STAT 5102
Regression in Practice
Final Project – Group 19

Topic:
Developing a Data-Driven Bitcoin Trading Strategy with Price Trend Forecasting

Group Members:

Name	**SID**
Chan Ka Chun, Rex	1155228490
Chan Siu Pong, Tyler	1155230195
Chan Yun Wa, Arthur	1155226834
Lee Ka Man, Nancy	1155102504
Tang Yuk Shan, Janice	1155228569

Table of Contents

1. Introduction

Objective

This report aims to develop an analytical framework for predicting Bitcoin price movements using a comprehensive set of technical indicators and patterns. The objective is to identify key predictors and patterns influencing medium-term (7-day) Bitcoin price changes, facilitating informed decision-making for cryptocurrency trading and investment.

Summary

The analysis begins with the collection and pre-processing of Bitcoin price data from historical datasets, along with external financial data, including gold prices, risk-free rates, S&P 500 index and ETH prices data. Feature engineering is performed to incorporate technical indicators (e.g., RSI, MACD, Bollinger Bands, ATR) and candlestick patterns (e.g., Ascending Triangle, Bullish Engulfing, Hammer, Three White Soldiers). The data undergoes transformations, including Yeo-Johnson normalization and scaling, to ensure compatibility with predictive models. Machine learning techniques, such as artificial neural networks (ANN), Random Forest, and stepwise regression, are applied to identify key predictors and evaluate model performance.

Upon evaluating various models, the baseline linear regression model achieved the best performance with an R^2 of 0.5791, RMSE of 0.5653, and MAE of 0.4367 for 7-day Bitcoin price changes. However, all models struggled with Bitcoin's volatility, multicollinearity, and dynamic market conditions. Backtesting a hybrid trading strategy, combining model predictions and technical indicators, yielded marginal success but highlighted the limitations of using technical signals alone in volatile markets. The study underscores the challenges of Bitcoin price prediction and the need for robust feature engineering and alternative data sources. Future research should explore using on-chain metrics, social media sentiment, and advanced machine learning models to improve predictive accuracy.

Current Literature

Bitcoin price prediction has been extensively studied in recent years, with researchers exploring the application of machine learning and technical analysis indicators. Abdelatif Hafid et al. (2024)[1] demonstrated the effectiveness of using indicators like RSI and MACD in financial forecasting, while Lahmiri and Bekiros (2019)[2] highlighted the potential of combining technical and fundamental analysis for cryptocurrency markets. However,

---

[1] A. Hafid, M. Rahouti, L. Kong, M. Ebrahim, and M. A. Serhani, "Predicting Bitcoin market trends with enhanced technical indicator integration and classification models," arXiv preprint, arXiv:2410.06935, 2024. [Online].

[2] Lahmiri et al. Cryptocurrency forecasting with deep learning chaotic neural networks Chaos, Solitons Fract. (2019)

most existing studies focus on either technical indicators or external data sources such as gold prices and risk-free rates in isolation, leaving the integration of these data types underexplored. This project seeks to address this gap by combining technical analysis, candlestick patterns, and external macroeconomic data to create a comprehensive predictive framework.

2. Data Collection

The primary dataset comprises historical Bitcoin price and volume data sourced from Investing.com, supplemented with additional financial datasets. All datasets were stored in CSV format and loaded into Collaboratory for processing using the Pandas library.

Dataset	Source	Format	Observations	Variables
Bitcoin Historical Data[3]	Investing.com	CSV	4113	6 (Date, Open, High, Low, Close, Volume, Change)
Gold Price Data[4]	Kaggle	CSV	4113	3 (Date, Gold Price, Volume)
Risk-Free Rate Data[5]	Federal Reserve Economic Data (FRED)	CSV	4113	3 (Date, Risk-Free Rate, Yield Spread)
S&P 500 Index Data[6]	Kaggle	CSV	4113	7 (Date, Open, High, Low, Close, Volume, Adjusted Close)
ETH Historical Data[7]	Kaggle	CSV	4113	3 (Date, ETH Price, ETH Volume)

Each dataset was merged using the date column to ensure temporal alignment for further data processing and feature engineering.

[3] Bitcoin Historical Data

[4] Gold Price and Volume Data

[5] Risk Free Data (US10Y), Risk Free Data (Yield Spread)

[6] S&P 500 Index Data

[7] ETH Historical Data

3. Data Processing

Feature Engineering

The combined dataset spans from Jan 2014 to Feb 2025 of daily observations. Feature engineering was performed to transform raw data into meaningful variables that could be used as predictors. Advanced techniques were employed to calculate technical indicators, detect trading patterns, and create trend-based features. Given Bitcoin's complex and volatile nature, along with its potential relationships with various external factors, we began by exploring a wide range of technical indicators, patterns, and variable combinations, creating around 100 predictors (see appendix). Some examples of predictors are shown below for illustration.

- **Technical Indicators**: RSI, MACD, Bollinger Bands, ATR, VWAP and Stochastic, calculated using standard technical analysis methods. These helped identify overbought or oversold conditions, resistance levels, and buy or sell signals.
- **Macroeconomic Indicators**: Gold Price, Gold Volume, S&P 500 index, US 10 year bond yield, Yield Spread between US10Y and US2Y, Ethereum Price and Ethereum Volume were extracted. These helped identify the correlation of bitcoin price and economic condition.
- **Patterns Detection**: Complex logic was applied to identify patterns like Ascending Triangle, ADX Pattern, Bullish Engulfing, Bearish Engulfing, Hammer, and Three White Soldiers/Black Crows. These required multiple conditions on price movements, shadows, and body sizes across consecutive days.
- **Target Variables**: 7-day percentage price changes, the target variable selected for this analysis is the 7-day percentage price change. This time frame was chosen based on the hypothesis that there exists a delay or reaction period in the Bitcoin market, as well as in related markets such as gold and equities. By focusing on a 7-day window, we aim to capture the effects of market dynamics and inter-market correlations that may influence Bitcoin price movements. This approach allows for a more nuanced understanding of the temporal relationships between Bitcoin and other financial instruments.

Data Cleaning

The raw dataset, consisting of Bitcoin historical data and external financial data, underwent comprehensive cleaning to enhance its consistency and usability.

