Kyle Krug
Northwestern University
MSDS 451
Programming Assignment 1

Problem Description

For Programming Assignment 1, I selected Bitcoin (BTC) as my asset class to evaluate time-series financial data. Bitcoin is a highly liquid and volatile digital asset that trades continuously, making it well-suited for studying short horizon forecasting techniques. The research goal was to predict the direction of the next day's return (up versus down) using binary classification. Working with returns rather than raw prices avoids issues with long-term drift in the price level and provides a more appropriate target for predictive modeling in financial time-series.

Data Preparation and Pipeline

The data for this assignment was downloaded from Kaggle. It was a dataset that contained minute-level Bitcoin trades, which I resampled to daily OHLCV (Open, High, Low, Close, Volume) format covering the period from 2012 to 2025. The resampling process yielded 5,020 daily observations spanning from January 1, 2012, to September 28, 2025. The target variable, "UpTomorrow," equals one if the next day's closing price exceeds today's' closing price and zero otherwise, resulting in a balanced binary classification problem.

I engineered a comprehensive set of features based on standard financial time-series techniques. These included lagged log returns at 1-day and 2-day horizons to capture momentum effects, price spreads such as the high-low range and open-close difference along with their own lags and rolling volatility measures computed over 20-day and 100-day windows to capture regime changes. I also computed a 20-day rolling mean of returns for trend detection, volatility-normalized z-scores to standardize returns by recent volatility, and exponential moving averages of lagged returns using a decay parameter of $\alpha=0.1$ In total, these transformations yielded 23 features that encapsulate various aspects of Bitcoin's price dynamics without introducing look-ahead bias.

To ensure methodological rigor, all features were constructed using only lagged or trailing values, and feature scaling was fit strictly on each training fold before being applied to the corresponding test fold. This design prevents information leakage from future data and maintains the temporal integrity required for time-series forecasting.

As an optional exercise, I incorporated macroeconomic variables from the Federal Funds Rate series. These features included the lagged level of the rate, its volatility-normalized z-score, regime indicators, and changes computed over 7-, 30-, and 90-day windows. This addition evaluated whether monetary policy information adds predictive value to Bitcoin's short-term price movements, given the asset's reputation as a hedge against central bank intervention.

Research Design

I implemented expanding walk-forward cross-validation to evaluate model performance in a manner consistent with real-world trading conditions. The training window began with about two years of historical data (approximately 750 daily observations), then rolled forward in 30-day test blocks while preserving strict temporal ordering. This approach allows the model to adapt to market drift and regime changes while preventing any form of look-ahead bias. The walk-forward methodology is the gold standard for evaluating time-series models in finance, as it simulates the actual experience of deploying a model in production where only past data is available at the time of prediction.

The baseline model was Logistic Regression, selected for its interpretability, computational efficiency, and effectiveness as a linear benchmark. For comparison, I assessed XGBoost, a gradient-boosted tree ensemble capable of capturing non-linear interactions on each fold, and performance was assessed using accuracy and ROC-AUC scores across all test blocks. I also generated visualizations including Bitcoin's historical price trajectory, rolling volatility over time, walk-forward performance metrics across folds, and ROC curves for the final test fold to provide a comprehensive view of model behavior.

Results

Across all folds, Logistic Regression achieved approximately 0.52-0.53 accuracy, with the best grid configuration (using time-decay weighting and k-best feature selection with twenty features) producing a mean accuracy of 0.526 and AUC of approximately 0.51. This performance was statistically above the 0.5 baseline (one-sided p-value = 0.0004), indicating a weak but statistically significant edge. XGBoost produced comparable results with an average accuracy of 0.509 and AUC of 0.504, representing marginally comparable performance to the linear baseline. Both models struggled to meaningfully exceed random guessing, which is consistent with substantial prior evidence that financial markets, and Bitcoin in particular, are highly noisy and difficult to predict at daily horizons. The efficient market hypothesis suggests that publicly available information is already reflected in prices, leaving little predictable signal for models trained on historical price data alone. I do not necessarily prescribe to the efficient market hypothesis, and for future iterations I will seek to add qualitative data such as market sentiment, social media reporting, and government policy decisions which may affect Bitcoin price moves as well.

The performance visualizations revealed that both models achieved slightly better results during periods of stable, lower volatility and underperformed during sharp volatility spikes and regime transitions. This pattern suggests that the models capture weak momentum or mean-reversion signals that are quickly overwhelmed by market noise during turbulent periods. The ROC curves for the final test fold confirmed that neither model achieved strong discriminatory power, with AUC values hovering just above the 0.5 baseline. While the models demonstrated proper implementation of walk-forward validation and feature engineering without data leakage, the marginal predictive edge indicates that additional features, alternative modeling approaches, or longer prediction horizons may be necessary to achieve actionable forecasting performance.

Federal Funds Rate Features

Adding lagged Federal Funds Rate features did not meaningfully improve predictive power for either model. The features included the rate level, its z-score, regime indicators, and multi-horizon changes of 7-, 30-, and 90-day windows. Mutual information analysis showed that these Fed features ranked in the middle of the pack relative to technical indicators, and model runs incorporating them produced AUC values that remained near 0.5. While Bitcoin's long-term narrative is often tied to macroeconomic policy and interest-rate cycles, these results suggest that its short-term daily direction is dominated by intrinsic volatility and market microstructure rather than central bank policy signals. One technical limitation is that many of the Fed rate change features contained frequent zeros (since the rate changes infrequently), which may have diluted their predictive signal. Future work could explore alternative representations of monetary policy stance, such as expectations derived from futures markets or more sophisticated regime indicators, to better capture the relationship between macroeconomic conditions and Bitcoin returns.

Conclusion

This project successfully adapted the WTI crude oil jump-start code to Bitcoin, applied drift-aware feature engineering, and evaluated predictive models using walk-forward cross-validation. The methodology demonstrates sound handling of temporal ordering and data leakage prevention, with proper scaling per fold and no look-ahead bias. The results highlight the substantial challenge of forecasting Bitcoin's daily returns, as even advanced non-linear models like XGBoost showed limited improvement over a simple linear baseline. Both models achieved accuracy and AUC scores barely distinguishable from random guessing, reinforcing the well-documented difficulty of predicting short-horizon price movements in highly efficient and volatile markets.

Despite modest predictive performance, the project successfully demonstrates the core technical skills required for time-series forecasting in finance. The implementation includes proper cross-validation for temporal data, comprehensive feature engineering spanning technical indicators and macroeconomic variables, and rigorous evaluation with appropriate metrics and visualizations. The approach could be extended in several promising directions, including testing alternative prediction horizons (weekly or monthly returns), incorporating sentiment data from news and social media, experimenting with deep learning architectures such as LSTMs or Transformers, or focusing on volatility forecasting rather than directional prediction. This work establishes a robust baseline and methodological framework for future exploration of Bitcoin price dynamics.