limited to Bitcoin and a single intraday trading window which could restrict its generalizability. In addition, the lookback periods of the technical indicators were fixed and not optimized, and the backtest assumes fixed transaction costs without considering the impact of the market and execution constraints. Future research may potentially extend the framework to other assets, optimize indicator parameters and assess performance under alternative market conditions.

Declaration of generative AI and AI-assisted technologies in the manuscript preparation process

During the preparation of this work the author(s) used Chatgpt in order to better formulate the sentences. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the published article.

REFERENCES

5. Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2623–2631). Association for Computing Machinery. https://doi.org/10.1145/3292500.3330701
6. Amihud, Y. (2002). Illiquidity and stock returns: Cross-section and time-series effects. *Journal of Financial Markets, 5*(1), 31–56. https://doi.org/10.1016/S1386-4181(01)00024-6
7. Anastasopoulos, A., Gradojevic, N., Liu, F., Maynard, A., & Tsiakas, I. (2026). Order flow and cryptocurrency returns. *Journal of Financial Markets, 79*, Article 101047. https://doi.org/10.1016/j.finmar.2026.101047
8. Bailey, D. H., Borwein, J. M., López de Prado, M., & Zhu, Q. J. (2017). The probability of backtest overfitting. *The Journal of Computational Finance, 20*(4), 39–69. https://doi.org/10.21314/JCF.2016.322
9. Ben Omrane, W., Houidi, F., & Savaser, T. (2024). Macroeconomic news and intraday seasonal volatility in the cryptocurrency markets. *Applied Economics, 56*(38), 4594–4610. https://doi.org/10.1080/00036846.2023.2212970
10. Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research, 13*, 281–305. https://jmlr.org/papers/v13/bergstra12a.html
11. Breedon, F., & Ranaldo, A. (2013). Intraday patterns in FX returns and order flow. *Journal of Money, Credit and Banking, 45*(5), 953–965. https://doi.org/10.1111/jmcb.12032
12. Caporale, G. M., & Plastun, A. (2020). Momentum effects in the cryptocurrency market after one-day abnormal returns. *Financial Markets and Portfolio Management, 34*, 251–266. https://doi.org/10.1007/s11408-020-00357-1
13. Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics, 21*, Article 6. https://doi.org/10.1186/s12864-019-6413-7

14. Corwin, S. A., & Schultz, P. (2012). A simple way to estimate bid-ask spreads from daily high and low prices. *The Journal of Finance, 67*(2), 719–760. https://doi.org/10.1111/j.1540-6261.2012.01729.x
15. Easley, D., O'Hara, M., Yang, S., & Zhang, Z. (2026). Microstructure and market dynamics in crypto markets. *Journal of Financial Markets*. Advance online publication. https://doi.org/10.1016/j.finmar.2026.101071
16. Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies, 33*(5), 2223–2273. https://doi.org/10.1093/rfs/hhaa009
17. Hasanli, R., & Dursun, M. (2025). Integrating high-dimensional technical indicators into machine learning models for predicting cryptocurrency price movements and trading performance: Evidence from Bitcoin, Ethereum, and Ripple. *FinTech, 4*(4), Article 77. https://doi.org/10.3390/fintech4040077
18. Hasbrouck, J. (1991). Measuring the information content of stock trades. *The Journal of Finance, 46*(1), 179–207. https://doi.org/10.1111/j.1540-6261.1991.tb03749.x
19. Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer. https://doi.org/10.1007/978-0-387-84858-7
20. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems* (Vol. 30, pp. 3146–3154). Curran Associates. https://proceedings.neurips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree
21. Kristoufek, L. (2018). On Bitcoin markets (in)efficiency and its evolution. *Physica A: Statistical Mechanics and Its Applications, 503*, 257–262. https://doi.org/10.1016/j.physa.2018.02.161
22. Kursa, M. B., & Rudnicki, W. R. (2010). Feature selection with the Boruta package. *Journal of Statistical Software, 36*(11), 1–13. https://doi.org/10.18637/jss.v036.i11
23. Kyle, A. S. (1985). Continuous auctions and insider trading. *Econometrica, 53*(6), 1315–1335. https://doi.org/10.2307/1913210
24. Lapitskaya, D., Eratalay, M. H., & Sharma, R. (2025). Prediction of cryptocurrency prices with the momentum indicators and machine learning. *Computational Economics, 66*, 2483–2501. https://doi.org/10.1007/s10614-024-10784-1
25. Liu, Y., Tsyvinski, A., & Wu, X. (2022). Common risk factors in cryptocurrency. *The Journal of Finance, 77*(2), 1133–1177. https://doi.org/10.1111/jofi.13119