- **Volume Cleaning**: The Volume column, originally stored as strings with suffixes ('K', 'M', 'B'), was converted to numeric values. A custom function removed commas, interpreted suffixes, and scaled values appropriately (e.g., "1.5M" to

1,500,000). Furthermore, since the Bitcoin volume spikes in 2021, we have narrowed our analysis to start from May 2022.

- **Handling Missing Data**: Missing data in some technical indicators, such as RSI_12 and EMA_14, arises because these indicators require a minimum of a certain consecutive days of price data for calculation. Since the missing values are confined to the beginning and end of the dataset, they were removed to ensure the accuracy and consistency of further analysis without impacting the overall integrity of the data.

Data Transformation

Several transformation techniques were applied to prepare the dataset for predictive modelling, ensuring compatibility, normalising data distributions, and extracting meaningful features.

- **Yeo-Johnson Transformation**: Bitcoin's price exhibits significant volatility and has increased dramatically from 2014 to 2025, making the price data extremely right-skewed. To address this, the **Yeo-Johnson Transformation** was applied to the distribution of the target variable, Pct_Change_7D, into a normalized target variable, YeoJohnson_Pct_Change_7D. This method effectively handles both positive and negative values, ensuring the transformed data is closer to a normal distribution without losing its underlying structure.
- **Normalization and Scaling:** All features were then standardized using StandardScaler to scale them to a comparable range. This step is crucial for machine learning algorithms like gradient-based models or clustering algorithms that are sensitive to feature magnitudes.

These transformations ensured the dataset was optimized for both regression and classification tasks, improving the predictive performance of the models.

4. Exploratory Data Analysis

Before diving into a detailed analysis, conducting exploratory analysis helps us to identify key patterns, trends, skewness and anomalies within the data. This preliminary step allows us to understand the underlying structure and relationships.

Univariate Analysis

To examine the distribution of our variables, we use histograms to assess data skewness and the spread of the variables. After applying standard scaling, most of our variables did not exhibit a symmetric distribution, except few variables such as RSI, Price_vs_MA50, Body (Candle Body data), and Yield_Spread.

Bivariate Analysis

To examine the relationship between our variables, we use scatter plots to visualise patterns and potential correlations. We have separated our visualizations between binary and continuous variables. However, both types of plots show no significant linear correlation with our target variables. Additionally, the correlation matrix indicates high multicollinearity between some variables, such as positive DI and RSI.

5. Model Building

Several machines learning algorithms, including linear regression, artificial neural networks (ANN), and random forest, will be adopted. To enhance the model, **interaction terms** based on domain knowledge will be implemented. These include combinations such as *ADX with Triangle Patterns, ATR with Bollinger Bands, RSI with Stochastic Oscillator, MACD with Volume and Support/Resistance, Macro Indicators with SPY, and Gold with ETH*. Additionally, to ensure a fair comparison, the dataset will be split into 70% for training, 15% for validation, and 15% for testing.

Model 1: Linear Regression - Baseline Model

The first model adopted is linear regression, chosen for its simplicity in modelling the relationship between bitcoin price changes in 7 days and all predictors and interaction terms (n=701). This approach seeks to find the best-fitting line that minimizes the sum of squared errors between predicted and actual values.

The baseline model explains 67% of price changes with a statistically significant F-statistic, confirming the collective importance of the predictor set. Assumption Tests revealed residuals with zero mean and adequate linear fit but exhibited heteroskedasticity, autocorrelation, and mild non-normality in the tails, alongside severe multicollinearity among predictors.

For the In- and Out-of-Sample Performance, although R^2 declines modestly from 0.666 (train) to 0.579 (test), the large MAPE on the training set reflects volatility when percentage changes approach zero. We will revisit the choice of error metrics after pruning redundant predictors.

Sample	RMSE	MAE	R^2	MAPE (%)
Train	0.6058	0.4648	0.6660	1,584.0
Test	0.5653	0.4367	0.5791	355.3

While linear regression is straightforward, effective, and highly interpretable, its robustness is worth considering. To enhance the model, we further refined the model by backward elimination, removing variables with high VIF values to reduce multicollinearity, and stepwise selection.

Model 1.1: Linear Regression - Reduced Model

The reduced model aims to harness the predictive power of linear regression while addressing multicollinearity issues present in the baseline model. This is first achieved by eliminating variables with a Variance Inflation Factor (VIF) greater than 10, and followed by backward elimination algorithm, removing the least significant variable at each step until all remaining predictors had p-values below 0.05.

Initial VIF filtering (threshold=10) reduced predictors from 240 to 89, alleviating severe collinearity. Backward elimination retained 21 predictors, yielding $R^2 = 0.212$

Model 1.2: Linear Regression - Stepwise Model

The stepwise model combines forward selection and backward elimination techniques to iteratively add and remove predictor variables based on their statistical significance. Starting with the feature list generated by VIF selection, predictors were added and removed based on their p-values threshold (0.05) This aimed to create a model that balances complexity and predictive power, retaining only the most significant variables.

Stepwise selection chose 19 predictors but produced a slightly lower R^2 of 0.205. Because backward elimination delivered the higher explanatory power, we adopted the Reduced Model.

Model 1.3: Principal Component Analysis

The PCA model seeks to overcome multicollinearity while reducing the dimension of the predictor and preserving predictive information by transforming the original variables into uncorrelated components.

Figure 1 shows the first 20 principal components explain ~80% of predictor variance. Regressing the transformed returns on PC1–PC20 yields a low R^2 (0.112), with some PCs being significant, it underperforms the earlier reduced linear model in explanatory power. The reduced linear model is better than the PCA model.

