VILNIUS UNIVERSITY

FACULTY OF MATHEMATICS AND INFORMATICS

DATA SCIENCE STUDY PROGRAMME

Master’s thesis

Regime-Aware Bitcoin Price Forecasting Using Alternative Data and Machine Learning Methods

Režimais grįstas „Bitcoin“ kainos prognozavimas naudojant alternatyvius duomenis ir mašininio mokymosi metodus

Lukas Ivanovas

Supervisor : Mindaugas Juodis

Vilnius
2026

Acknowledgments

The author is thankful to the Supervisor for helping to understand the topic and helping find solutions to arisen problems.

Author is also thankful to Vilnius University professors for the given knowledge that helped reach the end goal.

Author also acknowledges, that Generative AI was used in the writing of this thesis, particularly in grammar of text and code debugging areas.

Summary

This thesis examines whether regime awareness and alternative data improve next-day Bitcoin log return forecasting. A daily dataset for 2019–2024 is constructed by combining market data (OHLCV and technical indicators) with on-chain activity, sentiment measures, and macro-financial variables, while regimes are included as an additional conditioning signal.

Models are evaluated using a strict chronological split with a held-out 2024 test set. The study compares a naive zero-return baseline against LSTM-based sequence modeling, LightGBM regression on engineered predictors, and a hybrid approach that augments LightGBM with LSTM embeddings. Performance is assessed using RMSE, MAE, and directional accuracy.

Results indicate that performance differences across model families are small and remain close to the naive benchmark, highlighting the limited predictability of daily Bitcoin returns. The best configuration provides only marginal RMSE improvements and directional accuracy remains near 0.5, suggesting limited standalone trading value but supporting the use of the framework for regime-conditioned interpretation and risk-oriented extensions.

Keywords: Bitcoin; regimes; alternative data; LSTM; LightGBM; hybrid modeling.

Santrauka

Šiame magistro darbe analizuojama, ar režimų atpažinimas ir alternatyvių duomenų įtraukimas pagerina kitos dienos „bitcoin" logaritminių grąžų prognozavimą. Tikslui pasiekti sudaromas 2019-2024 metų dieninis duomenų rinkinys, apjungiantis rinkos duomenis, „On-chain" aktyvumo rodiklius, sentimento matavimus bei makrofinansinius kintamuosius, o rinkos režimai įtraukiami kaip papildomas signalas.

Modeliai vertinami taikant griežtą chronologinį duomenų padalijimą, atskiriant 2024 metus kaip tikrinimo imtį. Tyrime lyginamas naivus nulinės grąžos bazinis modelis su LSTM pagrįstu sekų modeliavimu, „LightGBM" regresija, paremta sukonstruotais prognozuotojais ir hibridiniu metodu, kuriame „LightGBM" papildomas LSTM sugeneruotomis reprezentacijomis. Modelių našumas vertinamas pagal RMSE, MAE ir krypties prognozavimo tikslumą.

Gauti rezultatai rodo, kad našumo skirtumai tarp skirtingų modelių šeimų yra nedideli ir išlieka artimi naivaus bazinio modelio lygiui, pabrėžiant ribotą dieninių „Bitcoin" grąžų prognozavimumą. Geriausia konfigūracija suteikia tik nežymius RMSE pagerėjimus, o krypties prognozavimo tikslumas išlieka arti 0,5, kas leidžia teigti, jog savarankiško pirkimo ir pardavimo vertė yra ribota, tačiau metodinė sistema yra tinkama režimais grįstai interpretacijai ir tolesnėms, į riziką orientuotoms analizėms.

Raktiniai žodžiai: Bitcoin; režimai; alternatyvūs duomenys; LSTM, LightGBM; hibridinis modeliavimas

List of Figures

List of Tables

Contents

List of abbreviations

OHLCV	Open High Low Close Volume
LSTM	Long Short-Term Memory
LGBM/lightGBM	Light Gradient Boosting Machine
RMSE	Root mean Square Error
MAE	Mean Absolute Error
DA	Directional accuracy

Introduction

Bitcoin markets are characterised by high volatility, abrupt sentiment-driven transitions, and pronounced non-linear dynamics, which together make short-horizon forecasting both economically interesting and statistically difficult. While a large body of work has explored return and volatility prediction using econometric and machine learning methods, robust out-of-sample improvements at the daily horizon remain challenging, particularly once realistic evaluation protocols are enforced. This motivates a careful empirical assessment of whether two widely discussed ideas(i) incorporating regime awareness and (ii) extending feature sets with alternative data (on-chain activity, sentiment, and macro-financial indicators)can deliver measurable incremental forecasting value.

The core objective of this thesis is to evaluate a regime-aware forecasting framework for Bitcoin next-day log returns using a consistent dataset and a leakage-safe time split. The study compares progressively more complex model families, starting from a naive baseline and moving toward non-linear machine learning and representation learning. In addition to standard feature-based models, the thesis tests whether sequential embeddings learned by an LSTM can serve as compact latent factors that improve performance when used alone or when combined with engineered predictors in a hybrid LightGBM model.

The research is organised around three practical questions. First, does adding alternative data sources provide measurable improvements relative to market-derived predictors alone? Second, do learned LSTM embeddings contain predictive structure that is complementary to engineered features? Third, under a strict out-of-sample protocol, do hybrid architectures outperform simpler approaches, or do they mainly confirm the low signal-to-noise ratio of daily return prediction?

To answer these questions, the thesis constructs a daily modeling pipeline for 2019–2024 and evaluates models using a chronological train/validation/test split with final reporting on the 2024 test period. Forecast quality is measured with RMSE and MAE, with directional accuracy reported as an additional decision-relevant metric.

1 Literature Review

1.1 Dynamic Pricing and State-Dependent Decision Models

Dynamic pricing research provides a conceptual foundation for forecasting tasks in volatile markets such as cryptocurrencies. It distinguishes between the predictive task, which is estimating near-term returns or volatility and the prescriptive task, which is deciding how strongly to adjust prices, spreads, or positions given those forecasts. A recent synthesis defines dynamic pricing as adapting decisions to four underlying drivers — **people, product, period, and place**, clarifying how adjustments reflect demand, timing, and context [9]. Complementary work shows that when agents are forward-looking and compare current prices to internal reference points, responsive (state-dependent) policies outperform simple fixed rules [6].

Although crypto exchanges do not "set" asset prices in the retail sense, the same logic explains how market makers and liquidity providers adjust quotes, spreads, and fees in response to changing sentiment, order flow, and volatility. This parallel between adaptive retail pricing and algorithmic trading supports viewing crypto markets as state-dependent systems. In practice, exchanges widen spreads or rebalance inventory after large news shocks -comparable to dynamic pricing reactions to demand surges in other markets. The four-driver framework generalizes to crypto settings through participant mix, instrument design (spot versus perpetual futures), time-of-day and volatility cycles, and liquidity fragmentation across venues[9].

Recent financial-engineering studies further connect dynamic pricing with Markov decision processes, where each action depends on the system's current state and expected transition. Reinforcement-learning frameworks have been shown to outperform static trading strategies by adapting to volatility regimes and transaction costs in real time [7]. Such approaches formalize adaptive decision-making and bridge traditional economics with data-driven optimization.

Finally, behavioral dynamics play a crucial role. Rapid sentiment shifts can alter perceived value and liquidity conditions, producing reference-price and herding effects that justify state-dependent adjustments. Empirical evidence shows that investor sentiment can explain up to one-third of short-term volatility in major cryptocurrencies [11]. These findings reinforce that effective price modeling must incorporate not only historical and fundamental indicators but also behavioral and contextual signals.

In summary, dynamic pricing and state-dependent decision frameworks highlight why cryptocurrency forecasting must capture regime shifts, feedback mechanisms, and adaptive behaviors. This conceptual background justifies the methodological focus on models capable of learning from heterogeneous, rapidly evolving market states.

1.2 Forecast Methods for Blockchain Assets

Forecasting the price dynamics of cryptocurrencies remains one of the most challenging and rapidly evolving areas of financial data science. Early approaches relied on statistical time-series models such as ARIMA, VAR, and GARCH, which assume a relatively stable and stationary process. These models were effective for short-term trend identification in early market stages, but as cryptocurrencies matured and became more volatile, their limitations became evident. Studies have shown that GARCH-type models can capture conditional non-constant variance of the residuals but often fail to adapt to nonlinear shifts or multimodal drivers of volatility [3]. In particular, heavy-tailed distributions and structural breaks in crypto price series frequently violate the statistical assumptions underpinning classical econometric models.

The transition toward machine learning (ML) and deep learning (DL) methods marked a significant leap in forecasting accuracy and adaptability. Supervised ML algorithms such as Random Forest, Gradient Boosting, and LightGBM have become popular due to their robustness to noise and ability to model nonlinear dependencies. Empirical studies show that LightGBM can match or exceed deep recurrent baselines on short-horizon crypto tasks, particularly for price-trend classification and when heterogeneous features are used [16]. Such models handle large feature spaces efficiently, which makes them well suited for integrating market, on-chain, and sentiment variables.

Deep learning models, especially recurrent architectures like LSTM and GRU, have become central in modeling temporal dependencies in cryptocurrency markets. Their sequential memory structure helps capture long-range dependence and regime persistence better than many classical benchmarks. Comparative studies on Bitcoin, Ethereum, and Litecoin report that LSTM/GRU models consistently outperform ARIMA/GARCH baselines on standard error and direction metrics [14]. Broader comparisons that include both deep and ensemble learners similarly find recurrent networks among the top performers across multiple coins and horizons [5]. However, deep networks require careful hyperparameter tuning and sufficient data to generalize, and overfitting remains a risk in non-stationary settings.

Hybrid approaches that combine machine learning (ML) and deep learning (DL) have emerged as a promising compromise between interpretability and predictive power. A common design integrates an LSTM (for temporal encoding) with a tree-based learner such as LightGBM or XGBoost (for nonlinear regression and feature importance), or uses DL together with feature-selection pipelines. Empirical studies report that such hybrids and ensembles improve forecasting stability and accuracy relative to single models, especially in volatile crypto markets [16], [5], [13].

More recent developments also explore transformer-based architectures, which replace recurrence with self-attention mechanisms, enabling the model to focus on relevant time steps dynamically. Studies using attention-enhanced models for crypto forecasting demonstrate significant improvements in long-horizon forecasts, particularly when combining price, sentiment, and volatility features [12]. The flexibility of attention mechanisms allows such models to capture interdependencies between different cryptocurrencies or cross-market variables, an essential step toward multi-asset modeling frameworks.

Overall, forecasting methods for blockchain assets have evolved from rigid, statistically grounded techniques toward highly flexible data-driven systems. While classical models like ARIMA and GARCH still serve as important baselines for volatility modeling, machine learning and hybrid deep-learning approaches dominate the current research landscape due to their superior performance in nonlinear and multimodal contexts. The literature increasingly points to hybrid, ensemble, and transformer-based architectures as the most promising direction for capturing the complex, state-dependent, and multidimensional nature of cryptocurrency price formation.

1.3 Multimodal and Hybrid Modeling Approaches

As the complexity of cryptocurrency markets increased, researchers began integrating diverse data modalities to capture both quantitative and behavioral determinants of price dynamics. Traditional time-series inputs—such as OHLCV data—proved insufficient for explaining abrupt regime shifts or sentiment-driven volatility bursts. To address this, recent studies emphasize multimodal data fusion, combining market, on-chain, sentiment, and macroeconomic features within unified predictive architectures. This approach reflects the multidimensional nature of crypto markets, where investor attention, blockchain activity, and broader financial indicators interact in nonlinear ways [1].

Multimodal learning frameworks employ different strategies for feature integration. The most common are early fusion, where heterogeneous features are concatenated before training; late fusion, which combines the outputs of independent models; and hybrid fusion, which integrates both levels. For instance, sentiment extracted from Twitter, Reddit, or news sources is often transformed into polarity indices and merged with numerical on-chain metrics such as transaction count or hash rate. One study found that investor sentiment is significantly connected with cryptocurrency risk spillovers, especially when sentiment and market variables interact [11]. This demonstrates the empirical value of multimodal information in capturing market psychology and microstructure effects.

Hybrid architectures combine statistical and machine-learning components to leverage complementary strengths. A common design integrates models such as GARCH or ARIMA to capture conditional variance patterns, together with ML models—such as LightGBM, SVR, or LSTM—for modeling nonlinear residual dynamics. Empirical studies indicate that these hybrid approaches often outperform single-model methods in forecasting performance and robustness, especially in volatile markets, when applying a fusion of sequential and tree-based techniques [5], [10]. The hybrid concept also extends to deep-learning ensembles, where model stacking can mitigate overfitting and enhance stability over varying regimes.

An important evolution in multimodal modeling involves attention- and graph-based learning, which allow the model to weigh information sources dynamically. For instance, transformer architectures and graph neural networks (GNNs) have been applied to jointly model correlations between assets, exchanges, and blockchain activity. Recent studies demonstrate that attention-based fusion of price, volatility, and sentiment data can enhance forecasting performance compared with conventional recurrent models, particularly in capturing inter-asset dependencies and temporal context [12] This highlights the advantage of selectively emphasizing the most informative signals in real time—a crucial property for high-frequency and regime-sensitive markets.

While multimodal and hybrid modeling approaches consistently outperform simpler baselines, they also introduce challenges related to data synchronization, noise, and interpretability. Disparate update frequencies between data sources—such as high-frequency trading data and slower sentiment indices—can distort model training. Furthermore, integrating unstructured text or social data introduces additional preprocessing complexity, requiring embedding and normalization techniques that preserve semantic meaning without bias. As a result, some recent studies have turned toward explainable hybrid architectures, combining model ensembles with SHAP or attention-based interpretability layers to visualize the relative contribution of each data source [8],[4].

In summary, multimodal and hybrid modeling represents the frontier of cryptocurrency forecasting research. By merging structured and unstructured data, these models better reflect the informational diversity of blockchain ecosystems. Their success lies not only in predictive accuracy but also in their ability to capture the behavioral, technical, and macro-financial interactions that drive price dynamics. Consequently, multimodal forecasting frameworks serve as a necessary bridge between raw data heterogeneity and the adaptive decision systems explored in the following section on volatility and regime dependence.

1.4 Volatility and Regime Dependence

One of the defining characteristics of cryptocurrency markets is their extreme volatility and recurrent regime shifts. Unlike traditional assets, where volatility tends to cluster within predictable bounds, digital asset markets exhibit sharp transitions between tranquil and turbulent phases, often triggered by sentiment shocks, liquidity shortages, or macroeconomic announcements. Early econometric work, such as the GARCH family of models, provided a foundational understanding of conditional variance dynamics but proved inadequate in capturing abrupt nonlinear transitions or multi-regime behaviors commonly observed in crypto assets [3]. These limitations led to the growing interest in regime-switching and volatility-aware forecasting frameworks.

Regime-switching models, including Markov-Switching GARCH (MS-GARCH) and Hidden Markov Models (HMMs), have been applied to detect and forecast structural breaks in cryptocurrency volatility. Such models assume that market behavior alternates between latent “states,” each characterized by distinct volatility persistence and mean return levels. Empirical studies show that regime-switching models outperform static GARCH specifications by meaningful margins in out-of-sample volatility forecasts, and can detect transitions linked to macro or sentiment-driven shocks [3].These results confirm that volatility in decentralized markets is not constant but context-dependent, responding dynamically to informational asymmetries and behavioral reactions.

Beyond purely econometric approaches, deep-learning-based volatility models have gained traction for capturing nonlinear and time-varying dependencies. LSTM- and GRU-based forecasting frameworks are capable of learning from multimodal inputs such as trading volume, realized volatility, and sentiment, and hybrid models—combining LSTM/GRU with GARCH components—have shown superior out-of-sample performance in crypto-markets [2]. These architectures capture asymmetric reactions to positive and negative shocks—an essential feature of speculative markets.

Regime awareness has also become central to modern crypto risk modeling. Researchers increasingly emphasize that predictive models should not only estimate volatility but also identify the state of the market—bullish, neutral, or bearish—since feature importance and model performance vary significantly across regimes. Integrating regime labels into ML training allows adaptive weighting of predictors such as sentiment polarity, transaction counts, or macro indicators. A 2024 study employing LightGBM within a regime-aware ensemble showed improved directional hit ratio and better robustness during crisis periods, illustrating how state-dependent architectures enhance stability [14].

