Machine Learning for Daily Return Direction Forecasting: A Comparative Study with Explainable AI Insights

Krzysztof Płachta and Robert Ślepaczuk[1]

[1]University of Warsaw, Faculty of Economic Sciences, Quantitative Finance Research Group, Department of Quantitative Finance

Draft - July 2025

Abstract

This study addresses two interrelated objectives. First, we conduct a comparative evaluation of six supervised learning models: Lasso, Random Forest, LightGBM, LSTM, and two feedforward neural networks, for forecasting the next-day direction of SPY returns. Using a long-horizon, expanding-window backtesting framework, we generate strictly out-of-sample forecasts and assess model performance based on risk-adjusted metrics, with the Sortino ratio as the primary criterion. Second, to address the opacity of complex machine learning models, we apply a model-agnostic explainable AI approach to the best-performing model. This involves systematically removing individual features and predefined feature groups, followed by full retraining with hyperparameter optimization, to quantify each variable's contribution to overall performance. The results were mixed. Lasso, LightGBM, and most notably Random Forest – which achieved a Sortino ratio of 0.61, over 30% higher than the benchmark, and exhibited a substantially lower maximum drawdown – outperformed the buy-and-hold strategy. On the other hand, contrary to expectations, all neural network-based models significantly underperformed. In the second part, feature importance analysis for the Random Forest model suggests that signals based on technical indicators, foreign exchange, and commodity prices consistently contributed to improved performance, while certain interest rate and equity index features were detrimental. These findings emphasize the importance of thoughtful model design and feature selection in the development of machine learning strategies for financial forecasting.

Keywords: machine learning, financial forecasting, algorithmic investment strategies, testing architecture, random forest, neural networks, LSTM, LightGBM, Lasso, explainable AI, feature importance

JEL codes: C45, C53, G11, G17

1 Introduction

Machine learning (ML) has emerged as a powerful tool for financial forecasting, owing to its ability to model complex, non-linear relationships and detect subtle patterns in noisy data. These capabilities are particularly valuable in equity markets, where return predictability is limited, signal-to-noise ratios are low, and interactions among variables are often nonlinear (Cont, 2001; Lo and MacKinlay, 1990). However, valid concerns can be raised that the complexity of many ML models' data processing makes it impossible for humans to understand how exactly they arrive at their prediction and which variables contribute the most to the decision-making process. Furthermore, consistent forecasting success would contradict the Efficient Market Hypothesis (Fama, 1970), which posits that excess returns based on public information should not be attainable in semi-strong-form efficient markets.

This study pursues two core objectives: (1) to assess whether ML models can forecast the next-day direction of S&P 500 returns with sufficient accuracy to yield economically significant, risk-adjusted excess returns, and (2) to analyze model interpretability by identifying which features contribute most to predictive performance, thereby providing insights into the degree to which market-relevant information is priced.

To achieve this, the following research hypotheses will be tested:

H1: ML-based trading strategies generate significantly higher total and risk-adjusted returns than the buy-and-hold benchmark over the backtest period.

H2: The performance advantage of ML-based strategies is more pronounced during bear markets and periods of high volatility.

H3: ML-based strategies reduce overall portfolio risk, as measured by lower return volatility and smaller drawdowns.

H4: Among the tested models, neural networks achieve the highest risk-adjusted performance, consistent with their capacity to capture nonlinear dependencies.

The following methodology will be applied in order to achieve the study goals. We evaluate six supervised learning models: Lasso, Random Forest, LightGBM, LSTM, and two feedforward neural networks - selected to represent a range of linear, tree-based, and deep learning approaches. Their ability to forecast the next-day direction of SPY returns is assessed using a long-horizon, expanding-window backtest with strictly out-of-sample predictions. Model performance is primarily evaluated using the Sortino ratio, complemented by other risk and return metrics. Second, we address the interpretability challenges of complex machine learning models by introducing a systematic, model-agnostic explainability framework. We quantify the impact of individual features and predefined thematic groups by retraining and re-optimizing the best-performing model after stepwise feature exclusion. This approach enables us to identify which market signals are most predictive, offering a more transparent view of model behavior and revealing how different sources of public information contribute to performance. This also provides a practical lens through which to evaluate the Efficient Market Hypothesis.

This paper makes two main novel contributions to the growing literature on machine learning in finance. First, it advances explainable AI in financial forecasting by introducing a dynamic feature attribution framework that moves beyond static, full-sample importance analysis. While prior work such as Gu, Kelly, and Xiu (2020) focuses on the aggregate contribution of individual predictors, our approach extends this by analyzing both thematic feature groups and the evolution of feature importance over time. This temporal decomposition enhances the interpretability of ML model behavior and helps distinguish whether models rely on persistently informative signals or adapt opportunistically to shifting market conditions. Second, our empirical findings yield an interpretable, economically grounded assessment of which publicly available signals - ranging from momentum indicators to macroeconomic variables - consistently contribute to predictive performance. By systematically identifying features that enhance returns versus those that add noise, our study provides a practical diagnostic tool for model design and offers new insights into how information is incorporated into asset prices.