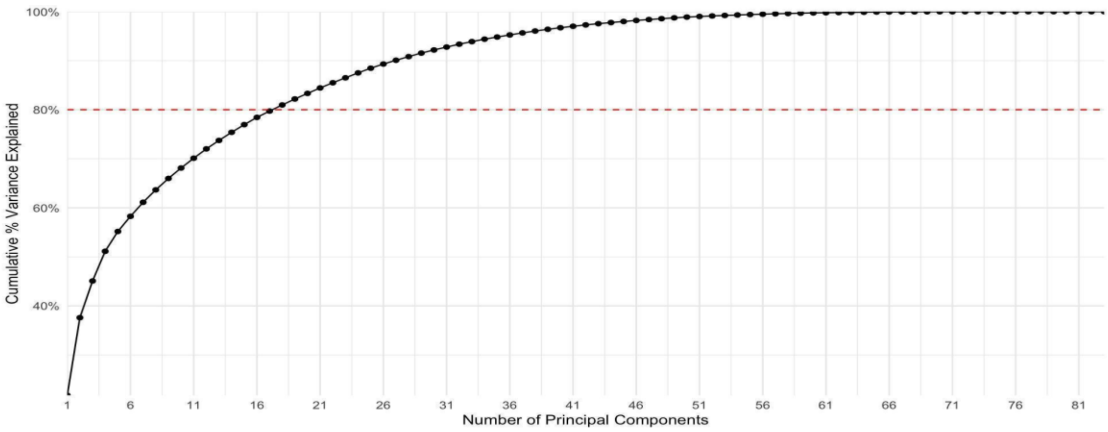
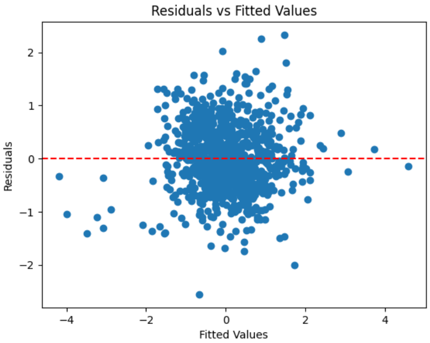
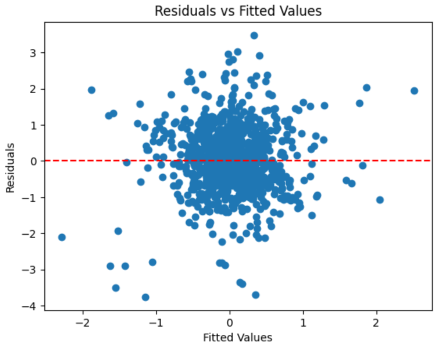
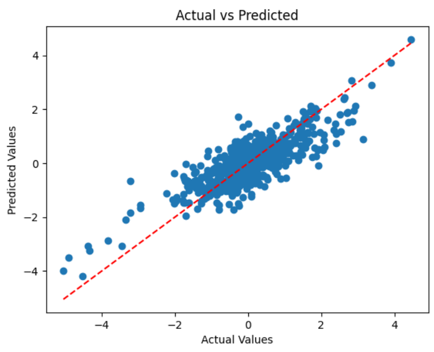
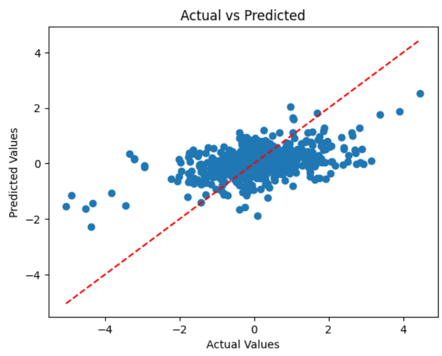
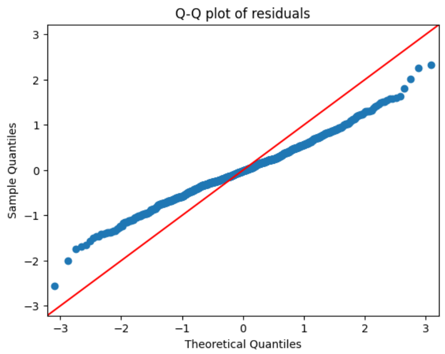
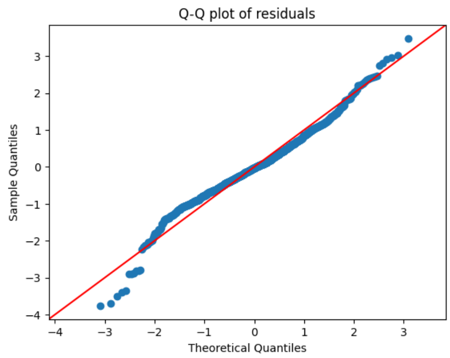


Figure 1: Principal Components % Variance Explanation Chart

Robustness checking

To ensure the robustness of the linear regression model, it's important to check for several key assumptions such as the mean of residuals, heteroscedasticity, linearity, autocorrelation and multicollinearity.

Result (plot if needed)	Baseline model	Reduced model
Mean of residuals	0	0
Heteroscedasticity	p-value: 0.0118 f-value: 1.5166 	p-value: 0.0000 f-value: 4.21 

Linearity		
Residuals Normality		
Autocorrelation	Durbin-Watson statistic: 0.859	Durbin-Watson statistic: 0.396
Multicollinearity	Drop VIF > 10	Nil

- **Mean of residuals:** For baseline and reduced models, the mean of the residuals is zero, indicating that the model's predictions are evenly distributed around the actual values.
- **Heteroscedasticity:** To assess the presence of heteroscedasticity, we conducted a Breusch-Pagan test. The results for both models showed a small p-value and a relatively high f-value, indicating that we can reject the null hypothesis of homoscedasticity with statistical significance. This suggests that the variance of the residuals is not constant across all levels of the independent variables.
- **Linearity:** To verify the linearity between the dependent and independent variables, we examined a scatter plot of the residuals versus the predicted values. The plot indicates that the residuals for the baseline model are mostly aligned in a linear pattern, suggesting a good fit. In contrast, the residuals for the reduced model show a poorer alignment, indicating that the reduced model does not perform as well as the baseline model in maintaining linearity.
- **Normality of the residuals:** To determine if the residuals of the regression model are normally distributed, we examined a Q-Q plot, which compares the quantiles of the residuals to the quantiles of a normal distribution. The visualization indicates that the residuals for the baseline model do not follow a normal

distribution. However, the reduced model shows a better alignment with the normal distribution, suggesting an improvement in the normality of the residuals.

- **Autocorrelation:** The Durbin-Watson statistic for the baseline regression model is 0.8587. This value is significantly less than 2, indicating the presence of positive autocorrelation in the residuals. Positive autocorrelation means that the residuals are correlated with each other, which can affect the reliability of the model's statistical inferences. In contrast, the reduced model has a Durbin-Watson statistic of 0.396, suggesting an even stronger presence of positive autocorrelation in the residuals.
- **Multicollinearity:** To detect multicollinearity, we use the Variance Inflation Factor (VIF). The VIF measures how much the variance of a regression coefficient is inflated due to multicollinearity. We drop the VIF value greater than 10 as the first step to build a reduced model.