Volatility dependence also reveals behavioral underpinnings in crypto trading. Sudden liquidity withdrawals or speculative surges often correspond to collective herding and panic cycles. Behavioral-finance research links these volatility bursts to cognitive biases such as loss aversion, overreaction to news, and imitation effects, all of which intensify during high-stress market phases [11]. Consequently, volatility modeling should not be treated purely as a statistical challenge but as a behavioral one, where regime shifts reflect the interplay between information diffusion and human decision-making under uncertainty.

In summary, volatility and regime dependence represent critical dimensions of cryptocurrency forecasting. Models that incorporate regime-awareness and nonlinear volatility learning outperform static predictors and better reflect real-world trading dynamics. Understanding how volatility evolves across behavioral, structural, and informational regimes is essential for constructing adaptive models that respond to sudden market transitions—a principle directly informing the methodological framework developed later in this thesis.

1.5 Summary and Research Gap

Addressing these gaps requires developing a forecasting framework that jointly integrates multimodal data, accounts for regime shifts, and remains interpretable enough for practical financial applications. This thesis aims to fill that gap by constructing a regime-aware multimodal forecasting model for cryptocurrency price sensitivity and volatility analysis. The next chapter - **Survey of Knowledge (SOK)** - extends this review by organizing prior studies and methodological insights into a structured overview of data types, modeling paradigms, and evaluation techniques that inform the empirical design of the research.

The reviewed literature demonstrates a clear evolution in how researchers conceptualize and model cryptocurrency price behavior. Early works drew heavily on traditional dynamic pricing and time-series models, treating crypto assets as extensions of conventional financial instruments. These approaches helped establish fundamental understanding but were limited by assumptions of linearity and stationarity. As markets matured, it became evident that state-dependent and adaptive mechanisms better describe price formation, where investor sentiment, liquidity, and external shocks continually reshape equilibrium conditions.

The transition from econometric frameworks to machine learning and deep learning models marked a major methodological shift. Studies consistently report that ensemble and deep architectures—such as LightGBM, LSTM, and GRU—achieve higher forecasting accuracy than classical ARIMA or GARCH baselines. The improvement is especially evident when integrating richer data sources, including on-chain activity and market sentiment. Nonetheless, despite these gains, model interpretability and generalization across regimes remain persistent challenges.

Recent advances in multimodal and hybrid modeling have opened promising avenues for integrating diverse data modalities—combining numerical, textual, and network-based inputs. These methods have demonstrated superior performance in capturing complex interactions and behavioral dynamics within blockchain ecosystems. However, multimodal learning also introduces new technical and conceptual difficulties, particularly regarding data alignment, feature fusion, and the transparent attribution of model decisions. The literature increasingly calls for frameworks that can balance predictive power with interpretability and adaptability.

At the same time, the volatility and regime-dependence literature underscores that cryptocurrency markets operate under non-stationary and highly state-sensitive conditions. Models that ignore regime transitions risk overfitting to temporary trends or failing during structural breaks. This reinforces the need for regime-aware forecasting systems capable of recognizing and adjusting to distinct market phases—such as speculative rallies, corrections, and stable consolidations—rather than treating price dynamics as homogeneous over time.

Taken together, the existing research highlights rapid progress but also reveals several unresolved gaps.

2 Survey of Knowledge

2.1 Summary of Existing Knowledge

Cryptocurrency markets exhibit strong non-stationarity: relationships between returns, volatility, and explanatory variables are not stable over time, and the same indicator can behave differently during expansions, drawdowns, and recovery phases. For forecasting, this implies that model credibility depends not only on the modelling algorithm but also on whether the data representation and evaluation design respect temporal structure and prevent information leakage. Two requirements follow. First, the modelling dataset must integrate information that reflects different mechanisms behind market movements rather than relying only on price-derived indicators. Second, all data blocks must be aligned so that features used for prediction correspond to information that was available at the time of forecasting.

To satisfy these requirements, the thesis uses a block-based dataset aligned to a common daily index. The daily index provides a consistent unit of observation across heterogeneous sources and supports regime inference as a persistent state process. Market OHLCV variables provide the baseline representation of realised price behaviour and remain indispensable for both regime detection and forecasting. On-chain activity complements this baseline by capturing blockchain utilisation and network stress, which are not directly observable from price features and tend to evolve more slowly. Sentiment and attention variables represent the information environment and behavioural component of the market, allowing the analysis to account for event-driven periods where narrative intensity may amplify volatility or accelerate directional moves. Macroeconomic variables provide external context and enable the analysis to test whether crypto dynamics change under broader risk conditions and cross-asset pressure.

A practical challenge of multimodal integration is that sources originate at different granularities and exhibit different publication delays. Therefore, dataset construction must specify alignment rules explicitly. In this thesis, market data is available at daily frequency and is used directly, followed by feature construction after alignment. On-chain and sentiment sources are aggregated from transaction/post-level observations to daily summaries, which makes them compatible with the daily modelling index. Macroeconomic series are converted to the daily index using forward filling, with lag-aware handling where appropriate to reduce the risk of incorporating values that would not have been observable at prediction time. The result is a single daily dataset where each feature corresponds to information that can be treated as available on (or before) the corresponding day.

In addition to raw information blocks, the thesis treats regime indicators as a derived block of information. The regime label and posterior regime probabilities from the HMM provide a compact representation of market state that supports phase-dependent diagnostics and interpretation. Using posterior probabilities is particularly useful around transitions, where regime membership can be uncertain and a soft representation avoids forcing abrupt boundaries.

1 table. summarises the data blocks used in this thesis, their raw frequency, daily alignment logic, representative variables, and the motivation for including each block.

***1 table.** Data blocks used in the thesis and their role in the modelling pipeline*

Data block	Source	Raw frequency	Daily alignment / aggregation	Key variables (examples)	Why it matters in this thesis

Market (OHLCV)	Binance API	1d	daily close; daily volume; derived features computed after alignment	BTC_close;
				BTC_log_ret;
				BTC_vol30;
				BTC_mom30;
				BTC_drawdown;
					Baseline information set for regime detection and forecasting; captures realized price response and market dynamics

On-chain activity	BigQuery public dataset	block/tx-level	aggregate to daily sums/means; optional rolling smoothing computed after daily alignment	btc_active_addresses, btc_tx_count, btc_total_fees_btc, btc_tx_volume_btc	Adds network-utilization state; improves regime interpretability and provides non-price context for market phases
Sentiment / attention	Reddit / CryptoPanic + FinBERT	post/news-level	daily counts and sentiment aggregates;	vader_compound_mean, post_count, finbert_score_mean_btc, news_count_btc	Captures investor attention and event-driven narrative shocks; complements technical indicators during discontinuities
Macroeconomic context	FRED / yFinance	daily/weekly/ monthly	forward fill with reporting lag; convert to daily index; lagged versions for safety	dxy_yf, sp500_yf, us10y_yield, vix_yf, gold_fut	Provides external risk/environment context; supports robustness checks and sensitivity analysis
Regime indicators (derived)	HMM output	daily (derived)	hard label + regime probabilities	BTC_regime; regime_prob_0–3	Operationalizes market phases; enables regime-aware modelling and diagnostics across states

2.2 Evaluation Metrics and Model Assessment

Model assessment in this thesis follows a time-ordered holdout design. The dataset is split chronologically into training/validation and a final test period: training and validation are constructed from the interval 2019-06-30 to 2023-12-31, while the test set consists of the 2024 period. This split reflects a realistic forecasting setting in which models are trained on historical data and evaluated on a later, unseen market period. The chronological separation also prevents information leakage that can occur under random splits in time-series settings.

Forecast accuracy is evaluated using Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE), which quantify the magnitude of forecast deviations. RMSE penalises large errors more strongly and is sensitive to volatility spikes, while MAE provides a robust measure of typical deviation that is less dominated by outliers. In addition to point accuracy, directional behaviour is evaluated using Directional Accuracy and Hit Ratio, which measure how often the model correctly predicts the sign of the next-period movement. These behavioural metrics are included because decision relevance in trading and risk contexts depends not only on magnitude but also on whether the model captures the correct direction.

Finally, because performance can vary across market phases, results are examined conditionally by regime where relevant. This ensures that conclusions are not driven only by one period or one market state and provides a more informative view of model stability under changing conditions.

2.3 Linking to the Thesis Methodology

The knowledge consolidated above motivates three methodological decisions implemented in this thesis. First, the modelling dataset is built as a daily-aligned multimodal panel with explicit aggregation and alignment rules, so that each feature corresponds to information available at the time of prediction. Second, regime information is introduced as a derived representation of market state, enabling phase-dependent diagnostics and supporting regime-aware interpretation. Third, model performance is assessed using a chronological train/validation/test split with a dedicated out-of-sample test year (2024), combined with both error-based and directional metrics.

This design translates the dataset and modelling choices into testable expectations: whether additional information blocks contribute signal beyond market baselines, whether regimes form coherent and persistent market phases rather than scattered switching, and whether predictive performance remains stable when evaluated on a later market period that was not used during model fitting.

3 Methodology

3.1 Data Collection

3.1.1 Pricing data

The primary source of price data used in this thesis is the **Binance cryptocurrency exchange**, which provides reliable, high-frequency market data through its public API. Binance is one of the largest global cryptocurrency exchanges by trading volume, and its spot market for Bitcoin (BTC) is considered deep, liquid, and representative of general market dynamics. This makes it an appropriate benchmark for constructing the core time series used in forecasting experiments.

For the purposes of this research, historical OHLCV (Open–High–Low–Close–Volume) data was extracted directly from the `Binance REST API`. The API allows querying historical candlestick data for any trading pair at various time intervals. In this study, `BTC/USDT` (Bitcoin traded against Tether) was selected due to its high liquidity and consistent availability across the entire study period. The USDT-denominated pair is also standard in empirical crypto-market research, as it avoids distortions from fiat currency conversion rates.

Although the Binance API provides OHLCV data at multiple granularities (e.g., 1-minute, 5-minute, hourly), this work uses daily candlesticks. Daily data is more stable, less noisy, and more suitable for forecasting tasks involving macroeconomic and sentiment variables, which are also available at daily frequency. Moreover, using daily data reduces the risk of overfitting, improves computational efficiency, and enables easier alignment with external datasets such as macroeconomic indicators or on-chain activity metrics.

To ensure reproducibility, the data collection process was automated using Python. The script sends requests to the Binance endpoint with parameters specifying:

- trading pair: BTCUSDT,
- time interval: 1 day,
- start and end timestamps covering January 2019 to end of 2024

The API returns each candlestick as a structured array containing the daily open, high, low, close prices, trading volume, and timestamps. These values were parsed into a `pandas` DataFrame, converted to proper datetime format, and sorted chronologically. Integrity checks were applied to ensure there were no missing days or duplicate entries. Finally, the dataset was enriched with additional engineered variables (e.g., log returns, volatility measures, technical indicators), which serve as input features for subsequent forecasting models.

Using `Binance API` has several advantages: the data is free, programmatically accessible, consistently formatted, and updated in real time. Because the exchange reports high-quality price information with deep market liquidity, the resulting dataset provides a robust foundation for modeling short-term and medium-term BTC price dynamics.

3.1.2 Macroeconomic elements

In addition to market-based information, this thesis incorporates a set of macroeconomic and global financial indicators that are commonly used in empirical research on asset pricing, volatility modeling, and risk forecasting. Although Bitcoin is a digital and decentralized asset, several studies have shown that its medium-term dynamics can be influenced by broader macro-financial conditions, including liquidity cycles, monetary policy expectations, equity market sentiment, and global risk appetite. For this reason, a selected group of macroeconomic variables was collected and aligned with the daily BTC dataset.

Macro data was obtained from two primary sources:

1. **The Federal Reserve Economic Data (FRED) API**, provided by the Federal Reserve Bank of St. Louis
2. **Yahoo Finance (yFinance Python package)** for market-based indices

These sources were selected because they offer free, programmatically accessible data with extensive historical coverage and reliable update schedules.

The `FRED API` provides a comprehensive repository of U.S. macroeconomic time series. Several variables were selected based on their use in prior literature examining monetary policy effects and risk sentiment on cryptocurrency markets. Specifically, the following indicators were collected:

- **M2 Money Supply (M2SL)**: captures broad liquidity conditions in the financial system.
- **Consumer Price Index (CPIAUCSL)**: used to approximate inflation trends.
- **Federal Funds Effective Rate (FEDFUNDS)**: reflects short-term U.S. monetary policy stance and interest rate expectations.

Because these indicators are published at different frequencies (monthly for M2 and CPI, daily for the federal funds rate), they were aggregated into a unified daily time index. Missing days were filled using forward fill, a standard approach in financial econometrics when aligning higher-frequency market data with lower-frequency macro series. This ensures consistent time alignment without introducing artificial volatility.

The data extraction was automated using the official `FRED API`. Python scripts queried each selected series, converted it to a pandas DataFrame, normalized the date format, and merged the indicators into a consolidated macro dataset. Additional transformations, such as year-over-year percentage changes (e.g., CPI YoY), were computed to capture trend dynamics rather than raw level changes.

Yahoo Finance Market Indices

To capture broader global financial sentiment, the following indices were retrieved using the `yfinance` library:

- **S&P 500 Index (^GSPC)**: a proxy for overall equity market performance and investor risk-taking behavior.

- **CBOE Volatility Index (VIX)**: a widely used measure of global risk aversion.
- **U.S. Dollar Index (DXY)**: tracks the strength of the U.S. dollar relative to other major currencies, relevant because Bitcoin is commonly viewed as an alternative store of value.

These indices are available at daily frequency, making them fully compatible with the BTC OHLCV time series. The downloaded data was cleaned by handling missing values (e.g., weekends and exchange holidays) and then aligned with the BTC dataset via the primary date field.

Once retrieved, all macroeconomic variables were merged into the core BTC dataset on a daily basis. When necessary, forward filling was applied to ensure continuous coverage. This unified dataset allows the forecasting models to incorporate both market internal dynamics and broader macro-financial signals, enabling a more comprehensive evaluation of how external economic conditions might influence Bitcoin's return predictability.

3.1.3 On-Chain data

On-chain activity provides valuable insights into blockchain network utilization, user adoption, and transactional intensity, all of which can influence or reflect market dynamics. Although several commercial providers offer aggregated on-chain metrics (e.g., CoinMetrics, Glassnode, CryptoQuant), many of these platforms impose paywalls or restrict access to historical data. Initial attempts to collect daily on-chain metrics through the CoinMetrics Community API resulted in incomplete BTC coverage, missing historical windows, and inconsistent metric availability. Due to these limitations, this thesis relies on **Google BigQuery's public blockchain datasets**, which provide historical, transaction-level Bitcoin data without access restrictions.

Google BigQuery maintains high-quality, continuously updated public datasets under the namespace `bigquery-public-data.crypto_*`. These datasets contain raw blockchain ledger data (blocks and transactions), enabling researchers to derive custom on-chain metrics directly from primary sources. This approach offers several advantages: (i) reproducibility, since the data is openly accessible; (ii) broad historical coverage; and (iii) the ability to compute metrics tailored to this research, rather than relying on vendor-specific definitions.

For Bitcoin, the dataset `bigquery-public-data.crypto_bitcoin` was used, which contains tables such as:

- **Blocks** – block-level metadata (e.g., timestamp, block hash, size)
- **Transactions** – all transactions included in each block (including transferred value and fees)
- **Inputs / Outputs** – granular details on transaction flows and address references

To construct daily on-chain indicators relevant for forecasting, SQL queries were written and executed through the BigQuery client, aggregating raw blockchain data to a daily frequency. The computed Bitcoin indicators include:

- **Daily active addresses**: number of unique addresses observed on either the input or output side of transactions per day;

- **Daily transaction count**: total number of confirmed transactions per day;
- **Transaction value and fee measures**: total transferred output value (in BTC), average transaction value, total fees, and average fee;
- **Transaction structure measures**: average number of inputs and outputs per transaction;
- **Block production proxies**: daily block count and average block size (bytes).