The remainder of the paper is structured as follows. Section 2 reviews the relevant literature. Section 3 describes the data and outlines the feature construction process. Section 4 details the methodological framework, including model selection, hyperparameter tuning, backtesting procedures, and evaluation metrics. Section 5 presents the empirical results, covering both forecasting performance and feature importance analysis. Section 6 concludes and offers directions for future research.

2 Literature Review

The application of machine learning to financial return forecasting has evolved significantly over the past two decades. Early research focused on simple feedforward neural networks and classical ML algorithms applied to directional return prediction. For example, Kara, Boyacioglu, and Baykan (2011) employed artificial neural networks and support vector machines to forecast the direction of the Turkish stock market index based on technical indicators, showing promising predictive performance despite limited sample sizes and basic validation procedures. Similarly, Huang, Nakamori, and Wang (2005) reviewed early applications of ML in stock market prediction, highlighting the use of decision trees and SVMs in the pre-deep learning era.

Chen et al. (2015) applied LSTM networks to forecast returns in the Chinese stock market, demonstrating that recurrent architectures could capture temporal dependencies in price data more effectively than traditional models. The use of LSTMs has since become more widespread, particularly for modeling time-dependent financial data with non-linear dynamics.

Building on this early foundation, more rigorous empirical studies have emerged. One of the most influential is Gu, Kelly, and Xiu (2020), who showed that deep learning models, particularly feedforward neural networks and tree-based ensembles such as random forests and gradient-boosted machines, consistently outperform linear models in forecasting cross-sectional U.S. stock returns. Their framework included robust expanding-window backtesting, realistic investment simulation, and variable importance evaluation using feature dropout.

Our study builds on this methodological foundation and applies it to the problem of forecasting the direction of returns for the SPY ETF. In doing so, we extend the approach to include recurrent neural networks, specifically LSTMs, which are well-suited for modeling temporal dependencies in financial time series (Fischer & Krauss, 2018; Chen et al., 2015). Fischer and Krauss (2018), for instance, demonstrated that LSTMs significantly outperform both traditional econometric models and simpler neural network architectures when predicting daily S&P 500 returns.

Other works have explored ML applications at the index level. Krauss, Do, and Huck (2017) employed deep neural networks and random forests to predict daily returns for S&P 500 constituents, achieving significant improvements over benchmark strategies. Takeuchi and Lee (2013) applied deep learning techniques to momentum-based strategies, demonstrating that neural networks can extract non-linear patterns in price data relevant for short-term trading decisions. In a comparative study, Persio and Honchar (2016) evaluated MLP, CNN, and LSTM architectures for forecasting returns on S&P 500 and EUR/USD, highlighting the suitability of convolutional and recurrent networks for capturing structure in financial time series data.

Our research incorporates explainable AI tools to assess how feature importance translates into trading performance. SHAP (SHapley Additive exPlanations), introduced by Lundberg and Lee (2017), has become a widely adopted, model-agnostic tool for quantifying the contribution of individual features. While SHAP has been applied in financial contexts (e.g., Bussmann et al., 2020), most implementations stop short of linking interpretability results to investment decisions. Our approach addresses this gap by evaluating how feature relevance impacts out-of-sample trading performance across time. This offers a more practical interpretation of feature relevance and contributes to the emerging literature that links predictive accuracy with real-world investment outcomes.

3 Data

The selection of data for the purpose of this study was driven by the aim to balance competing goals and practical limitations. Specifically, our objectives were to:

1. Include economically relevant variables with demonstrated or plausible predictive power for equity returns.
2. Ensure diversity across feature types to enrich the explainability analysis using XAI tools.
3. Facilitate thematic grouping by selecting features that can be logically categorized.
4. Maintain parsimony to ensure computational feasibility while retaining explanatory breadth.

In order to achieve that, we constructed a custom dataset using publicly available data from Yahoo Finance and Stooq, spanning the period from January 2000 to December 2024 at a daily frequency. After data preprocessing and imputation, the final dataset comprises 6,289 daily observations and 20 independent variables, along with the target variable - the next-day return of the SPY ETF. This yields a total of 132,069 data points. A complete list of variables, along with their summary statistics, is provided in Table 1.