Model 2: Artificial Neural Network (ANN)

The ANN model aims to find the best model by iterating through multiple training attempts to find the best model based on Mean Squared Error (MSE). In our implementation, we set the maximum number of training iterations 10,000 and 5 training attempts.

Figure 2 shows that across hidden-layer sizes 2–20, the training R^2 (blue) climbs gently as the network gains capacity, but the test R^2 (orange) actually peaks at size 3—after which it collapses into large negative values, signalling severe overfitting. Likewise, Figure 3 shows that the generalization gap ΔMSE (green) is smallest at size 3 and then balloons as the hidden layer grows. Therefore, a 3-unit hidden layer optimizes the bias-variance trade-off, maximizing test R^2 without overfitting noise.

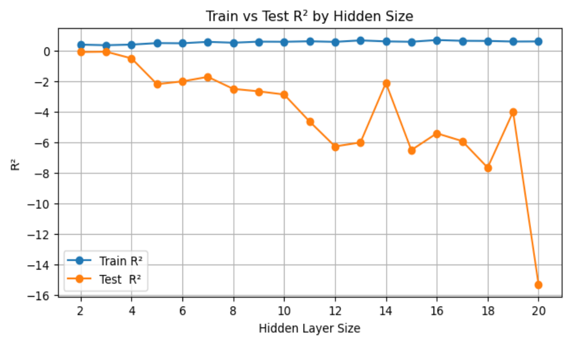


Figure 2: Train vs Test model error rate change by Hidden Size

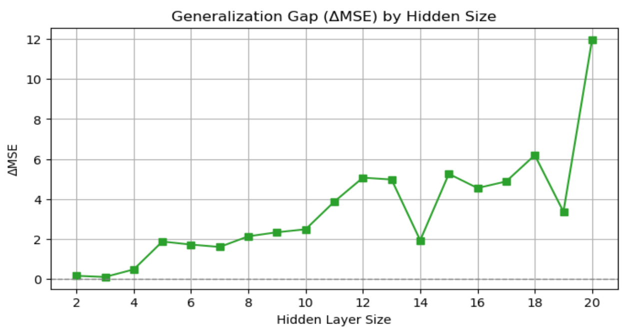


Figure 3: Generalization Gap by Hidden Size

Although the 3-node network generalizes consistently (train vs. test RMSE differ by only ~0.06), its in-sample R^2 of 0.37 indicates it explains less than 40 % of the variation—and the negative test R^2 shows it performs worse than a constant-mean predictor out of sample. In other words, the model is underfitting: it lacks the capacity or structure to capture the underlying patterns. The inflated MAPE values simply reflect the instability of percent-error metrics when actual returns cluster near zero. Overall, this 3-unit ANN does not outperform the linear baseline, and further architecture tuning or richer features will be required to improve predictive accuracy.

Sample	RMSE	MAE	R^2	MAPE (%)
Train	0.8293	0.5876	0.3741	343.8
Test	0.8934	0.6778	−0.0509	201.5

Model 3: Random Forest

The random forest model aims to find the best model by iterating through multiple training attempts to minimize the Mean Squared Error (MSE). In our implementation, we used a RandomForestRegressor with 200 trees, a maximum depth of 10, and other specified hyperparameters to ensure robust performance. This ensemble approach combines multiple decision trees, with each tree trained on a bootstrapped sample of the data with a random subset of features, providing a mechanism to handle the multidimensional aspects of Bitcoin price data, aiming to capture complex non-linear relationships in the highly volatile cryptocurrency market.

- **Feature Importance:** Our model has identified 185 features with positive importance scores out of 269 original features. From below figure 4, it shows the top 15 features in predicting bitcoin price and 'MACD_cum_sell_x_Volume_Ratio' is the most important feature, followed by 'OBV' and 'Price_vs_MA200_x_Support'. The analysis reveals that a majority of the top 15 features are interaction terms, with volume-related metrics appearing in multiple high-ranking features. Technical indicators including MACD and support/resistance levels also demonstrate significant importance in the model. The predominance of interaction terms among top features suggests the model is detecting conditional relationships between technical indicators that single indicators alone cannot capture. Volume-related metrics appear repeatedly in high-ranking features, suggesting trading volume plays a particularly important role in price formation.

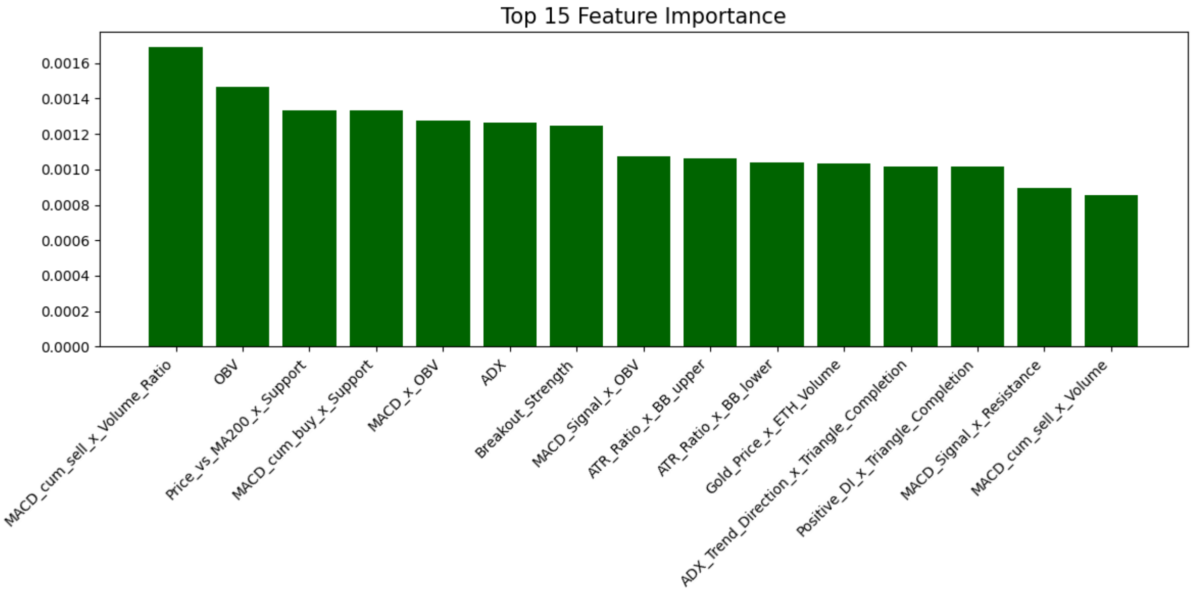


Figure 4: Top 15 Feature Importance

- **Model Validation and Hyperparameter Tuning:** To improve the predictive power, we performed model validation and hyperparameter tuning to optimize the performance of our Random Forest model. Using a grid search approach with time-series cross-validation, we evaluated multiple combinations of hyperparameters, including the number of trees (n_estimators), maximum depth (max_depth), and feature selection criteria (max_features). After evaluation, the optimal parameters included 300 trees (increased from the initial 200), a maximum depth of 10, min_samples_leaf of 4, min_samples_split of 2, and max_samples of 0.8 with log2 for feature selection. These parameters strike a balance between model complexity and generalization ability, with the higher n_estimators providing more robust ensemble predictions.