Each query aggregated raw blockchain data to a daily timeframe, enabling direct alignment with the daily OHLCV price series. The extracted tables were exported as CSV files and merged with the main dataset using the calendar date as the joining key. Any missing values introduced by feature construction (e.g., rolling windows) or merge alignment were retained and handled during model-specific preprocessing to avoid introducing artificial information into the time series.

Using Google BigQuery for on-chain data collection provides a transparent and academically defensible methodology. It avoids reliance on proprietary analytics services, ensures reproducibility, and grants flexibility to define on-chain metrics in a consistent and customizable manner. This approach creates a robust foundation for examining how blockchain-level network activity contributes to the predictability of Bitcoin returns.

3.1.4 News Articles and Reddit posts

In addition to market-based and on-chain indicators, this thesis incorporates social and news text data in order to capture the broader behavioural and informational environment surrounding Bitcoin. Compared to traditional financial markets, the cryptocurrency ecosystem is strongly shaped by online discourse and rapid information diffusion. For this reason, text-based sentiment features were included as an additional modality in the forecasting framework.

Reddit hosts some of the largest and most active cryptocurrency discussion communities and is frequently used in academic work as a proxy for retail investor sentiment. In particular, subreddits such as **r/Bitcoin** and **r/CryptoCurrency** contain continuous discussions about market narratives, protocol updates, macro news reactions, and short-term trading views. Such content can provide timely signals about market attention and prevailing sentiment.

Direct historical access to Reddit content is limited through the official Reddit API due to rate limits and restricted access to older posts. To obtain a complete historical sample, this thesis relies on publicly available Reddit archives[17], which provide full subreddit snapshots in compressed `.zst` format. These archives were processed to extract all posts within the selected study window, followed by basic filtering to remove unusable records (e.g., missing text fields) and to reduce low-quality noise (e.g., minimal engagement). For sentiment modelling, each post was represented using the combined textual content (title and body where available), and the resulting post-level dataset was later scored using VADER and aggregated to daily indicators

News information was collected from a `GitHub` repository which includes a dataset that aggregates cryptocurrency news items originally distributed via the CryptoPanic platform[15]. CryptoPanic acts as a crypto news aggregator, providing structured entries that typically include a news headline, publishing timestamp, short description/snippet, and metadata used for categorization.

To prepare the news data for sentiment scoring, the dataset was cleaned by standardizing timestamps to a daily date index, removing duplicated items (e.g., repeated headlines or repeated URLs when present), and retaining only observations with sufficient textual information for natural language processing. News items were then assigned to the relevant asset category (Bitcoin-focused subset) using the available metadata and/or keyword-based filtering in the title/description. After cleaning, the resulting article-level table served as input to the FinBERT sentiment pipeline, and the scored outputs were aggregated to daily measures (mean sentiment and daily article counts) for merging into the master dataset

3.2 Setting Sentiment using VADER and finBERT

To add a simple measure of market mood, two text-based sentiment sources were converted into daily numeric features: (i) Reddit posts scored with VADER, and (ii) crypto news articles scored with FinBERT. Both outputs were aggregated to a daily frequency and later merged into the master dataset using the `date` column.

3.2.1 Reddit sentiment with VADER

Reddit submissions were first converted into a clean "core" table by streaming raw `.zst` files, keeping only posts within the selected time window and applying a basic quality filter based on minimum post score. The text used for sentiment was created by combining the post title and body (when present) into a single field.

VADER was then applied to each post to obtain standard sentiment components (`neg`, `neu`, `pos`) and the `compound` score. These post-level scores were aggregated to daily indicators using simple summary statistics (mainly daily means) together with activity controls such as the number of posts and score summaries.

3.2.2 News sentiment with FinBERT

For news sentiment, the transformer model `ProsusAI/finbert` was used via `AutoTokenizer` and `AutoModelForSequenceClassification`. For each article, the model produces probabilities for positive, negative, and neutral sentiment. A single continuous sentiment score was defined as:

$$s_i^{\text{FinBERT}} = p_i^{\text{pos}} - p_i^{\text{neg}}.$$

Finally, article-level results were aggregated to daily values (grouped by `date` and `asset`) and exported as a daily table for merging with the rest of the features.

3.3 Data Processing and Cleaning

After collecting the individual data sources (OHLCV market data, on-chain indicators, macro-financial variables, and sentiment proxies), a unified daily panel dataset was constructed to ensure that all predictors are temporally aligned and directly usable for downstream modelling. Since the sources differ in coverage, frequency, and structure (e.g., some sentiment tables are sparse, and some macro variables are reported less frequently), a consistent cleaning pipeline was applied: (i) standardizing the date index, (ii) merging all sources by date, (iii) resolving structural duplication introduced by asset-tagged sentiment tables, and (iv) handling missing values in a way that preserves the time-series interpretation and avoids introducing forward-looking information.

3.3.1 OHLCV Derived Values

The base market series consists of daily OHLCV bars $(O_t, H_t, L_t, C_t, V_t)$. From the closing price C_t, both simple returns and log-returns are computed to capture day-to-day price changes. Log-returns are used as the primary stationary transformation due to their time-additive property:

$$r_t = \ln\left(\frac{C_t}{C_{t-1}}\right), \qquad R_t = \frac{C_t - C_{t-1}}{C_{t-1}}.$$

To represent short-term and medium-term risk dynamics, rolling volatility features are computed as the moving standard deviation of log-returns over window length w (e.g., $w = 7$ and $w = 30$ days):

$$\sigma_{t,w} = \sqrt{\frac{1}{w-1}\sum_{i=0}^{w-1}(r_{t-i} - \bar{r}_{t,w})^2}, \qquad \bar{r}_{t,w} = \frac{1}{w}\sum_{i=0}^{w-1} r_{t-i}.$$

Trend and momentum are captured using moving averages of C_t (e.g., 7-, 30-, and 90-day simple moving averages) and exponential moving averages (e.g., 12- and 26-day EMAs). Additional technical indicators are included to provide compact measures of market state, such as RSI(14) for relative strength and MACD (EMA12–EMA26) together with its signal line (9-day EMA of MACD). Finally, volume dynamics are represented via the percentage change in traded volume, which helps distinguish periods of unusually high or low activity.

For supervised learning, the next-step prediction target is constructed directly from the log-return series via a one-day forward shift:

$$y_{t+1} = r_{t+1}.$$

3.3.2 Temporal alignment and merging strategy

All datasets were first converted to a common daily date format and sorted chronologically. The merge was performed using the calendar date as the joining key, producing a single time-indexed table that combines OHLCV-derived features with on-chain, sentiment, and macro-financial variables. To avoid inconsistent sample lengths across sources, the final modelling window was restricted to the period where the alternative datasets provide coverage (up to 2024-12-31), ensuring that the combined dataset remains comparable across all feature groups.

3.3.3 Missing-value handling

Missing values were handled according to the interpretation of each feature group:

- **Sentiment features (news and Reddit):** Missing sentiment values were treated as *no observable signal* for that day and replaced with zeros. This is particularly important for sparse sources (e.g., days without scraped news items or without sufficient Reddit activity), where the absence of entries should not remove the day from the modelling sample.
- **Macro-financial indicators:** Macro series were retained as continuous predictors over the daily grid after merging. For variables published at a lower frequency (e.g., monthly), the daily dataset inherits step-wise behaviour across days, and the modelling design additionally uses lagged variants (see next subsection) to avoid contemporaneous information leakage.

3.3.4 Lag construction and leakage prevention for low-frequency macro variables

To ensure a realistic forecasting setup, selected lower-frequency macroeconomic variables (e.g., money supply, inflation, policy rate) were transformed into lagged predictors. Specifically, for each monthly series x_t, a one-day lag x_{t-1} was constructed and used as an input feature. This design prevents the model from exploiting same-day releases that would not be known at prediction time in a strict end-of-day forecasting workflow.

3.3.5 Column consistency and final quality checks

After merging all sources, redundant columns originating from overlapping providers (e.g., duplicated market series) were removed to avoid multicollinearity and ambiguity in variable definitions. Finally, the cleaned dataset was validated by checking remaining missingness across all columns. The resulting master dataset was exported as a single CSV file for modelling, with only negligible missingness remaining in lagged macro variables due to the mechanical shift operation on the first available observation.

3.4 Exploratory Data Analysis

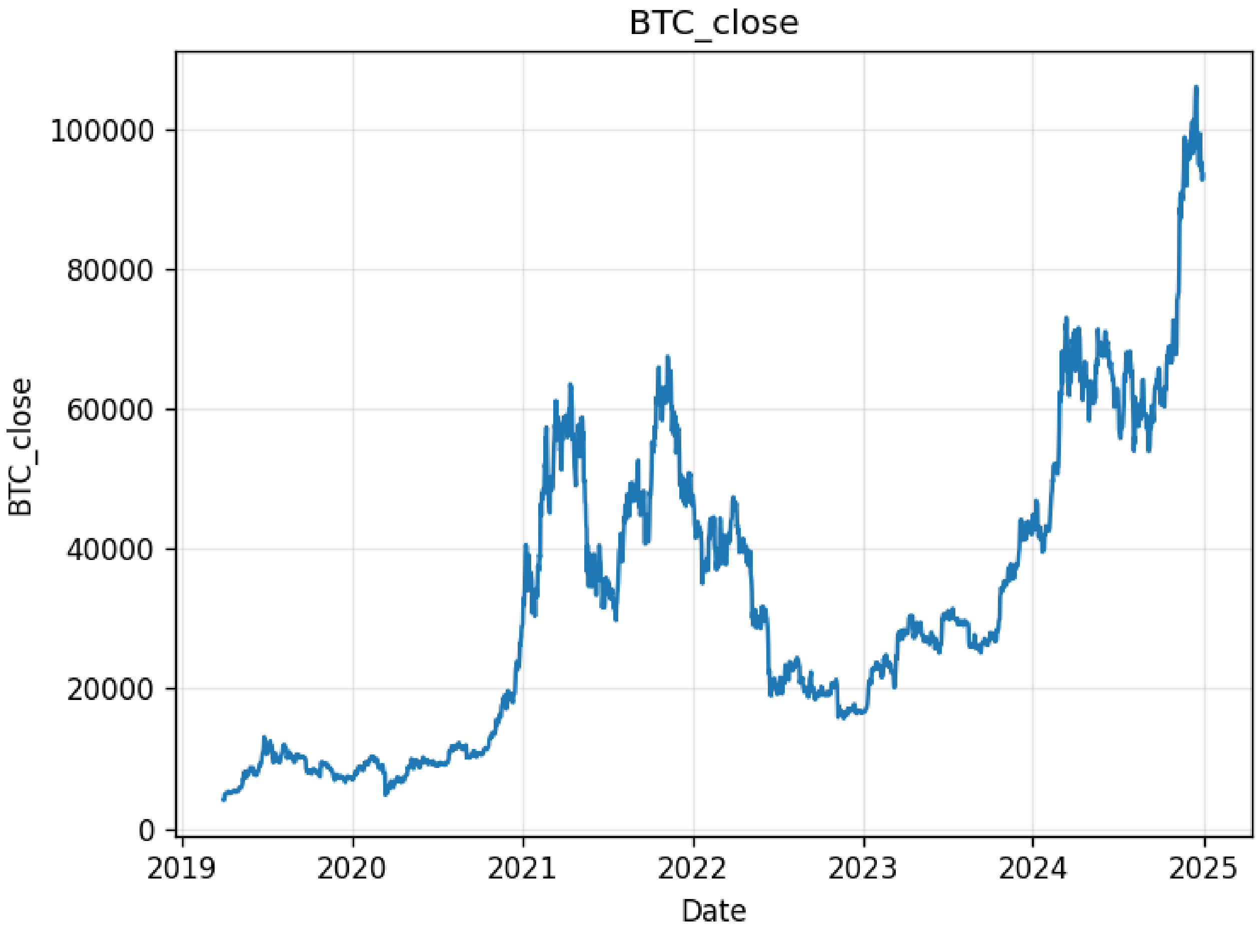
3.4.1 Data integrity and quality checks

The dataset was sorted chronologically by the date index and inspected for missing values, duplicates, and trivial predictors. Overall data quality is high. Only three lagged variables contain missing values (`M2_money_stock_lag1`, `CPI_all_items_lag1`, `federal_funds_rate_lag1`), each with a single missing observation (approximately 0.05% of rows). No duplicated rows were detected. In addition, no constant or near-constant numeric predictors were found, indicating that all retained variables carry time variation that could potentially contribute to modelling.

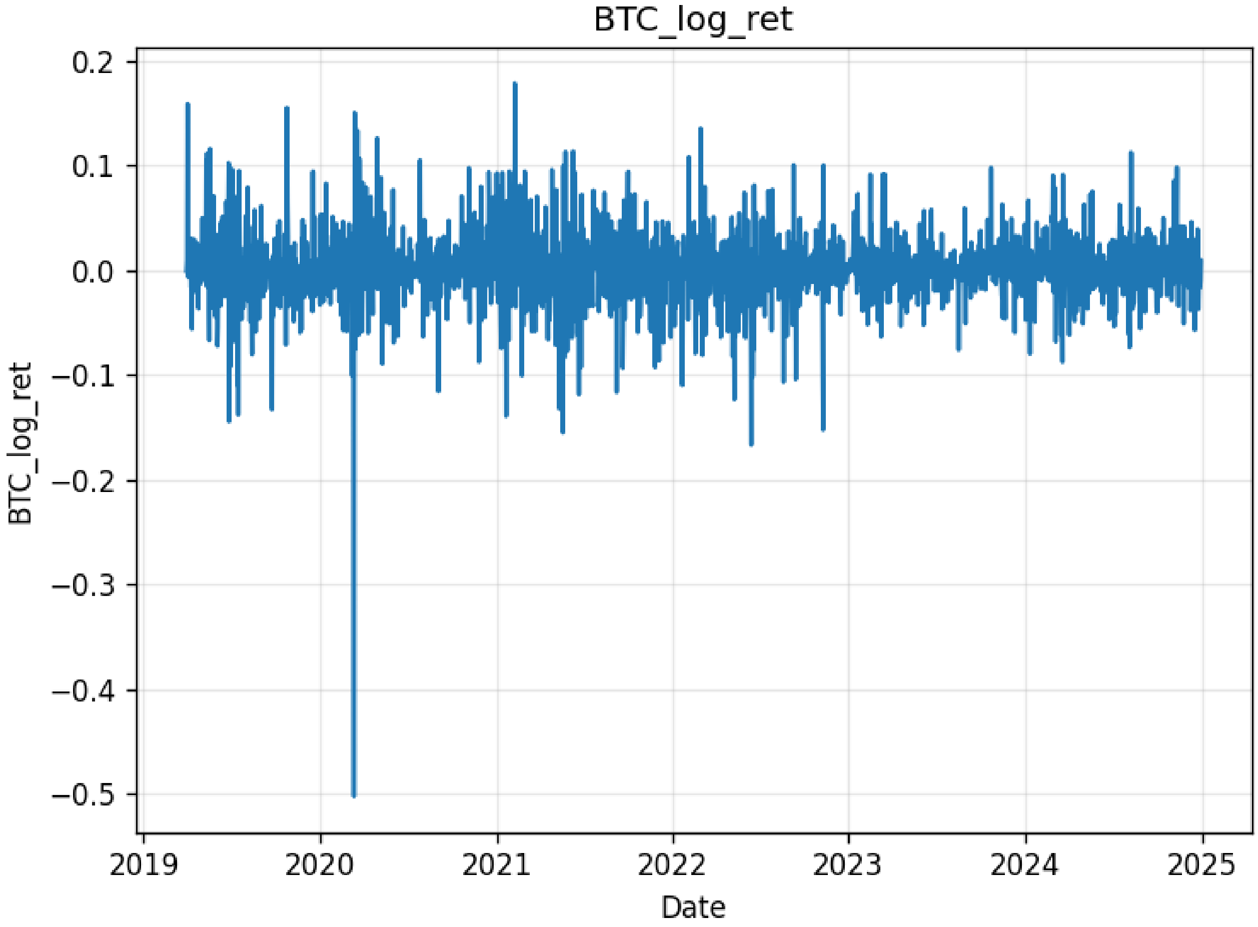
3.4.2 Market dynamics and target behaviour

Figure 1 figure. shows the Bitcoin closing price over the full sample, highlighting multiple market cycles and pronounced regime changes. Since the price level is non-stationary, the modelling target is defined in return space. Figure 2 figure. presents daily Bitcoin log returns, which fluctuate around zero but exhibit occasional extreme moves.