26. Long, H., Zaremba, A., Demir, E., Szczygielski, J. J., & Vasenin, M. (2020). Seasonality in the cross-section of cryptocurrency returns. *Finance Research Letters, 35*, Article 101566. https://doi.org/10.1016/j.frl.2020.101566
27. López de Prado, M. (2018). *Advances in financial machine learning*. John Wiley & Sons.
28. Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence, 2*(1), 56–67. https://doi.org/10.1038/s42256-019-0138-9
29. Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems* (Vol. 30, pp. 4765–4774). Curran Associates. https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html
30. Makarov, I., & 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
31. Mourad, Z., & Gül, M. (2025). Crypto-driven growth: A comparative study of Bitcoin and Ethereum on economic growth for multi-country analysis. *Russian Journal of Economics, 11*(4), 403–425. https://doi.org/10.32609/j.ruje.11.164511
32. Pik, J., Chan, E. P., Broad, J., Sun, P., & Singh, V. (2025). *Hands-on AI trading with Python, QuantConnect, and AWS*. Wiley.
33. Probst, P., Wright, M. N., & Boulesteix, A.-L. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9*(3), Article e1301. https://doi.org/10.1002/widm.1301
34. Roll, R. (1984). A simple implicit measure of the effective bid-ask spread in an efficient market. *The Journal of Finance, 39*(4), 1127–1139. https://doi.org/10.1111/j.1540-6261.1984.tb03897.x
35. Sirignano, J., & Cont, R. (2019). Universal features of price formation in financial markets: Perspectives from deep learning. *Quantitative Finance, 19*(9), 1449–1459. https://doi.org/10.1080/14697688.2019.1622295
36. Urquhart, A. (2016). The inefficiency of Bitcoin. *Economics Letters, 148*, 80–82. https://doi.org/10.1016/j.econlet.2016.09.019
37. Vojtko, R., & Javorská, J. (2023). The seasonality of Bitcoin. *International Journal of Cryptocurrency Research, 3*(2), 26–35. https://doi.org/10.51483/IJCCR.3.2.2023.26-35

Statements and Declarations

Funding

The authors declare that no funds, grants, or other financial support were received during the preparation of this manuscript.

Competing Interests

The authors have no relevant financial or non-financial interests to disclose.

Author Contributions

Shantnu Sood: Conceptualization, methodology, data collection, data analysis, model development, and writing of the original draft.

Dr. Puneet Bhushan: Supervision, conceptualization, review and editing of the manuscript.

All authors read and approved the final manuscript.

APPENDIX

Indicator	Description
SMA (10) SMA (20), SMA (50)	Simple moving average of closing prices over the specified window; measures underlying trend.
EMA (10), EMA (20), EMA (50)	Exponential moving average giving more weight to recent prices.
Bollinger Bands Upper, Bollinger Bands Middle, Bollinger Bands Lower	Bollinger Bands; measure price deviation from its moving average and capture volatility.
Double EMA (20)	Double exponential moving average; reduces lag relative to standard EMA.
Triple EMA (20)	Triple exponential moving average; further lag reduction.
Weighted MA (20)	Weighted moving average emphasizing recent observations.
Mesa Adaptive Moving Average, Following Adaptive Moving Average	Adaptive moving averages that adjust to market cycles.
Parabolic SAR	Parabolic Stop-and-Reverse; identifies trend direction and potential reversals.
Relative Strength Index (14)	The Relative Strength Index (RSI) quantifies both the velocity and magnitude of recent changes in price.
MACD, MACD Signal, MACD Histogram	Difference between fast and slow EMAs; captures momentum and trend changes.
Stochastic Oscillator %K	The fast stochastic line that measures the position of the closing price relative to the high–low range over a specified period, indicating momentum and potential overbought/oversold conditions.
Stochastic Oscillator %D	A smoothed moving average of %K that acts as a signal line to identify momentum shifts and trading signals.
Stochastic RSI %K	This indicator is a momentum oscillator that uses the stochastic formula on RSI instead of price data. By measuring RSI relative to its recent high–low range, it

	becomes more sensitive to short-term changes in momentum.
Stochastic RSI %D	A smoothed moving average of STOCHRSI %K that serves as a signal line to identify momentum shifts and potential overbought/oversold conditions.
Average Directional Index (14 period)	Measures the strength of a trend regardless of its direction. Higher values indicate a stronger trend (either upward or downward), while low values indicate a weak or sideways market.
Average Directional Movement Index Rating (14 period)	A smoothed version of ADX that averages the current ADX with its past value to provide a more stable measure of trend strength. It reduces noise and reflects persistent trend conditions.
Positive Directional Indicator (+DI)	Evaluates how strongly prices have moved upward over a given time frame. Stronger buying pressure and upward trend are indicated by higher values.
Negative Directional Indicator (-DI)	Measures the strength of downward price movement over a specified period. Higher values indicate stronger selling pressure and downward directional movement.
Positive Directional Movement (+DM)	Measures the magnitude of upward price movement between consecutive periods. It captures how much the current high exceeds the previous high, representing buying pressure.
Negative Directional Movement (−DM)	Measures the magnitude of downward price movement between consecutive periods. It captures how much the previous low exceeds the current low, representing selling pressure.
AROON Up, AROON Down	Indicators that measure the time since the most recent high (Aroon Up) and the most recent low (Aroon Down) within a given period. They are used to identify trend strength and potential trend reversals.
AROON Oscilator	Difference between Aroon up and down.