- **Performance Analysis:** Our model performance metrics show a significant drop in predictive accuracy when moving from training to test data. The negative R-squared value on test data is particularly concerning, as it indicates that the model underperforms compared to simply predicting the mean value. This substantial training-test performance gap highlights the fundamental difficulty in cryptocurrency price forecasting and suggests potential overfitting. These results underscore the inherent unpredictability of Bitcoin prices using our current feature set and modelling approach.

Sample	RMSE	MAE	R^2	MAPE (%)
Train	0.9982	0.6951	0.0933	248.6
Test	1.1269	0.8572	-0.6721	488.7

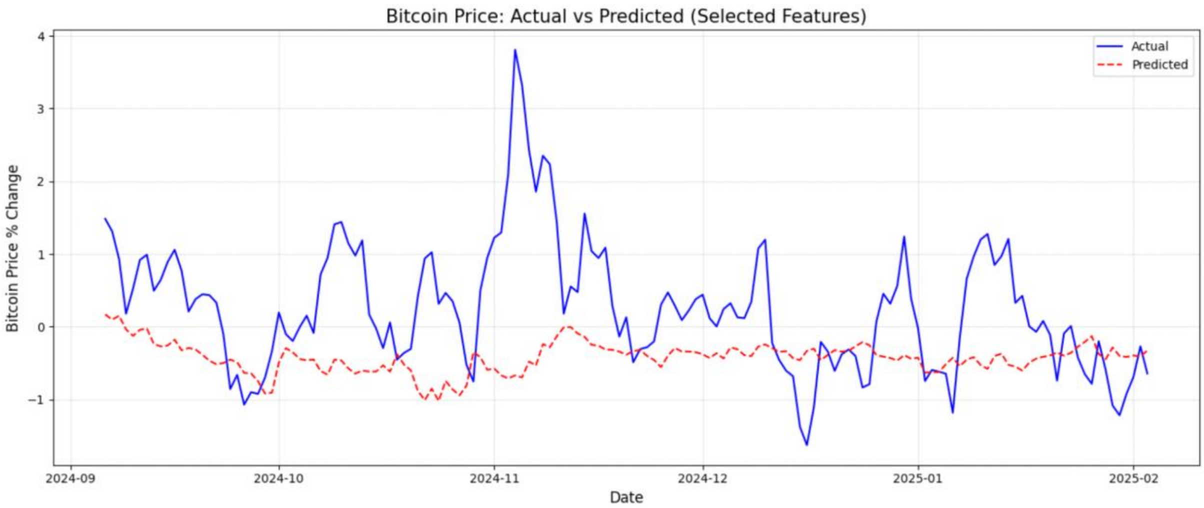


Figure 5: Bitcoin Price: Actual vs Predicted (Baseline Model)

- Figure 5 shows the model tends to produce compressed predictions that fail to capture the magnitude of Bitcoin's price volatility. While the model occasionally aligns with directional movements, it consistently underestimates the scale of significant price changes, particularly during periods of extreme market movement. Most notably, the model struggles with rapid price accelerations, often predicting more modest, mean-reverting price behaviour while actual prices demonstrate substantial volatility.
- The observed results might be due to the weaknesses of random forest ensemble averaging effect, poor extrapolation beyond training data, inability to model temporal patterns, and failure to adapt to changing market conditions, which explain its negative R-squared performance and highlight fundamental challenges in using technical indicators with tree-based models for cryptocurrency forecasting.

6. Model Performance Comparison

	Testing Data			
Model	**RMSE**	**MAE**	**R^2**	**MAPE (%)**
Linear Regression (Baseline)	0.5653	0.4367	0.5791	355.3042
Reduced Model (VIF)	0.8149	0.6371	0.1256	255.7304
Reduced Model (Backward Elimination)	0.8499	0.6532	0.0489	204.3612
Reduced Model (Stepwise)	0.8537	0.6464	0.0404	194.0169
Reduced Model (PCA)	0.9487	0.7304	-0.1851	279.9510
ANN	0.8934	0.6778	-0.0509	201.5402
Random Forest Regressor	0.8943	0.6746	-0.0531	135.6446
Tuned Random Forest Regressor	1.1269	0.8572	-0.6721	488.6793

Model Evaluation and Results

In our study, we compared the performance of various machine learning models to predict Bitcoin price changes, and the evaluation metrics were used RMSE, MAE, R-squared and MAPE.

The Linear Regression (Baseline) model demonstrated the best overall performance with the lowest RMSE (0.5653) and MAE (0.4367) and the highest R-squared (0.5791), indicating stronger predictive accuracy than other approaches. The performance metrics across our 8 modelling approaches showed consistent prediction difficulties for Bitcoin price movements. From the baseline Linear Regression to the Tuned Random Forest models, we observed varying levels of accuracy, with more complex models showing declined performance. The Tuned Random Forest Regressor registered a negative R-squared (-0.6721), indicating it performed worse than simply predicting the mean value.

Model Selection

Based on these results, the Linear Regression (Baseline) model has been selected as our final model for the implementation of our trading strategy due to its performance in predictive accuracy, and this will be examined in the next section. However, even this best-performing model showed limitations, with high MAPE values reflecting the challenging nature of Bitcoin price prediction. The difference between training and testing performance also indicates that historical patterns in this market may have limited predictive power for future price movements.