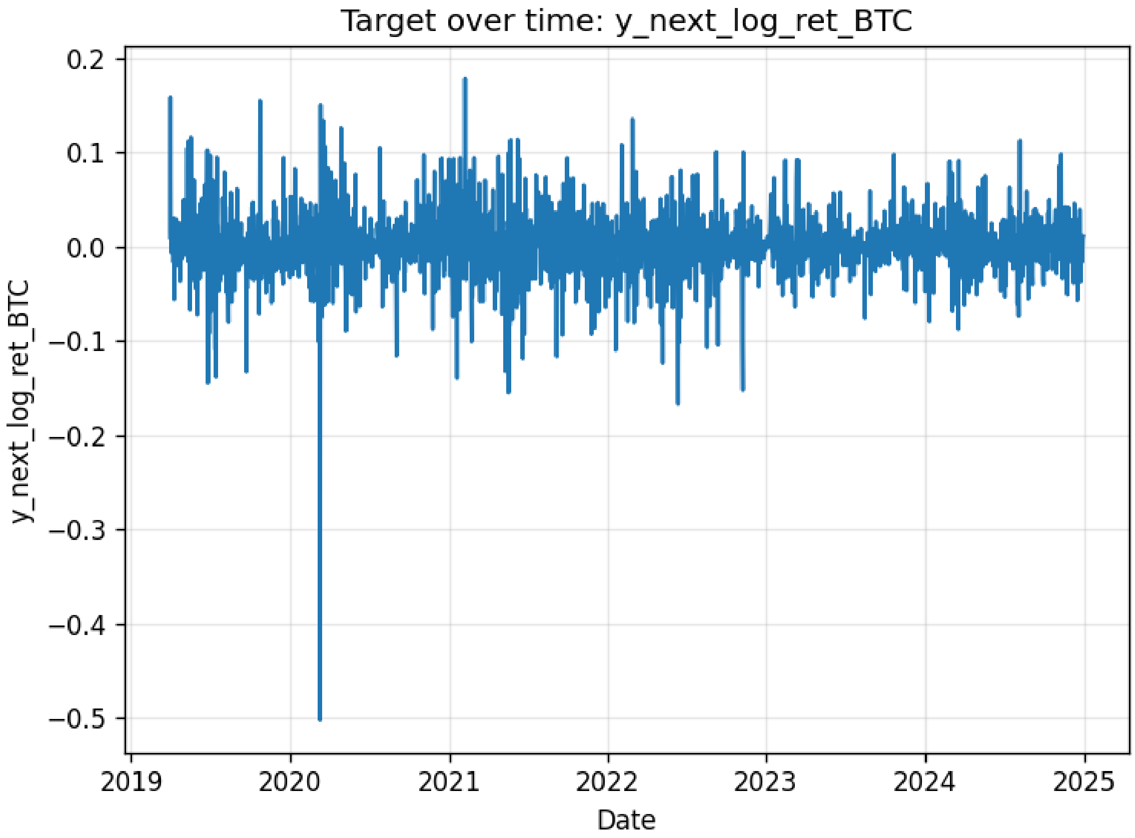
The forecasting target is the next-day log return (`y_next_log_ret_BTC`). Its empirical distribution is sharply peaked near zero (Figure 3 figure.), while the Q–Q plot versus a normal distribution (Figure 4 figure.) shows clear tail deviations, confirming heavy-tailed behaviour. These distributional properties motivate the use of robust evaluation metrics (MAE alongside RMSE) and support the use of loss functions that are less sensitive to rare outliers in training.



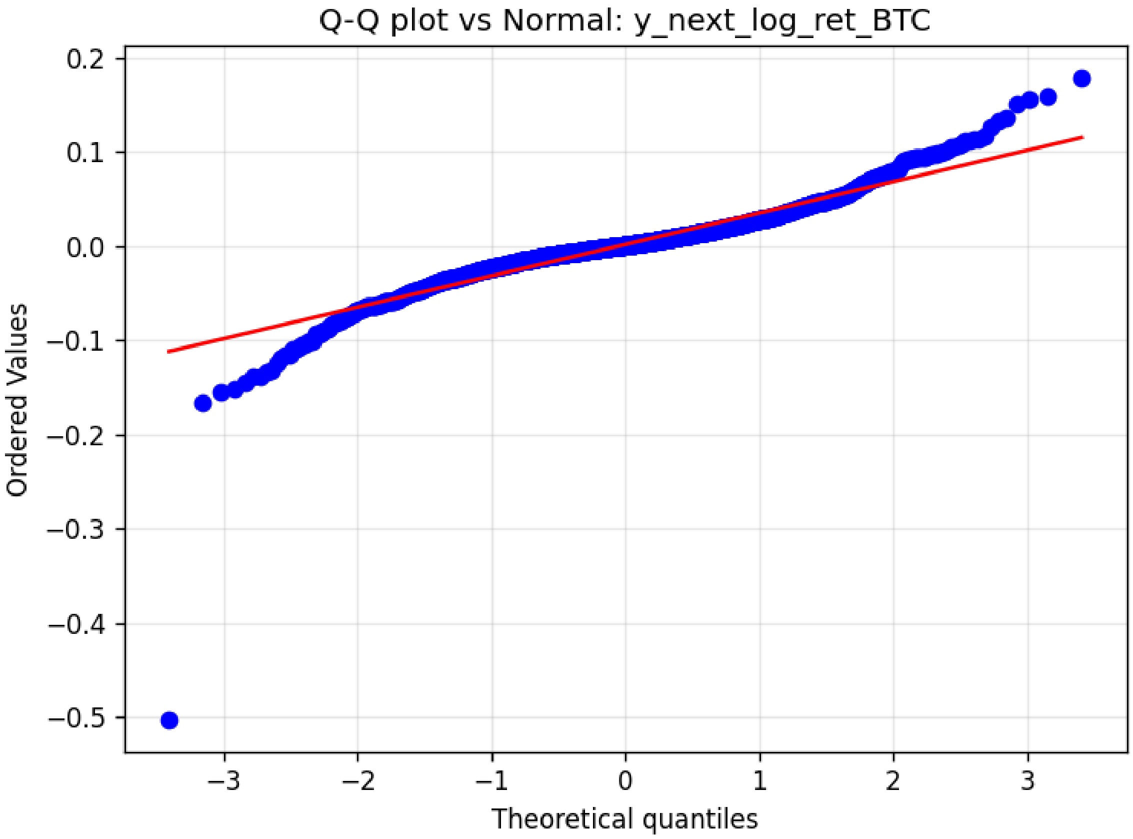
1 figure. *Bitcoin closing price over time (daily).*



2 figure. *Bitcoin daily log returns over time.*



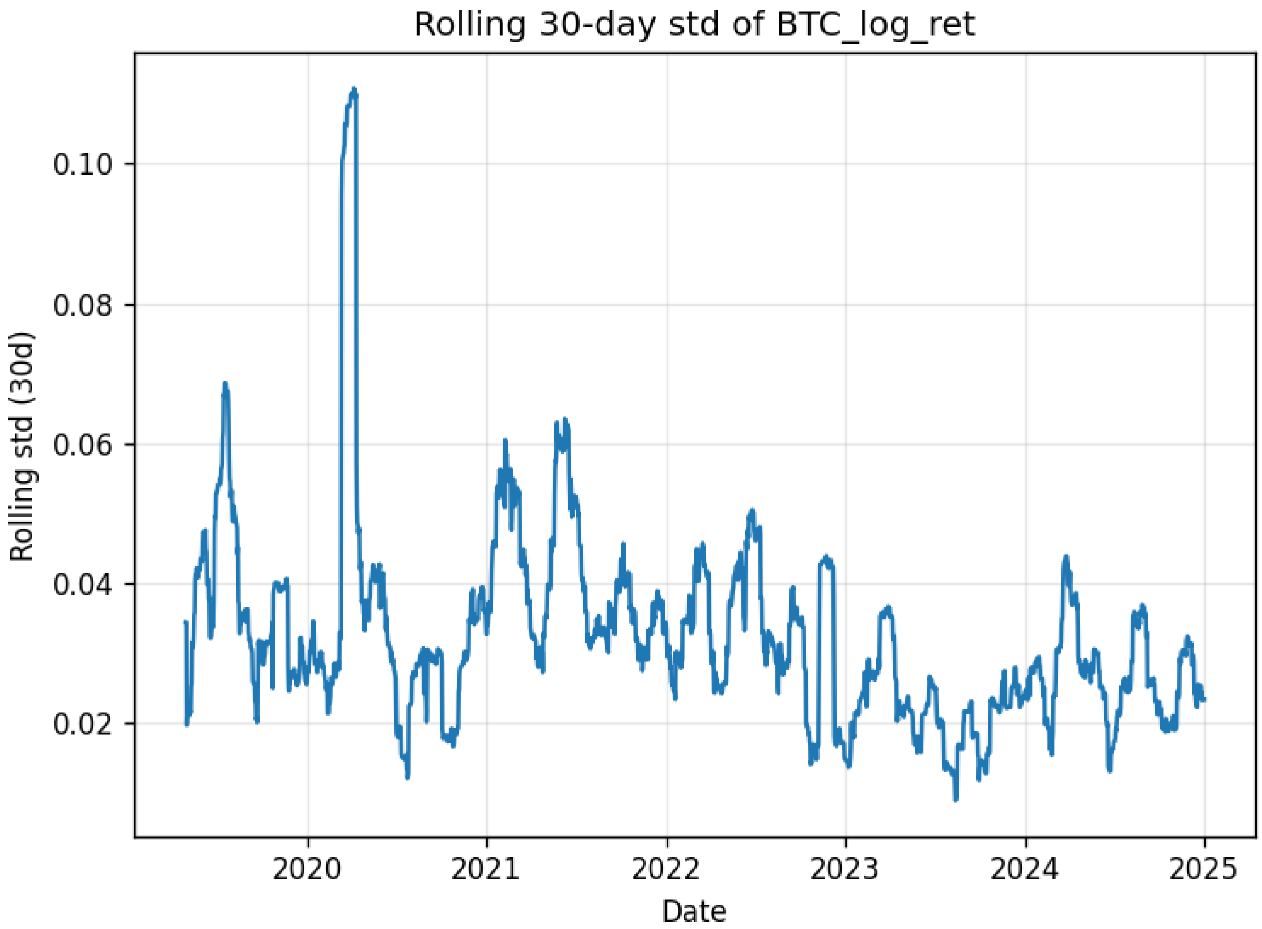
3 figure. *Histogram of the next-day BTC log return target ($y\_next\_log\_ret\_BTC$).*



4 figure. *Q–Q plot of the target versus a normal distribution, highlighting heavy tails.*

3.4.3 Volatility clustering

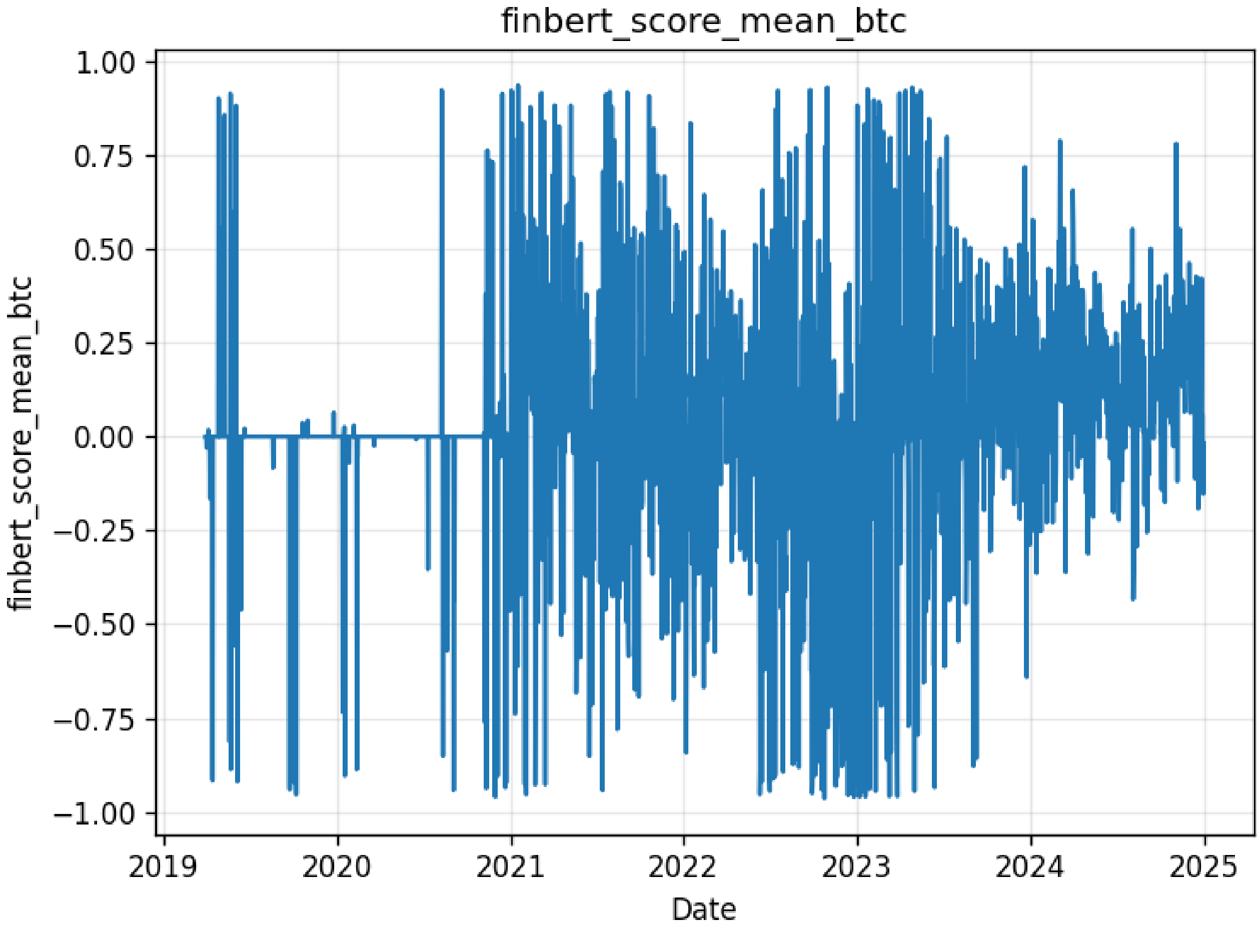
A key stylised fact in financial time series is volatility clustering, where large absolute returns tend to occur in contiguous periods. This effect is visible in the time-varying dispersion of returns, and is summarised using a rolling volatility proxy. Figure 5 figure. plots the rolling 30-day standard deviation of Bitcoin log returns, showing extended high-volatility and low-volatility phases. This observation supports the inclusion of volatility-based predictors (e.g., rolling standard deviations) and motivates later modelling components that explicitly address regime and risk dynamics.