Balance Of Power	The Balance of Power (BOP) indicator measures the relative strength of buyers versus sellers within a trading period by comparing the position of the closing price relative to the opening price within the high–low range.
Commodity Channel Index (14-period)	The Commodity Channel Index (CCI) measures the deviation of the price from its statistical mean over a specified period. It is used to identify overbought/oversold conditions and the strength of price trends.
Chande Momentum Oscillator (14-period)	The Chande Momentum Oscillator (CMO) looks at how much prices have recently gone up versus how much they've gone down during a certain time frame, and shows this as a percentage of all the price changes. It's a momentum indicator—much like the RSI—but instead of considering just price direction, it uses the total amount of increasing and decreasing moves
Money Flow Index (14-period)	The Money Flow Index (MFI) is a volume-weighted momentum indicator that measures buying and selling pressure by combining price and trading volume. It is often referred to as a volume-weighted version of the Relative Strength Index (RSI).
Momentum (10-period)	The Momentum indicator measures the rate of change in price by comparing the current closing price to the closing price nnn periods ago.
Percentage Price Oscillator (PPO)	The Percentage Price Oscillator (PPO) is a momentum indicator that calculates the percentage difference between a short-term and a long-term exponential moving average (EMA) of price.
Triple Exponential Average Oscillator (14 period)	TRIX measures the rate of change of a triple-smoothed exponential moving average (EMA) of price. It is designed to capture trend momentum while filtering out short-term noise.

Ultimate Oscillator	The Ultimate Oscillator is a momentum indicator that combines short-, medium-, and long-term price momentum into a single measure. It was designed to reduce false signals that occur when using a single time horizon.
Williams %R (14 period)	Williams %R tells how quickly the price is changing by looking at the current price in relation to the high and low prices from the past 14 times (days, hours etc.). This helps to spot when something is likely to be overbought or oversold.
Average True Range (14 period)	Average True Range (ATR) is a volatility indicator that measures the average magnitude of price movements over a specified period. It captures market volatility without indicating price direction.
Normalized Average True Range (14 period)	Normalized Average True Range (NATR) is a percentage-based volatility indicator that scales the Average True Range (ATR) relative to the asset's price
Hilbert Transform – Dominant Cycle Period	Estimates the dominant cycle length (in periods) present in the price series using the Hilbert Transform. It identifies the number of time periods over which the market exhibits a repeating cyclical pattern.
Hilbert Transform – Dominant Cycle Phase	Measures the phase position of the price within its dominant market cycle using the Hilbert Transform. It indicates where the market currently lies in the cycle (e.g., early, middle, or late stage of a cycle).The output is expressed in degrees (0° to 360°).
Amihud Lambda	Amihud illiquidity measure: absolute return divided by dollar trading volume; higher values indicate lower liquidity.
Kyles Lambda	Kyle's price impact measure capturing the sensitivity of price changes to trading volume.
Hasbrouck Lambda	Permanent price impact proxy based on absolute returns scaled by dollar volume.

Roll Measure	Bid–ask spread estimator based on the negative serial covariance of returns (Roll, 1984).
Corwin Schultz	High–low spread estimator based on price ranges over consecutive periods (Corwin and Schultz, 2012).
Volume	Total traded quantity within the period.
Dollar Volume	Trading volume multiplied by price; proxy for market activity and liquidity.
Signed Volume	Volume multiplied by the sign of returns; proxy for order flow imbalance (buy vs. sell pressure).
Volatility (6 hour)	Short-horizon realized volatility computed from squared log returns.
Volatility (24 hour)	Long-horizon realized volatility.
Volatility Ratio	Ratio of short-term to long-term volatility; captures volatility regime shifts.
Volatility Of Volatility	Rolling standard deviation of volatility; measures instability in volatility.
Return Volatility Scaled Return (60 period)	Return scaled by contemporaneous volatility (risk-adjusted return).
Trend Strength	Absolute cumulative return over a short window divided by volatility; proxy for directional strength.
Bekker–Parkinson Volatility	Range-based volatility estimator using high–low information, providing a more efficient estimate than close-to-close volatility.
Autocorrelation (1), Autocorrelation (2), Autocorrelation (3)	Rolling autocorrelation of returns at lag 1, 2, 3.
Hurst Exponent (500)	Hurst exponent estimated over a rolling window, values above 0.5 indicate persistence (trend), below 0.5 indicate mean reversion.
Permutation Entropy	Permutation entropy measuring the randomness and complexity of the return series (normalized between 0 and 1).
Fractal Dimension	Box-counting fractal dimension capturing the geometric complexity of price movements.