Challenges in Bitcoin Price Prediction and Modelling Limitations

Our attempts to improve model performance through dimension reduction (PCA) and various feature selection methods (calculated VIF, backward elimination, stepwise regression) did not yield improved results. The reduced models consistently underperformed compared to the baseline model, suggesting that feature selection approaches may actually remove valuable predictive information in this context.

Our results demonstrate the difficulty in predicting the Bitcoin price trend using regression and machine learning models. The cryptocurrency market exhibits substantial noise and potentially non-linear relationships that resist modelling through conventional technical analysis frameworks. The high MAPE values across all models underscore the magnitude of this challenge, with our tuned Random Forest model performing particularly poorly despite its theoretical advantages in capturing complex patterns.

Additionally, Bitcoin demonstrates regime-shifting behaviour. This dynamic nature appears to confound our models, as evidenced by the deteriorating performance metrics as we move from simpler to more complex approaches. The negative R-squared values in our advanced models suggest they struggle to adapt to these changing market conditions. Despite incorporating sophisticated technical indicators and their interactions, our models still fail to capture essential market dynamics. This suggests that critical exogenous factors driving significant price movements may be missing from our feature set.

Despite these substantial challenges, our analysis provides valuable insights about the relative influence of different technical indicators on Bitcoin price movements. These findings contribute to our understanding of technical analysis applications in cryptocurrency markets and suggest potential directions for future research that might incorporate additional data sources or alternative modelling approaches.

7. Back-testing

The trading strategy for predicting Bitcoin combines a hybrid approach of buy-and-hold, active trading, and profit and loss risk management. Given the recent lack of investor confidence in the stock market (2025), reflected by a continuous drop in the long-short term bond yield spread (US10Y vs US2Y), we are optimistic that capital will flow from the stock and bond markets to gold and Bitcoin.

Our strategy starts with a 70% buy-and-hold portion and a 30% active trading portion. Trading signals are generated based on model predictions, requiring a predicted price movement exceeding a 2.0% threshold with a 1.0% confidence margin. Additionally, these signals are validated by at least three technical indicators, such as MA20, Bollinger Bands, ADX, and volume trend. A market regime filter prioritizes long positions in an uptrend (price above MA200) and vice versa.

Active trading position sizing is dynamic, capped at 15% of trading capital, based on confirmation strength and volatility. The profit-to-loss ratio is also considered as part of risk management, with a trailing risk and reward ratio of 1:2 based on a dynamic stop-loss adjusted for volatility. To avoid overtrading, we limit to one open trade and one trade per day.

The strategy tracks portfolio performance by calculating returns, cumulative returns, and drawdowns for trading, buy-and-hold, and total capital, benchmarking against market returns. Assuming an initial capital of US$10,000, the back testing results are as follows:

```
Performance Summary:
Total Return: 2.61% (Market: 46.58%)
Annual Return: 3.18% (Market: 59.25%)
Sharpe Ratio: 0.28
Max Drawdown: -46.18%
Total Trades: 13
Win Rate: 38.46% 5/13
```

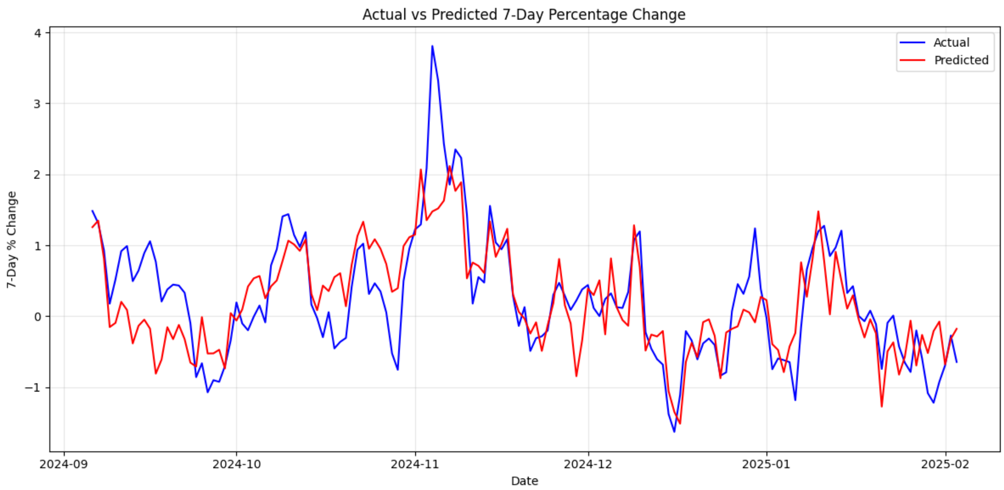


Figure 6: Bitcoin Price: Actual vs Predicted (Backtesting Strategy)

8. Limitations

Dependency in the technical indicators

- **Multicollinearity of predictors:** When predictors are highly correlated, it leads to multicollinearity, which can inflate the variance of coefficient estimates. This makes it difficult to assess the individual contribution of each predictor to the dependent variable. This correlation can significantly distort regression results in the context of technical indicators derived from OHLC data.

- **Violation of regression assumptions:** The assumption of independence among predictors is essential for the validity of hypothesis tests and confidence intervals derived from the regression model. When this assumption is violated, the statistical inferences drawn from the model can be misleading.

Difficulties of the Bitcoin market

- **Higher Volatility Levels:** High volatility can lead to rapid and unpredictable price swings, making it challenging to model and forecast prices accurately. Traditional regression models often struggle to capture such erratic behaviour, leading to less reliable predictions.