5 figure. *Rolling 30-day standard deviation of Bitcoin log returns (volatility proxy).*

3.4.4 Example alternative signals

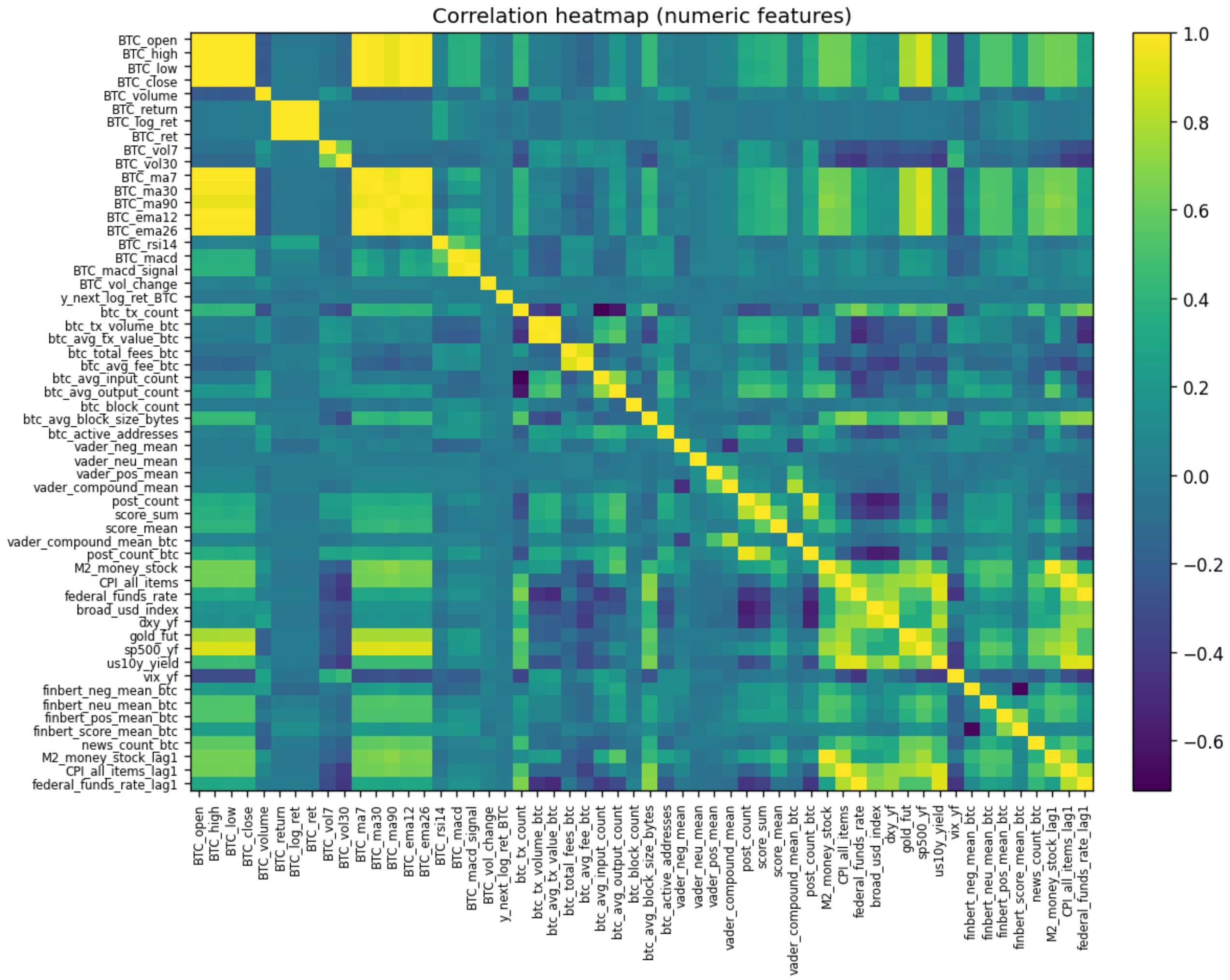
To confirm that non-price features provide meaningful time variation, representative series from the alternative data blocks were inspected. For example, sentiment measures such as the FinBERT score (Figure 6 figure.) exhibit pronounced fluctuations over time, indicating a potentially informative signal channel. Similar time variation is present in on-chain activity indicators and macro-financial variables. In this thesis, these variables are used as candidate predictors alongside market-derived features.



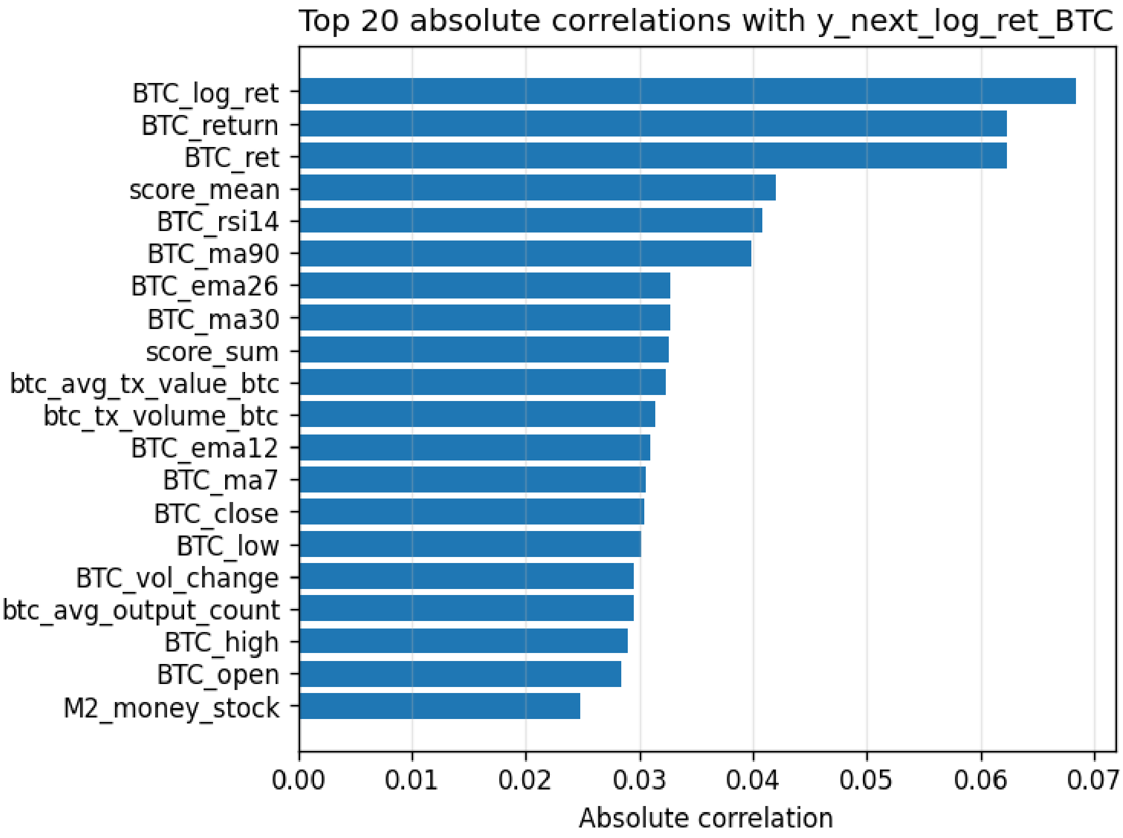
6 figure. *Example sentiment signal over time: $finbert\_score\_mean\_btc$.*

3.4.5 Correlation structure

Finally, the linear correlation structure across numeric predictors was inspected. Figure 7 figure. shows the correlation heatmap, where strong blocks are primarily observed among highly related market variables (e.g., OHLC price features and overlapping technical indicators). However, the next-day target exhibits only weak linear correlations with individual predictors. This is illustrated by the top absolute correlations with the target (Figure 8 figure.), where even the strongest relationships remain small in magnitude. This finding aligns with the general expectation that next-day return prediction is a low-signal task and that potential predictability may be non-linear, regime-dependent, or concentrated in specific periods rather than expressed as stable linear correlations.



7 figure. *Correlation heatmap for numeric variables. Strong correlations are concentrated among closely related market-derived features.*



8 figure. *Top 20 absolute correlations with the next-day BTC log return target. Correlations are generally weak, consistent with low predictability at the daily horizon.*

3.5 Regime Awareness

Cryptocurrency markets exhibit pronounced non-stationarities and structural changes, where the relationship between explanatory signals (e.g., volatility, volume, on-chain activity) and price dynamics can differ substantially across market phases. To account for these shifts, a regime-aware layer was introduced by estimating latent market states using a Hidden Markov Model (HMM). The resulting regime labels and/or regime probabilities are then used downstream as conditioning information for forecasting and risk analysis.

3.5.1 Data preparation and feature engineering

The HMM was fit on a daily Bitcoin dataset (sorted chronologically) and constructed on top of a dedicated table to isolate regime-relevant information. The following groups of explanatory variables were used:

1. Return and volatility characteristics

   - **BTC_log_ret** (daily log return)
   - **BTC_vol30** (long-horizon volatility proxy; computed in preprocessing)

2. Trend and stress characteristics (price-derived, but not price level)

- 30 day momentum
- Drawdown from peak

3. Abnormal activity in trading volume (Volume z-score)

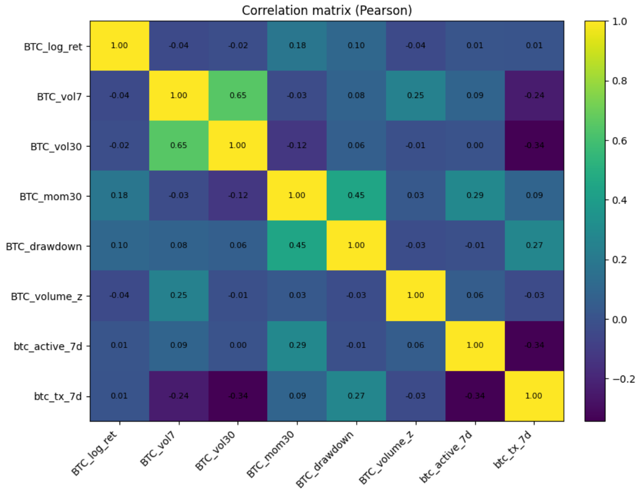
4. Smoothed on-chain activity

   - **btc_active_7d** as a 7-day moving average of active addresses
   - **btc_tx_7d** as a 7-day moving average of transaction count

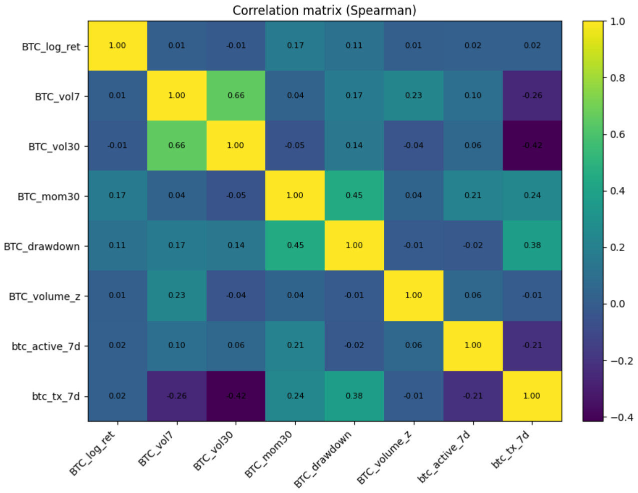
Rolling-window transformations necessarily introduce missing values at the beginning of the sample. Therefore, rows affected by initial NaNs were removed (dropna()), leaving 2073 daily observations for HMM estimation. When regimes were merged back into the full dataset (left join by date), the earliest dates remained unlabeled (NaN regime) purely due to this rolling-window truncation rather than data quality issues.

3.5.2 Correlation screening and feature set refinement

Before fitting the HMM, pairwise dependence among candidate features was examined via Pearson and Spearman correlation matrices (9 figure. , 10 figure.). The most notable overlap was observed between short- and long-horizon volatility proxies, with BTC_vol7 showing moderate-to-strong correlation with BTC_vol30 (≈0.65 in both Pearson and Spearman). To avoid redundant volatility signals within the HMM emission model, BTC_vol7 was excluded from the final regime feature set. The remaining variables exhibited only weak-to-moderate correlations, supporting their use as complementary descriptors of market state rather than duplicates of the same effect.



9 figure. *Pearson Correlation Matrix*



10 figure. *Spearman Correlation Matrix*

3.5.3 HMM specification and estimation

A GaussianHMM (from `hmmlearn`) with four hidden states was estimated using a diagonal covariance structure (covariance_type=”diag”). A diagonal covariance was preferred as a stability-oriented choice for financial time series, since it reduces parameter complexity and mitigates overfitting risks when features are correlated or heavy-tailed.

Because the features operate on very different scales (e.g., returns vs. on-chain levels), the feature matrix was standardized using `StandardScaler` prior to model fitting. This step is critical: without scaling, the HMM likelihood can become dominated by the largest-magnitude variables (typically on-chain activity), which would distort the inferred regimes.

The model converged successfully (monitor flag indicated convergence), providing a consistent basis for regime inference and interpretation.

3.5.4 Selecting the number of regimes

To explore model complexity, HMMs were estimated for $K = 2, ..., 10$ regimes, and **AIC/BIC** values were computed (2 table.). Both information criteria decreased as the number of regimes increased, indicating improved in-sample fit with higher K. This pattern is typical in regime models applied to long financial samples: additional states often split broad phases into sub-phases (e.g., separating “bull” into calm vs euphoric, or “bear” into capitulation vs recovery).

However, the objective of this thesis is not only fit, but also actionable and interpretable regime awareness that can be integrated into subsequent forecasting models and risk logic. For that reason, $K = 4$ was selected as a pragmatic compromise: it supports a market narrative close to **bull/bear/sideways/panic** while keeping the regime layer stable, explainable, and usable for conditioning downstream models.

2 table. *AIC/BIC values for HMMs with $K = 2$ to $K = 10$*

K	AIC	BIC
2	36809	36983
3	33386	33668
4	30901	31302
5	29237	29767
6	28913	29584
7	25582	26405
8	24855	25841
9	24102	25263
10	23980	25327

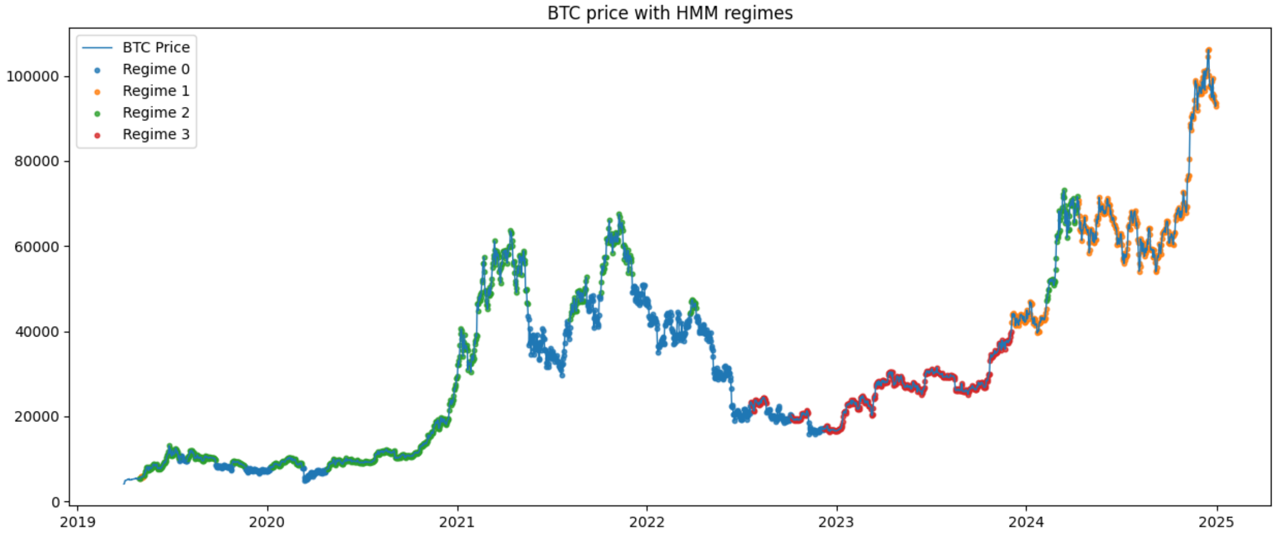
3.5.5 Regime interpretation and labeling

After fitting, regimes were interpreted using the regime-conditional means of the input features (3 table.) and a visual overlay of regimes on the BTC price path (11 figure.). Using the estimated state-dependent averages:

- **Regime 0 (Panic)**: negative average return, highest volatility, negative momentum, deep drawdowns, and positive volume z-score consistent with stress-driven trading.
- **Regime 1 (Sideways)**: relatively low volatility, small positive drift, moderate drawdown, and subdued abnormal volume.
- **Regime 2 (Bull)**: highest average return and strongest momentum, with moderate volatility and controlled drawdown.
- **Regime 3 (Bear)**: characterized by the deepest drawdown (distance from peak), reflecting post-peak market depression; returns can be mildly positive on average due to relief rallies, but the level remains far below the previous high.

3 table. *Regime-conditional means for interpretation*

BTC_regime	BTC_log_ret	BTC_vol30	BTC_mom30	BTC_drawdown	BTC_volume_z	btc_active_7d	btc_tx_7d
0	-0.0035	0.0420	-0.1248	-0.4750	0.1172	866223.2367	262592.6355
1	0.0015	0.0258	0.0564	-0.1569	-0.0283	770991.4529	555052.4441
2	0.0045	0.0342	0.1992	-0.1558	-0.0117	943192.2544	313601.9162
3	0.0021	0.0218	0.0758	-0.6125	-0.0487	945504.0950	375040.0960



11 figure. *Bitcoin Prices with Regime Overlay*

3.5.6 Regime persistance and transition dynamics

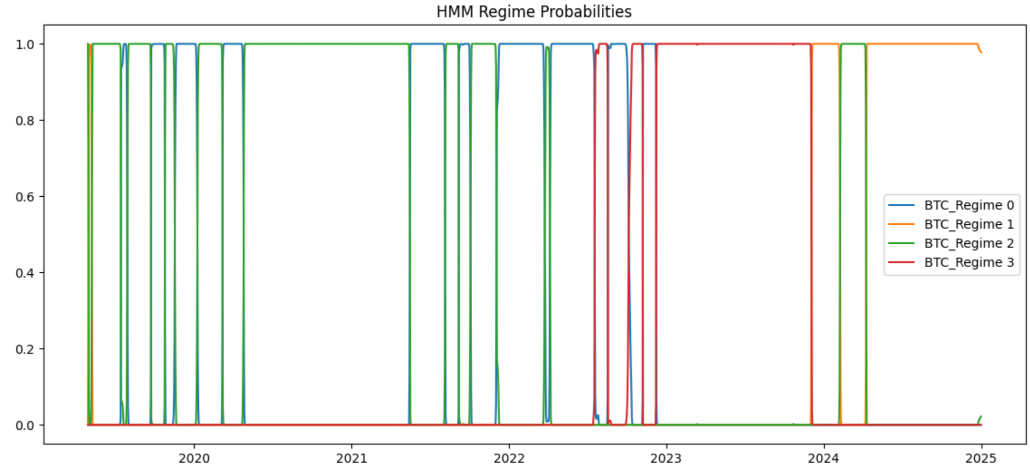
To validate that regimes represent persistent market states rather than noisy day-to-day clustering, the transition matrix was extracted from the fitted model (4 table.). The diagonal entries are very high (approximately 0.98–0.99 for all states), implying strong state persistence.

4 table. *Estimated HMM regime transition matrix*

	to_0	to_1	to_2	to_3
from_0	0.981811	$< 10^{-10}$	0.012709	0.005480
from_1	$< 10^{-10}$	0.994050	0.005950	$< 10^{-10}$
from_2	0.010602	0.002641	0.986756	$< 10^{-10}$
from_3	0.004802	0.002367	$< 10^{-10}$	0.992832

3.5.7 Regime probabilities and confidence

In addition to discrete state assignments, the model was used to compute state membership probabilities. The resulting probability trajectories (12 figure.) are largely close to 0 or 1 for extended periods, suggesting that the inferred regimes are typically assigned with high confidence. Occasional transitions are visible as probability cross-overs, indicating periods of regime uncertainty or switching.

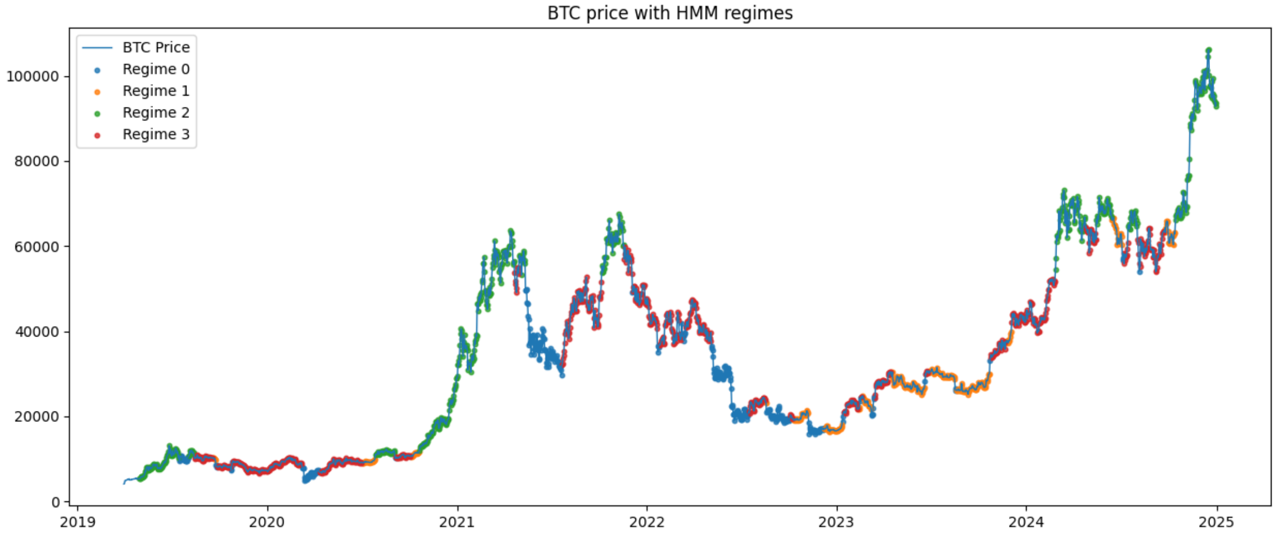


12 figure. *Regime probabilities and confidence*

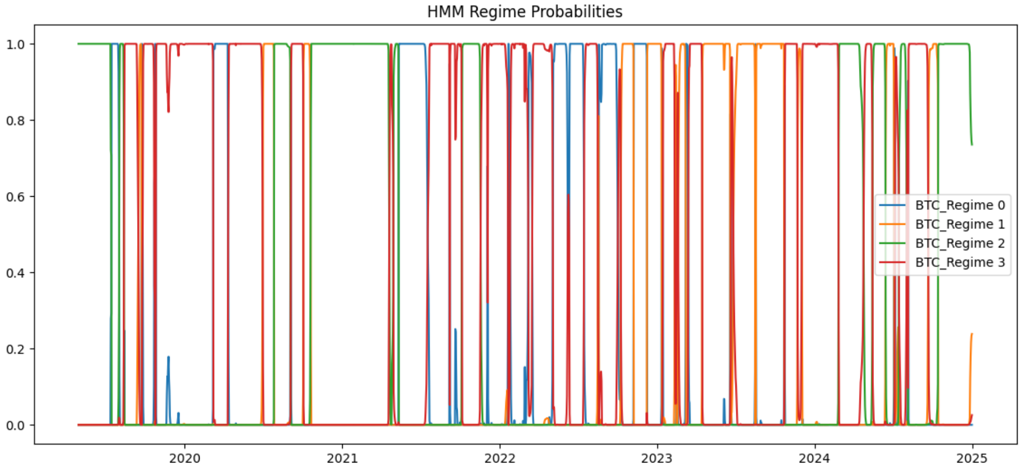
3.5.8 Importance of On-chain Data

Including on-chain variables materially changes the behaviour of the HMM regime signal. When on-chain activity is part of the feature set, the regimes form longer, visually coherent blocks on the BTC price trajectory. In 11 figure., regime segments align with sustained market phases rather than flipping during small local movements. The same stability is reflected in the transition matrix (4 table.), where diagonal elements are close to one, indicating strong persistence and therefore longer expected regime durations. In addition, regime membership is typically assigned with high confidence: the posterior probabilities remain close to 0 or 1 for extended periods, with relatively few ambiguous intervals (Figure 12 figure.). Finally, the regime-conditional averages (Table 3 table.) provide a clearer separation of states, which simplifies interpretation and labeling.

When on-chain variables are excluded, the regimes become noticeably more scattered and change too often. This is directly visible in the price overlay plot (Figure 13 figure.), where regime colors switch frequently even during periods where the price path looks continuous and directionally consistent. In practical terms, the HMM is then driven mainly by faster-moving technical signals (returns, volatility, momentum, drawdown, and volume z-scores), so short-term fluctuations can trigger repeated reclassification. The regime probability plot shows the same issue: probabilities repeatedly cross over and spike, meaning that regime membership is reassigned frequently rather than remaining stable (Figure 14 figure.).



***13 figure.** Bitcoin Prices with Regime Overlay without On-chain*



***14 figure.** Regime probabilities and confidence without on-chain*

Overall, the on-chain variables act as an anchoring information source that reduces regime fragmentation. They provide a network-level activity signal that is less sensitive to day-to-day noise than purely price-derived features, which stabilizes both the hard regime labels and the posterior probabilities. As a result, the HMM with on-chain inputs produces regimes that are more persistent, less prone to rapid switching, and more clearly interpretable compared to the specification that relies only on market-derived indicators.

3.6 Model evaluation setup

All predictive models are evaluated using a chronological (time-based) split to reflect a realistic forecasting scenario. The dataset is divided into three non-overlapping blocks: a training period (used for parameter estimation), a validation period (used for model selection and early stopping), and a final test period (used only for the final out-of-sample evaluation). The target variable is the next-day

Bitcoin log-return, defined as:

$$y_{t+1} = \ln\left(\frac{C_{t+1}}{C_t}\right),$$

where C_t is the daily closing price.

Model quality is reported using standard error measures for regression and a directional metric:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}, \qquad \text{MAE} = \frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t|,$$

$$\text{DA} = \frac{1}{n}\sum_{t=1}^{n}\mathbb{I}\left(\text{sign}(y_t) = \text{sign}(\hat{y}_t)\right),$$

where $\hat{y}_t$ is the model prediction and $\mathbb{I}(\cdot)$ is the indicator function.

For neural network models, input features are standardized using statistics computed on the training set and then applied to validation and test sets. This avoids information leakage from future periods into the training pipeline.

3.7 Naive baseline

As a reference point, simple naive predictors are used to quantify the difficulty of next-day return prediction. The main naive baseline is the *zero-return* forecast:

$$\hat{y}_{t+1} = 0.$$

In addition, a constant-mean baseline can be considered, where the prediction is the mean return estimated on the training period:

$$\hat{y}_{t+1} = \bar{y}_{\text{train}}.$$

These baselines do not use any features and provide a minimum standard that more complex models should aim to match or improve.

3.8 Long Short-Term Memory (LSTM)

To model temporal dependencies in Bitcoin price dynamics, a Long Short-Term Memory (LSTM) neural network was employed. LSTM networks are a class of recurrent neural networks designed to capture sequential patterns and long-range dependencies in time series data, making them a natural candidate for financial time series modeling.

The model was trained to predict the next-day logarithmic return of the Bitcoin closing price. Log returns were chosen instead of price levels to ensure stationarity of the target variable and to avoid trivial predictability arising from the strong autocorrelation of prices. Each observation was constructed as a fixed-length input sequence consisting of the previous 60 daily observations, corresponding to approximately two months of historical information.

Two alternative feature configurations were considered in order to assess the impact of additional alternative data on LSTM-based forecasting:

- **Baseline feature set:** Bitcoin market and technical variables, including closing price, trading volume, daily log returns, realized volatility measures, momentum, drawdown indicators.
- **Extended feature set:** The baseline features augmented with alternative data sources, including on-chain indicators (active addresses and transaction counts), aggregated sentiment measures derived from textual data, and additional contextual variables.

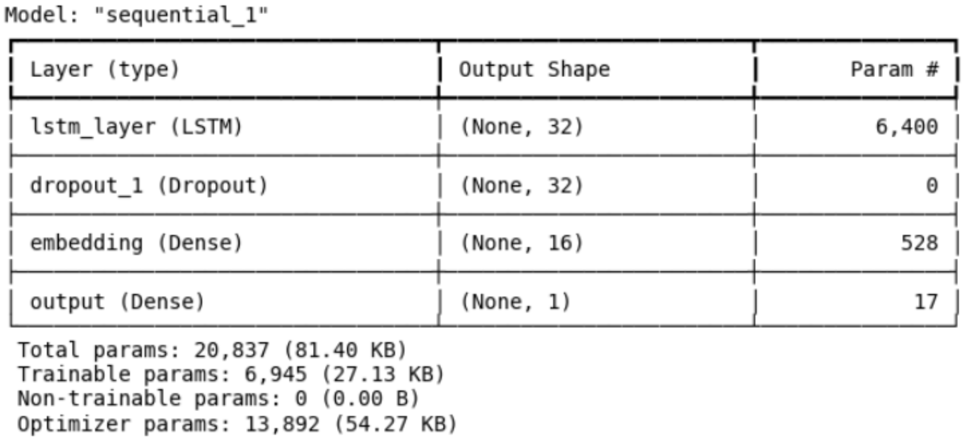
Unlike tree-based models, which benefit from explicit regime conditioning, the LSTM processes temporal dependencies implicitly through its recurrent structure, which may already capture regime persistence through sequential patterns in returns and volatility.

All features were standardized using parameters estimated exclusively from the pre-2024 data to avoid information leakage.

A time-based split was applied to reflect a realistic forecasting scenario. The training period covered 2019–2022, the validation period corresponded to 2023, and the test period consisted of data from 2024. The validation set was used solely for early stopping and model selection, while the test set was reserved for out-of-sample evaluation.

The final LSTM architecture consisted of a single LSTM layer with 32 hidden units, followed by a dropout layer with a dropout rate of 0.2, a dense embedding layer with 16 neurons and ReLU activation, and a linear output layer producing the predicted next-day return. The model was trained using the Adam optimizer with a learning rate of 10^{-3} and the Huber loss function, which is more robust to outliers than mean squared error and well suited to heavy-tailed return distributions. Early stopping was applied based on validation loss, with the best-performing model weights restored.

A schematic overview of the LSTM architecture is shown in 15 figure.



```
Model: "sequential_1"
┏━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━┓
┃ Layer (type)                    ┃ Output Shape           ┃       Param # ┃
┡━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━┩
│ lstm_layer (LSTM)               │ (None, 32)             │         6,400 │
├─────────────────────────────────┼────────────────────────┼───────────────┤
│ dropout_1 (Dropout)             │ (None, 32)             │             0 │
├─────────────────────────────────┼────────────────────────┼───────────────┤
│ embedding (Dense)               │ (None, 16)             │           528 │
├─────────────────────────────────┼────────────────────────┼───────────────┤
│ output (Dense)                  │ (None, 1)              │            17 │
└─────────────────────────────────┴────────────────────────┴───────────────┘
 Total params: 20,837 (81.40 KB)
 Trainable params: 6,945 (27.13 KB)
 Non-trainable params: 0 (0.00 B)
 Optimizer params: 13,892 (54.27 KB)
```

15 figure. *Schematic representation of the LSTM architecture used for next-day Bitcoin return prediction.*

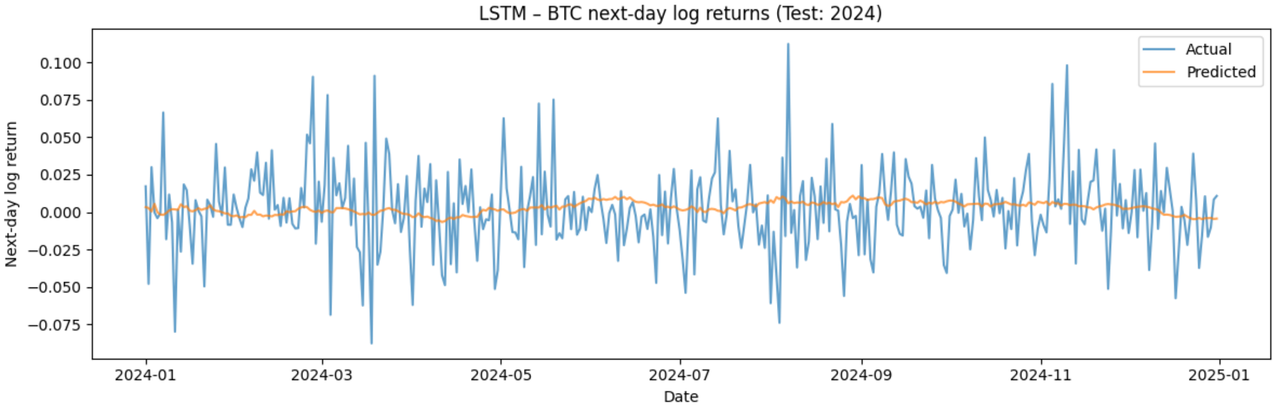
Model performance was evaluated on the 2024 test set using root mean squared error (RMSE), mean absolute error (MAE), directional accuracy, and the correlation between predicted and realized returns. For reference, zero-return prediction was added.