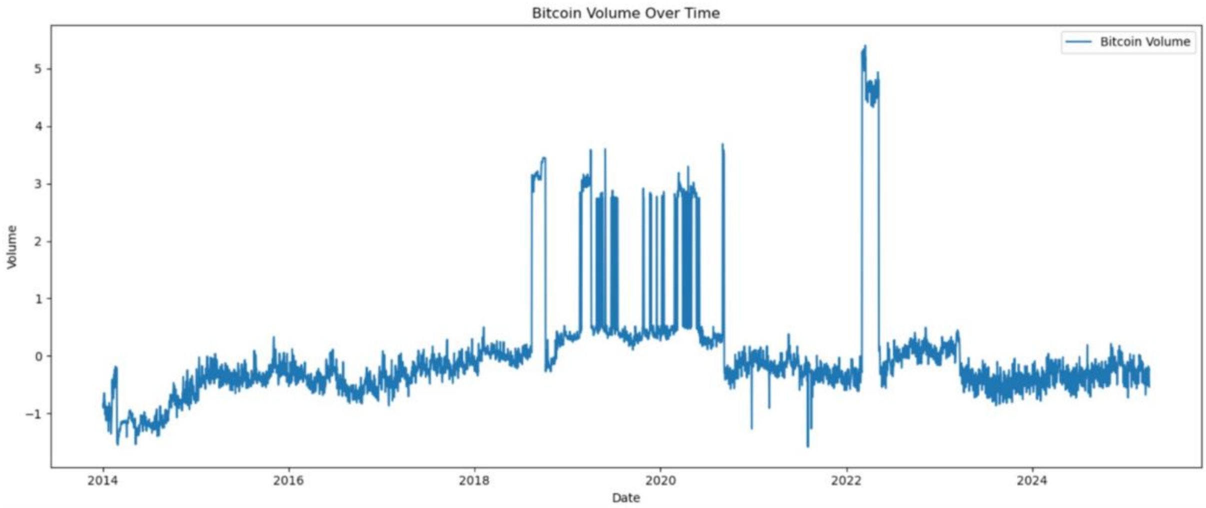


Figure 7: Bitcoin Volume Evolution Chart

- **Market Maturity and Liquidity:** The stock market is generally more mature and regulated, with established mechanisms for price discovery. In contrast, the Bitcoin market is still evolving, leading to increased uncertainty and speculative behaviour.
  Stocks typically have higher liquidity, which can dampen extreme price movements. While Bitcoin is growing, it can still experience significant price changes due to lower liquidity in certain trading conditions.

- **Influence of External Factors:** Bitcoin prices can be susceptible to news, regulatory announcements, and market sentiment, often leading to sudden price changes. For instance, during 2024 November, Bitcoin price surge was Trump re-election and pro-crypto policies[8], by reducing the regulatory burdens. These external factors can introduce noise that complicates predictive modelling.

Sampling size of the technical patterns

Several technical patterns have been examined, including the Cup and Handle pattern and the Three Black Crows pattern. However, these patterns occur infrequently in the Bitcoin market. As a result, their predictive power is limited, making it difficult to justify or predict percentage changes in Bitcoin prices accurately.

[8] Bitcoin heads for nearly 40% November gain as it edges closer to $100,000

9. Conclusion

In conclusion, the predictive power of technical signals and trading patterns appears to be limited, particularly in complex and volatile market environments such as the Bitcoin market.

This limitation underscores the importance of a nuanced approach to market analysis. Relying solely on a one-sided interpretation of technical signals can lead to misleading conclusions, as financial markets are influenced by a myriad of factors, including macroeconomic indicators, market sentiment, and regulatory changes. This complexity extends beyond the Bitcoin market, affecting a wide range of financial instruments.

Utilizing a broader set of predictors may lead to higher accuracy in forecasting models. Machine learning techniques, which can handle large datasets and identify complex patterns, could be particularly beneficial in this regard. These methods allow for the exploration of non-linear relationships and interactions between variables that traditional statistical approaches may overlook. Data such as the ask/bid ratio and tick data possibly bring improvements to the machine learning model.

Future research should explore hybrid modelling approaches that combine technical analysis with alternative data sources such as on-chain metrics, social media sentiment, and order book dynamics. These enriched datasets could be analysed through deep learning models designed for temporal pattern recognition, such as specialized time series models or graph neural networks that capture the interconnected nature of cryptocurrency markets and network effects.

10. Reference

1. A. Hafid, M. Rahouti, L. Kong, M. Ebrahim, and M. A. Serhani, "Predicting Bitcoin market trends with enhanced technical indicator integration and classification models," arXiv preprint, arXiv:2410.06935, 2024. [Online].

   https://doi.org/10.48550/arXiv.2410.06935

2. Lahmiri et al. Cryptocurrency forecasting with deep learning chaotic neural networks Chaos, Solitons Fract. (2019)

   https://sci-hub.se/tree/e8/eb/e8eb8271a05d35739e24a67ac9cca45b.pdf

3. Bitcoin Historical Data
   https://www.investing.com/crypto/bitcoin/historical-data

4. Gold Price and Volume Data

   https://www.kaggle.com/datasets/novandraanugrah/xauusd-gold-price-historical-data-2004-2024

5. US 10 Year Bond Yield Data

   https://fred.stlouisfed.org/series/DGS10

   https://fred.stlouisfed.org/series/T10Y2Y

6. S&P 500 Index Data

   https://www.kaggle.com/datasets/gkitchen/s-and-p-500-spy

7. ETH Price and Volume Data

   https://www.coinlore.com/coin/ethereum/historical-data/-28800/1745769599#reportrange

8. Bitcoin heads for nearly 40% November gain as it edges closer to $100,000
   https://www.cnbc.com/2024/11/29/bitcoin-heads-for-nearly-40percent-november-gain-as-it-edges-closer-toward-100000.html

11. Appendix

List of independent variables

Binary	MACD_buy	1 if MACD crosses above MACD_Signal otherwise 0
Binary	MACD_sell	1 if MACD crosses below MACD_Signal otherwise 0
Binary	MACD_above_signal	1 if MACD > MACD_Signal otherwise 0
Binary	MACD_below_signal	1 if MACD <= MACD_Signal otherwise 0
Binary	BB_Buy	1 if Price crosses below the Lower BB otherwise 0
Binary	BB_Sell	1 if Price crosses above the Upper BB otherwise 0
Binary	Price_below_BB_lower	1 if Price < BB_lower otherwise 0
Binary	Price_above_BB_upper	1 if Price > BB_upper otherwise 0
Binary	VWAP_Buy	1 if Price crosses above VWAP otherwise 0
Binary	VWAP_Sell	1 if Price crosses below VWAP otherwise 0
Binary	RSI_12_Oversold	1 if RSI_12 < 30 otherwise 0
Binary	RSI_12_Overbought	1 if RSI_12 > 70 otherwise 0
Binary	RSI_Divergence	1 if RSI_6 > RSI_12 otherwise 0
Binary	Positive_7D	1 if Pct_Change_7D > 0 otherwise 0
Binary	Positive_2D	1 if Pct Change_2D > 0 otherwise 0
Binary	Higher_Low	1 if current low > previous low otherwise 0
Binary	Breakout	1 if Price > Resistance_Level otherwise 0
Binary	Ascending_Triangle_Breakout	Same as Breakout
Binary	Breakout_With_Volume	1 if breakout with volume confirmation otherwise 0
Binary	Ascending_Triangle_Breakout_With_Volume	Same as Breakout_With_Volume
Binary	ADX_Buy_Signal	1 if ADX > 25 and Positive_DI > Negative_DI otherwise 0
Binary	ADX_Sell_Signal	1 if ADX > 25 and Negative_DI > Positive_DI otherwise 0
Binary	Stochastic_Buy_Signal	1 if Stochastic %K < 20 and %K > %D otherwise 0
Binary	Stochastic_Sell_Signal	1 if Stochastic %K > 80 and %K < %D otherwise 0
Binary	Bullish_Engulfing	1 if Bullish Engulfing pattern is detected otherwise 0
Binary	Bearish_Engulfing	1 if Bearish Engulfing pattern is detected otherwise 0
Binary	Hammer	1 if Hammer candlestick pattern is detected otherwise 0