The quantitative results for both feature configurations are summarized in 5 table.

5 table. *Out-of-sample performance of LSTM models on the 2024 test set.*

Feature set	RMSE	MAE	Directional accuracy
Baseline features	0.028700	0.020924	0.479
Extended features	0.027653	0.020064	0.501
Naive baseline (0 return)	0.027563	0.019987	–

Figure 16 figure. illustrates the comparison between actual and predicted next-day returns for the test period.



16 figure. *Actual versus predicted next-day Bitcoin log returns in 2024 using the LSTM model.*

The results indicate that the standalone LSTM model achieves performance comparable to naive benchmarks in terms of RMSE and MAE, which is consistent with the low signal-to-noise ratio typically observed in daily cryptocurrency returns. While the inclusion of alternative data marginally affected directional accuracy, the extended feature set did not lead to consistent improvements in error-based metrics such as RMSE and MAE. This indicates that, for a standalone LSTM architecture, the additional information does not translate into a sufficiently stable incremental signal for direct next-day return prediction, despite modest changes in directional behavior.

Consequently, in this study the LSTM model is not interpreted as a competitive standalone forecasting tool. Instead, its primary role is to act as a sequence encoder, transforming recent market dynamics into a compact latent representation that can be exploited by subsequent tree-based and hybrid models. This perspective motivates the use of LSTM-derived embeddings in later stages of the analysis.

3.9 Light Gradient Boosting Machine (LightGBM)

This study employs LightGBM as the main tree-based benchmark for predicting the next-day Bitcoin log return. The model is implemented using the `LGBMRegressor` interface and is trained under a strictly chronological split to avoid information leakage: the period before 2023 is used for training, year 2023 is used for validation (for early stopping and model selection), and year 2024 is reserved as the final out-of-sample test set. Two configurations are evaluated: (i) a *basic* feature set built only from price/volume-derived variables, and (ii) an *extended* feature set that additionally includes alternative data (on-chain activity, sentiment proxies, macro-financial variables and regimes).

LightGBM is trained with an L1 regression objective to reduce sensitivity to outliers that are typical in cryptocurrency returns. The main hyperparameters are:

- `n_estimators=10000`
- `learning_rate=0.01`
- `num_leaves=15`
- `min_child_samples=50`
- `subsample=0.8`
- `colsample_bytree=0.8`
- `reg_lambda=5.0`

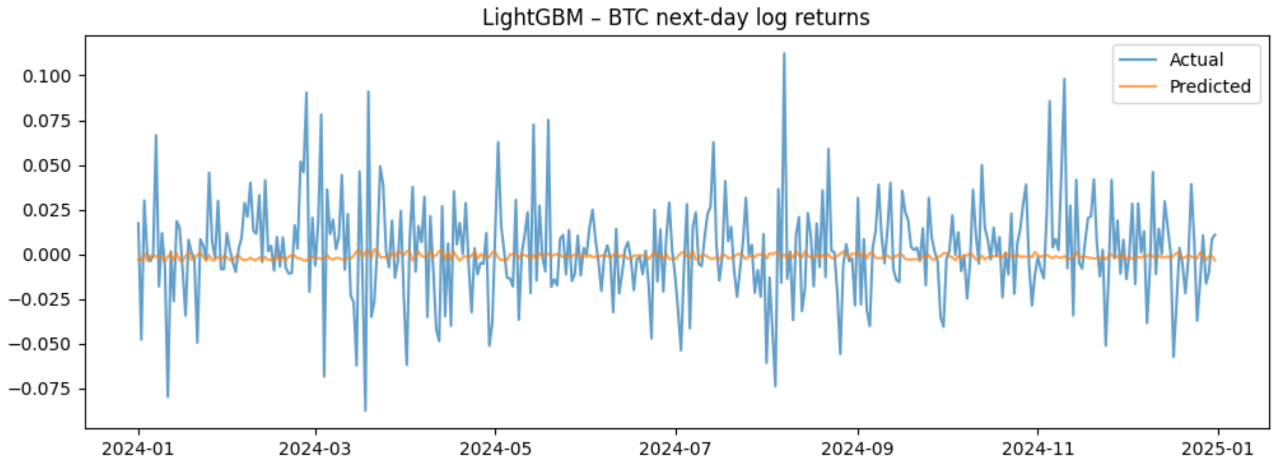
and Early stopping is applied on the validation set with `stopping_rounds=300`, and the selected number of boosting iterations is recorded as `best_iteration`. The market regime indicator (`BTC_regime`) is encoded via one-hot encoding and feature matrices across splits are aligned to ensure consistent column structure.

Model performance is reported on the 2024 test period using RMSE and MAE as point forecast accuracy metrics, and directional accuracy (DA) as a sign-based metric, defined as the share of observations where the predicted and realised returns have the same sign. The naive baseline for comparison is a zero-return predictor.

Table 6 table. summarises performance for both LightGBM configurations. Figure 17 figure. visualises the realised versus predicted next-day returns over the test period to illustrate typical forecast amplitude and tracking behaviour.

6 table. *LightGBM test performance (2024) under two feature configurations.*

Configuration	**RMSE**	**MAE**	**DA**	**Best iter.**
LightGBM (basic)	0.027647	0.020043	0.516	78
LightGBM (extended + alt data)	0.027723	0.020053	0.505	69
Naive baseline (predict 0)	0.027563	0.019987	–	–



17 figure. *LightGBM predictions vs. realised next-day Bitcoin log returns on the 2024 test set.*

3.10 Hybrid model: LSTM embeddings + LightGBM

To combine sequential representation learning with a strong tabular learner, this thesis evaluates a hybrid forecasting approach in which an LSTM network is used as a feature extractor and LightGBM is used as the final supervised predictor. Concretely, the LSTM produces a low-dimensional embedding vector that summarises recent temporal dynamics, and this embedding is concatenated with engineered predictors (return lags, volatility summaries, and selected external signals). The resulting hybrid feature matrix is then used to predict the next-day Bitcoin log return.

The hybrid dataset is assembled from `hybrid_btc_lstm_lightgbm_dataset.csv`, which contains the target variable and LSTM embedding columns (`lstm_emb_*`). Additional predictors are constructed as simple time-series descriptors based on available information at time t, including return lags (e.g., r_t, r_{t-1}), as well as rolling 7-day mean and standard deviation. A consistent chronological split is applied to all hybrid experiments: training up to the end of 2022, validation on year 2023, and out-of-sample testing on year 2024, with the target defined as the next-day log return. This setup ensures that the embedding features and all engineered predictors are evaluated under the same leakage-safe temporal protocol.

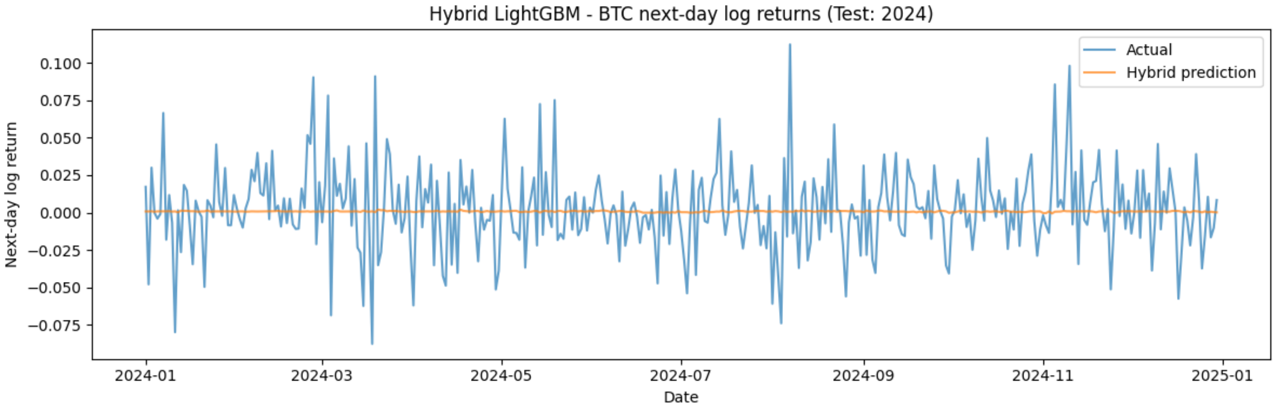
LightGBM is used as the final regression model for two comparable configurations: *embeddings only*, and the *hybrid* configuration that concatenates embeddings and extra features. The LightGBM regressor is trained with a standard squared-error objective and early stopping on the validation set. The main hyperparameters are: `n_estimators=20000`, `learning_rate=0.01`, `num_leaves=63`, `min_child_samples=5`, `subsample=0.8` with `subsample_freq=1`, `colsample_bytree=0.8`, and ℓ_2 regularisation `reg_lambda=0.5`. Early stopping is applied with `stopping_rounds=200`. After selecting the best iteration, the model is refit on the combined training and validation data using the selected number of estimators and then evaluated on the 2024 test set.

Forecast accuracy is assessed using RMSE and MAE on the test set, complemented by directional accuracy (DA) computed as the share of days where the predicted and realised returns share the same sign. Two naive baselines are reported for context: predicting zero return and predicting the training-set mean return. In addition, the variability of predictions is monitored (prediction standard deviation) to verify that the model produces non-degenerate forecasts.

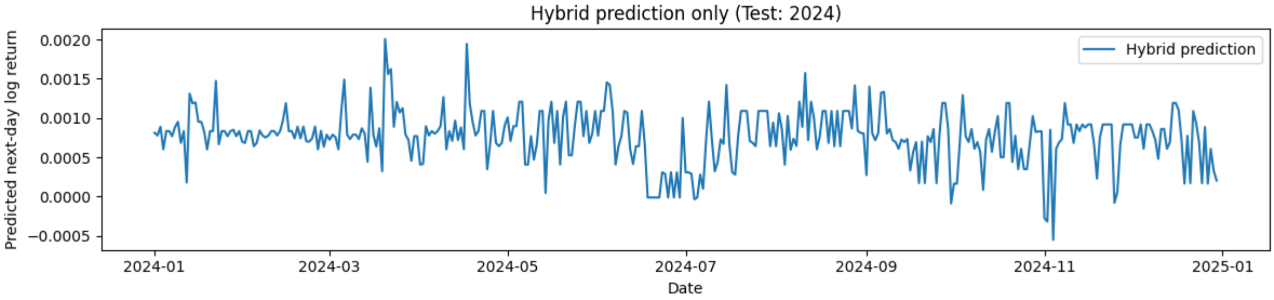
Table 7 table. reports performance for the baselines and the two LightGBM configurations (embeddings-only, and hybrid). Figure 18 figure. visualises the hybrid model's predictions over the 2024 period, 19 figure. shows the same predictions without actual returns ofr readability and Figure 20 figure. summarises the most influential predictors according to LightGBM feature importance.

7 table. *Hybrid model results on the 2024 test set.*

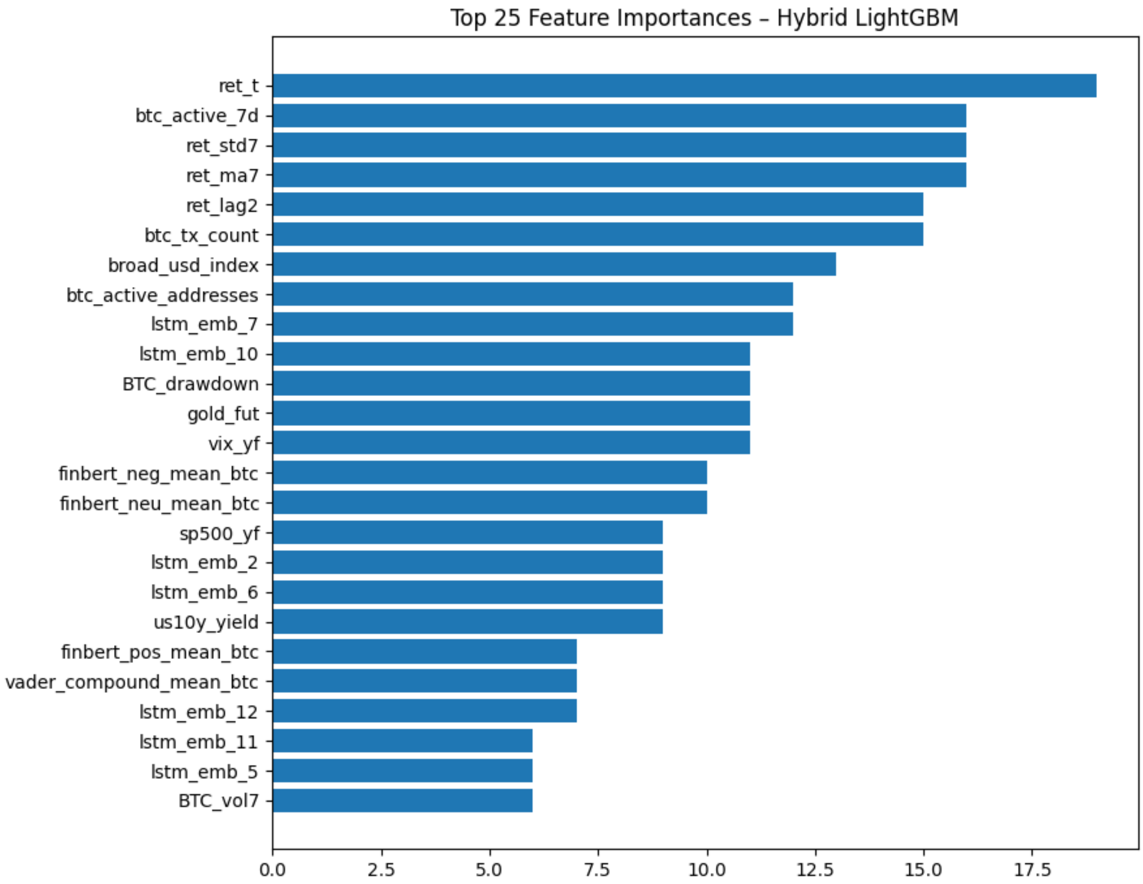
Model	RMSE	MAE	DA
Naive baseline (predict 0)	0.027563	0.019987	–
Naive baseline (train mean)	0.027541	0.019975	0.5288
LightGBM (embeddings only)	0.027509	0.019963	0.529
LightGBM (hybrid: extra + embeddings)	0.027534	0.019960	0.542



18 figure. *Hybrid model predictions vs. realised next-day Bitcoin log returns on the 2024 test set.*



19 figure. *Hybrid model predictions for log returns on the 2024 test set.*



20 figure. *Top-25 feature importances for the hybrid LightGBM model (extra predictors and LSTM embeddings).*

4 Results

This section presents out-of-sample forecasting results for Bitcoin next-day log returns. All models are evaluated under a strict chronological protocol, where model selection is performed on a validation period and final performance is reported on the 2024 test set. Forecast quality is assessed using standard point-forecast error measures (RMSE and MAE) as well as a directional metric (directional accuracy, DA), which measures whether the model correctly predicts the sign of the next-day return.

4.1 Benchmark definition and evaluation metrics

Given the well-documented difficulty of predicting daily cryptocurrency returns, results are interpreted relative to simple baselines. The primary naive benchmark used throughout this thesis is a constant zero-return predictor, which corresponds to forecasting no price change on the next day. As an additional reference point (where applicable), a constant predictor equal to the training-sample mean return is also reported. These baselines provide a conservative yardstick: any consistent improvement over them indicates that the model is extracting at least some predictive structure beyond trivial behaviour.

For all models, RMSE is used to penalise larger errors more strongly, while MAE is reported for robustness and interpretability in the presence of heavy-tailed return distributions. Directional accuracy (DA) complements these metrics by focusing on sign correctness, which is particularly relevant when forecasting results are later discussed from the perspective of trading-oriented decision rules (buy/sell/hold). However, DA is interpreted cautiously because sign prediction near zero returns is inherently unstable and can be influenced by small numerical differences.

4.2 LightGBM results

LightGBM is evaluated in two configurations: a basic specification using only price/volume-derived predictors, and an extended specification that additionally includes alternative data sources (on-chain activity, sentiment indicators, and macro-financial variables). In both cases, the model is trained with an L1 objective and early stopping on the validation set, and the selected number of boosting iterations is recorded as the best iteration.

Table 6 table. summarises test-set performance. Overall, LightGBM achieves a small improvement over the naive baseline in RMSE/MAE, indicating that the model captures limited but non-zero predictive structure in the feature set. At the same time, the magnitude of improvement remains modest, which is consistent with the low signal-to-noise ratio of daily return prediction. Figure 17 figure. provides a visual comparison between realised and predicted returns over the test period; the predictions typically remain close to zero and tend to under-react to large realised return spikes, which is a common outcome in short-horizon financial forecasting when models are optimised for average error.

4.3 Hybrid model results

The hybrid approach combines LSTM-based embeddings with a LightGBM regressor. Both variants follow the same chronological split and are evaluated on the same 2024 test set.

Table 7 table. reports that the hybrid configuration performs comparably to the individual components, with differences across variants being relatively small. This suggests that, under the current modelling assumptions and feature extraction procedure, the embedding representation does not provide a large incremental gain beyond the engineered predictors. Nevertheless, feature importance analysis indicates that the model's predictive decisions rely primarily on short-term return dynamics (lags and rolling volatility summaries), while alternative data variables and selected embedding dimensions still contribute to the fitted ensemble. Figure 18 figure. illustrates the hybrid model's prediction trajectory over the test period.

4.4 Summary of findings

Across all evaluated specifications, the best-performing models demonstrate only incremental improvements over naive baselines. This outcome is not unexpected in the context of daily return forecasting, where returns are close to serially uncorrelated and dominated by exogenous shocks. The empirical results therefore motivate a cautious interpretation: the models appear to capture weak predictive structure that is statistically and practically limited at the daily horizon.

To provide a compact cross-model comparison, Table 8 table. aggregates the main results by model family on the 2024 test set. The table is intended as a high-level summary rather than a full diagnostic report. The "best" model is determined based on the lowest RMSE (with MAE and DA used as supporting indicators). In this study, differences between top candidates are small, and therefore the ranking should be interpreted as marginal rather than decisive.

8 table. *Overview of model-family performance on the 2024 test set. The best model is identified by the lowest RMSE (primary), with MAE and DA reported for additional context.*

Model family	RMSE	MAE	DA
Naive baseline (predict 0)	0.027563	0.019987	Not applicable
LSTM Baseline Features	0.028700	0.020924	0.479
LSTM Extended Features	0.027653	0.020064	0.501
LightGBM Baseline Features	0.027647	0.020043	0.516
LightGBM Extended Features	0.027723	0.020053	0.505
LightGBM embeddings only	0.027509	0.019963	0.529
LightGBM Hybrid (embeddings + extra)	0.027534	0.019960	0.542

Overall, the summary table highlights that model performance is tightly clustered around the naive benchmark, with only minor improvements in error metrics. From a practical standpoint, this suggests that point forecasts of next-day Bitcoin returns have limited standalone usefulness.

To summarize: alternative data provides only marginal improvements and does not consistently outperform simpler market-only specifications at the daily horizon, LSTM embeddings add some value (the embeddings-only LightGBM is best from RMSE standpoint), but the full hybrid model does not improve further beyond embeddings alone. Overall performance stays close to the naive zero-return baseline, confirming that next-day Bitcoin return forecasting is inherently low-signal and difficult to improve meaningfully with point forecasts.

5 Conclusions

This thesis evaluated whether regime awareness, alternative data, and hybrid modeling can improve next-day Bitcoin log-return forecasts under a leakage-safe out-of-sample protocol. Across all tested model families, the results indicate that daily return prediction remains dominated by noise: test-set performance is tightly clustered around the naive baseline that predicts zero return, and incremental gains are small.

From the comparative evaluation, several conclusions follow. First, neither LSTM-based forecasting nor feature-based LightGBM models produce large, robust improvements over the naive benchmark on the 2024 test set, which supports the ideology that short-horizon Bitcoin returns have low predictability in point-forecast terms.

Second, adding alternative data (on-chain, sentiment, and macro-financial variables) does not consistently outperform simpler configurations; improvements, where present, are marginal and should be interpreted as small edges rather than decisive model superiority. Third, learned embeddings are informative: the embeddings-only LightGBM configuration achieves the best overall RMSE among the compared families, but the advantage over the naive baseline remains very small in absolute terms. Fourth, directional accuracy is generally close to 0.5 across models, implying that sign prediction at the daily horizon is unstable and sensitive to small numerical differences around zero returns.

The practical implication is that next-day point forecasts of Bitcoin returns have limited standalone usefulness for aggressive directional trading. Instead, the modeling outputs are more naturally interpreted as components of a risk-aware decision pipeline, where regime conditioning and volatility/risk modeling can constrain actions and reduce sensitivity to noisy return point estimates. In that sense, the thesis contributes an empirically grounded comparison showing where complexity helps only marginally, and where regime-aware interpretation remains valuable even when pure forecast accuracy improvements are small.

References and sources

[1] M. S. Alexander Brauneis. “Crypto Volatility Forecasting: Mounting a HAR, Sentiment, and Machine Learning Horserace.” In: *Asia-Pacific Financial Markets* (2024). https://doi.org/doi.org/10.1007/s10690-024-09510-6. URL: https://link.springer.com/article/10.1007/s10690-024-09510-6.

[2] E. A.-M. Andrés García-Medina. “LSTM–GARCH Hybrid Model for the Prediction of Volatility in Cryptocurrency Portfolios.” In: *Computational Economics* (2024). https://doi.org/doi.org/10.1007/s10614-023-10373-8. URL: https://link.springer.com/article/10.1007/s10614-023-10373-8.

[3] D. Ardia, K. Bluteau, M. Rüede. “Regime changes in Bitcoin GARCH volatility dynamics.” In: *Finance Research Letters* 29 (2019). ISSN: 1544-6123. https://doi.org/https://doi.org/10.1016/j.frl.2018.08.009. URL: https://www.sciencedirect.com/science/article/pii/S1544612318303970.

[4] G. Babaei, P. Giudici, E. Raffinetti. “Explainable artificial intelligence for crypto asset allocation.” In: *Finance Research Letters* 47 (2022). ISSN: 1544-6123. https://doi.org/https://doi.org/10.1016/j.frl.2022.102941. URL: https://www.sciencedirect.com/science/article/pii/S1544612322002021.

[5] A. Bouteska, M. Z. Abedin, P. Hajek, K. Yuan. “Cryptocurrency price forecasting – A comparative analysis of ensemble learning and deep learning methods.” In: *International Review of Financial Analysis* 92 (2024). ISSN: 1057-5219. https://doi.org/https://doi.org/10.1016/j.irfa.2023.103055. URL: https://www.sciencedirect.com/science/article/pii/S1057521923005719.

[6] J. Chaab, G. Zaccour. “Dynamic pricing in the presence of social externalities and reference-price effect.” In: *Omega* 122 (2024). ISSN: 0305-0483. https://doi.org/https://doi.org/10.1016/j.omega.2023.102963. URL: https://www.sciencedirect.com/science/article/pii/S0305048323001275.

[7] H. Ghadiri, E. Hajizadeh. “Designing a cryptocurrency trading system with deep reinforcement learning utilizing LSTM neural networks and XGBoost feature selection.” In: *Applied Soft Computing* 175 (2025). ISSN: 1568-4946. https://doi.org/https://doi.org/10.1016/j.asoc.2025.113029. URL: https://www.sciencedirect.com/science/article/pii/S1568494625003400.

[8] J. W. Goodell, S. Ben Jabeur, F. Saâdaoui, M. A. Nasir. “Explainable artificial intelligence modeling to forecast bitcoin prices.” In: *International Review of Financial Analysis* 88 (2023). ISSN: 1057-5219. https://doi.org/https://doi.org/10.1016/j.irfa.2023.102702. URL: https://www.sciencedirect.com/science/article/pii/S1057521923002181.

[9] P. K. Kopalle, K. Pauwels, L. Y. Akella, M. Gangwar. "Dynamic pricing: Definition, implications for managers, and future research directions." In: *Journal of Retailing* 99.4 (2023). Reinvigorating the store. ISSN: 0022-4359. https://doi.org/https://doi.org/10.1016/j.jretai.2023.11.003. URL: https://www.sciencedirect.com/science/article/pii/S0022435923000544.

[10] A. Ladhari, H. Boubaker. "Deep Learning Models for Bitcoin Prediction Using Hybrid Approaches with Gradient-Specific Optimization." In: *Forecasting* 6.2 (2024). ISSN: 2571-9394. https://doi.org/10.3390/forecast6020016. URL: https://www.mdpi.com/2571-9394/6/2/16.

[11] X. Lin, Y. Meng, H. Zhu. "How connected is the crypto market risk to investor sentiment?" In: *Finance Research Letters* 56 (2023). ISSN: 1544-6123. https://doi.org/https://doi.org/10.1016/j.frl.2023.104177. URL: https://www.sciencedirect.com/science/article/pii/S1544612323005494.

[12] M. M. Mohammad Ali Labbaf Khaniki. "Enhancing Price Prediction in Cryptocurrency Using Transformer Neural Network and Technical Indicators." In: (2024). https://doi.org/https://doi.org/10.48550/arXiv.2403.03606. URL: https://arxiv.org/abs/2403.03606.

[13] D. E. Oluwadamilare Omole. "Deep learning for Bitcoin price direction prediction: models and trading strategies empirically compared." In: 10 (2024). https://doi.org/https://doi.org/10.1186/s40854-024-00643-1. URL: https://link.springer.com/article/10.1186/s40854-024-00643-1.

[14] P. L. Seabe, C. R. B. Moutsinga, E. Pindza. "Forecasting Cryptocurrency Prices Using LSTM, GRU, and Bi-Directional LSTM: A Deep Learning Approach." In: *Fractal and Fractional* 7.2 (2023). ISSN: 2504-3110. https://doi.org/10.3390/fractalfract7020203. URL: https://www.mdpi.com/2504-3110/7/2/203.

[15] soheilrahsaz. *cryptoNewsDataset*. URL: https://github.com/soheilrahsaz/cryptoNewsDataset.

[16] X. Sun, M. Liu, Z. Sima. "A novel cryptocurrency price trend forecasting model based on LightGBM." In: *Finance Research Letters* 32 (2020). ISSN: 1544-6123. https://doi.org/https://doi.org/10.1016/j.frl.2018.12.032. URL: https://www.sciencedirect.com/science/article/pii/S1544612318307918.

[17] Watchful1. "Subreddit comments/submissions 2005-06 to 2024-12." In: (). URL: https://www.reddit.com/r/pushshift/comments/1itme1k/separate_dump_files_for_the_top_40k_subreddits/?.

Appendix 1. Code

The full source code used for this Thesis is uploaded to `Github`. Link: `https://github.com/Lukelis/master-thesis-regime-aware-forecasting/`

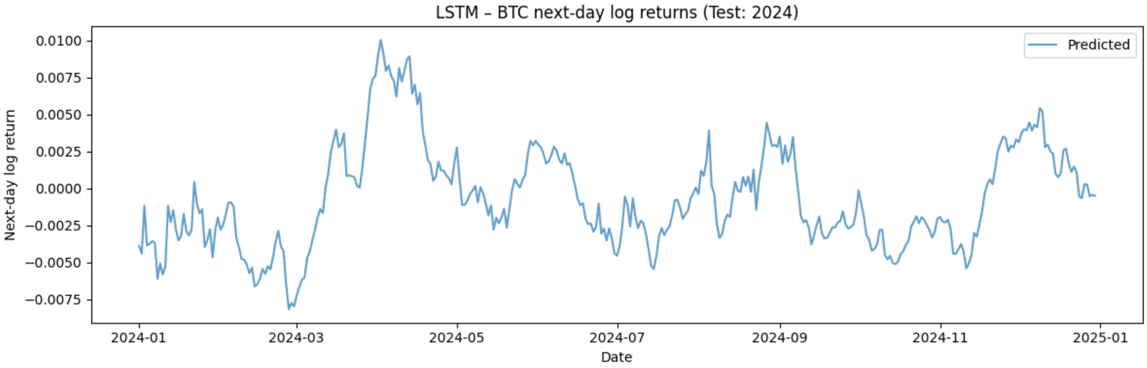
Appendix 2. AI usage

Some parts of the Thesis was updated using Generative AI. All AI outputs were gathered from OpenAI ChatGPT. Generative AI was used for:

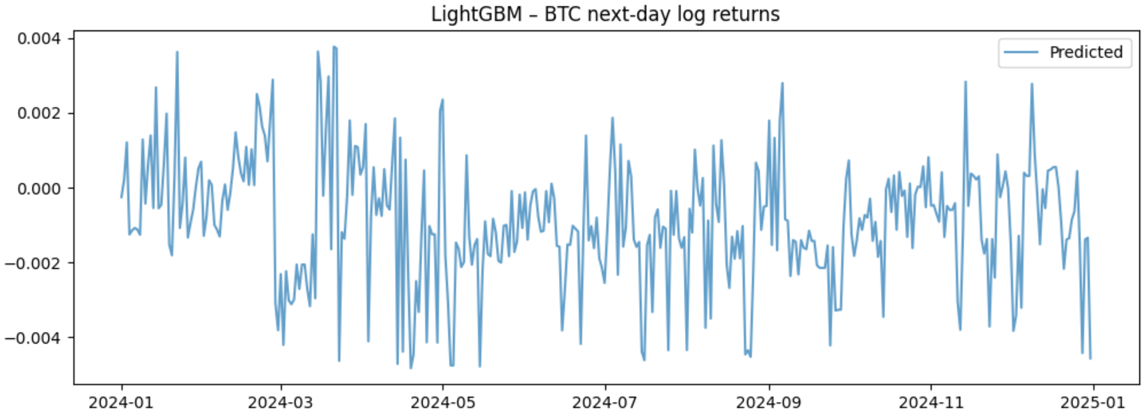
- Text paraphrasing and grammar checking.
- Generating small parts of code.
- Debugging the code.
- Generating list of ban words for Reddit cleaning.

Appendix 3. Graphs

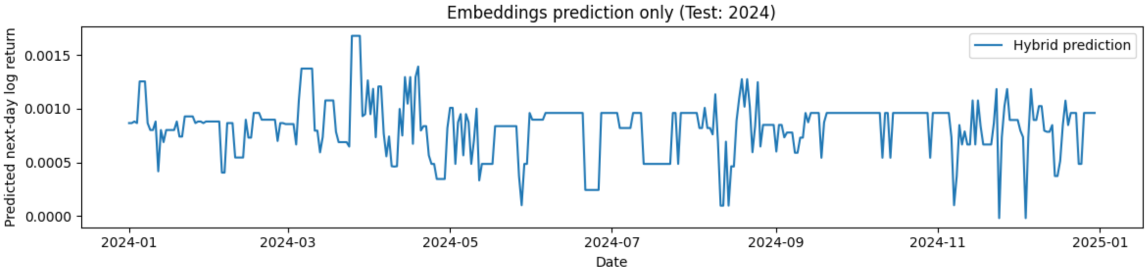
Additional Graphs that were not included in main thesis:



21 figure. *LSTM with OHLCV derived variables only predictions for 2024*



22 figure. *LightGBM with OHLCV derived variables only predictions for 2024*



23 figure. *Embeddings only LGBM prediction for 2024*