Binary	Three_White_Soldiers	1 if Three White Soldiers pattern is detected otherwise 0
Binary	Three_Black_Crows recent	1 if Three Black Crows pattern is detected otherwise 0
Binary	Extreme_Fear	1 if Fear and Greed Index indicates extreme fear otherwise 0
Binary	Extreme_Greed	1 if Fear and Greed Index indicates extreme greed otherwise 0
Continuous	Price	Current price of the asset
Continuous	Open	Opening price of the asset
Continuous	High	Highest price of the asset in the period
Continuous	Low	Lowest price of the asset in the period
Continuous	Change	Price change in the period
Continuous	RSI_6	Relative Strength Index with a 6-period length
Continuous	RSI_12	Relative Strength Index with a 12-period length
Continuous	EMA_14	Exponential Moving Average with a 14-period length
Continuous	SMA_14	Simple Moving Average with a 14-period length
Continuous	OBV	On-Balance Volume
Continuous	MACD	Moving Average Convergence Divergence
Continuous	MACD_Signal	Signal line of MACD
Continuous	MACD_cum_buy	Cumulative count of consecutive days where MACD_above_signal is 1
Continuous	MACD_cum_sell	Cumulative count of consecutive days where MACD_below_signal is 1
Continuous	BB_upper	Upper Bollinger Band
Continuous	BB_lower	Lower Bollinger Band
Continuous	BB_cum_Buy	Cumulative count of consecutive days where Price_below_BB_lower is 1
Continuous	BB_cum_Sell	Cumulative count of consecutive days where Price_above_BB_upper is 1
Continuous	ATR	Average True Range
Continuous	ATR_Ratio	ATR as a percentage of Price
Continuous	Stop_Loss_Long	Stop-loss level for long trades
Continuous	Stop_Loss_Short	Stop-loss level for short trades
Continuous	VWAP	Volume Weighted Average Price
Continuous	Support	Lowest low in the past 20 days

Continuous	Resistance	Highest high in the past 20 days
Continuous	Consecutive_RSI_12_Overbought	Cumulative count of consecutive days where RSI_12_Overbought is 1
Continuous	Consecutive_RSI_12_Oversold	Cumulative count of consecutive days where RSI_12_Oversold is 1
Continuous	Pct_Change_7D	Percentage price change over the next 7 days
Continuous	Pct_Change_2D	Percentage price change over the next 2 days
Continuous	MA50	50-day moving average
Continuous	MA200	200-day moving average
Continuous	Price_vs_MA50	Price relative to 50-day MA (%)
Continuous	Price_vs_MA200	Price relative to 200-day MA (%)
Continuous	MA_Distance_Ratio	Ratio of MA50 to MA200 distance
Continuous	Avg_Volume_20	20-day average volume
Continuous	Volume_Ratio	Current volume / 20-day avg volume
Continuous	OBV_Slope	5-day slope of OBV
Continuous	Resistance_Level	Exponential moving average of high price
Continuous	Avg_Volume	7-day rolling average of volume
Continuous	Triangle_Completion	Rolling mean of Higher_Low
Continuous	Triangle_Height	Height of the triangle as % of price
Continuous	Triangle_Duration	Sum of Higher_Low over 7 days
Continuous	Breakout_Strength	Volume-adjusted breakout strength
Continuous	ADX	Average Directional Index
Continuous	Positive_DI	Positive Directional Index
Continuous	Negative_DI	Negative Directional Index
Continuous	Consecutive_ADX_Buy	Consecutive days of ADX_Buy_Signal
Continuous	Consecutive_ADX_Sell	Consecutive days of ADX_Sell_Signal
Continuous	ADX_Strength	How far ADX is above the threshold
Continuous	ADX_Trend_Duration	Consecutive days where ADX > 25
Continuous	ADX_Trend_Direction	Difference between Positive_DI and Negative_DI
Continuous	Stochastic_perK	%K line of the Stochastic Oscillator
Continuous	Stochastic_perD	%D line of the Stochastic Oscillator
Continuous	Consecutive_Stochastic_Buy	Consecutive days of Stochastic_Buy_Signal
Continuous	Consecutive_Stochastic_Sell	Consecutive days of Stochastic_Sell_Signal

Continuous	Consecutive_Hammer	Consecutive hammer patterns
Continuous	SPY_Price	Price of SPY (S&P 500 ETF)
Continuous	SPY_Volume	Trading volume of SPY
Continuous	ETH_Price	Price of Ethereum (ETH)
Continuous	ETH_Volume	Trading volume of Ethereum (ETH)
Continuous	Gold_Price	Price of gold
Continuous	Gold_Volume	Trading volume of gold
Continuous	US_10Y	Yield of the US 10-year Treasury bond
Continuous	Yield_Spread	Spread between different bond yields
Continuous	YeoJohnson_Price	Yeo-Johnson transformed Price
Continuous	YeoJohnson_Change	Yeo-Johnson transformed Change
Continuous	YeoJohnson_7D_Pct_Change	Yeo-Johnson transformed Pct_Change_7D
Continuous	YeoJohnson_2D_Pct_Change	Yeo-Johnson transformed Pct_Change_2D
Continuous	Fear_and_Greed_Index	Fear and Greed Index value
Continuous	Consecutive_Extreme_Fear	Consecutive days of Extreme_Fear
Continuous	Consecutive_Extreme_Greed	Consecutive days of Extreme_Greed