Most variables display negative skewness and substantial excess kurtosis, indicating the presence of asymmetric distributions and heavy tails, which is consistent with expectations (Cont, 2001). These deviations from normality are particularly pronounced for the two energy-related variables - oil and natural gas, both of which exhibit very high excess kurtosis, reflecting the frequent occurrence of large price shocks in these markets.

Feature	**Data Type**	**Mean**	**Std Dev**	**Min**	**Max**	**Skewness**	**Exc. Kurtosis**
vol_change	fract. change	0.06	0.37	-0.83	4.64	1.75	8.07
1d_lag	log-rets	0.00	0.01	-0.12	0.14	-0.26	11.17
5d_mom	c. log-rets	0.00	0.02	-0.22	0.18	-0.90	7.37
21d_mom	c. log-rets	0.01	0.05	-0.40	0.22	-1.48	7.17
63d_mom	c. log-rets	0.02	0.08	-0.54	0.34	-1.29	4.17
252d_mom	c. log-rets	0.07	0.17	-0.64	0.57	-1.13	1.75
SSE_CI	log-rets	0.00	0.01	-0.09	0.09	-0.32	6.35
HSI	log-rets	-0.00	0.01	-0.14	0.13	-0.01	8.09
Nik225	log-rets	-0.00	0.01	-0.13	0.10	-0.69	8.06
Rus2000	log-rets	0.00	0.02	-0.15	0.09	-0.49	7.06
13w_TBill	level	0.02	0.02	-0.00	0.06	0.82	-0.77
10y_TNote	level	0.03	0.01	0.00	0.07	0.12	-0.72
VIX	level	0.20	0.08	0.09	0.83	2.21	8.09
yield_spread	level	0.01	0.01	-0.02	0.04	-0.28	-0.74
usdeur	log-rets	-0.00	0.01	-0.03	0.03	-0.02	1.95
usdjpy	log-rets	0.00	0.01	-0.04	0.05	-0.21	4.38
usdcny	log-rets	-0.00	0.01	-0.06	0.05	-0.29	9.86
WTI_c	log-rets	0.00	0.03	-0.29	0.22	-0.29	14.33
Gold_c	log-rets	0.00	0.01	-0.09	0.10	-0.31	5.80
NatGas_c	log-rets	0.00	0.04	-0.35	0.62	0.83	16.61

Table 1: Summary statistics of the explanatory variables and target. The table reports key descriptive statistics for all features used in the model based on daily observations from January 2000 to December 2024 (N = 6,289).

Most of the original data consisted of asset prices or interest rates, which were pre-processed to enhance model convergence. For equities (excluding the VIX), foreign exchange rates, and commodities, log returns were computed to remove long-term trends. In contrast, since the VIX and interest rates typically do not exhibit such trends, they were not differenced but instead scaled down by a factor of 100 to align their magnitudes with the log-return features. Ensuring comparable feature scales was particularly important for neural networks, which are sensitive to input magnitudes due to the way their weights are initialized (Passalis et al., 2021). Missing values for log-return features were imputed with zeros, while variables in levels were forward-filled using the last available observation - effectively assuming zero return when data were unavailable. Finally, all features were aligned such that each predictor is lagged one day relative to the SPY return, ensuring valid out-of-sample prediction.

In addition to macroeconomic and market-based features, several technical indicators were engineered from SPY data. These included the daily lagged return, percentage change in volume (also scaled down by 100), and cumulative momentum calculated as rolling sums of past log returns over 5, 21, 63, and 252-day windows. Lastly, the yield spread was computed as the difference between the 10-year Treasury note and the 13-week T-bill yields.

The independent variables are drawn from five economically motivated categories:

1. Equities: Daily returns of major global equity indices – Shanghai Composite Index, Hang Seng Index, Russell 2000, Nikkei 225, and the CBOE Volatility Index (VIX).

2. Foreign Exchange: Daily returns of the U.S. Dollar against the Euro, Japanese Yen, and Chinese Yuan.

3. Commodities: Daily returns on West Texas Intermediate (WTI) crude oil, natural gas, and gold.

4. Fixed Income: Yields on the 10-year U.S. Treasury Note, the 13-week Treasury Bill, and the spread (difference) between the long and short yields.

5. Technical Indicators: Changes in SPY trading volume, 1-day lagged returns, and rolling momentum indicators over 5, 21, 63, and 252-day windows.

These features were selected based on established findings in financial economics, where variables such as momentum (Jegadeesh & Titman, 1993), macroeconomic indicators, and market sentiment proxies (e.g., VIX) have demonstrated predictive relevance for asset returns.

4 Methodology

4.1 Overview of the methodology

This study uses a two-stage framework to evaluate the predictive performance and interpretability of machine learning models for forecasting the direction of SPY ETF returns. In the first stage, six supervised learning models, representing linear, tree-based, and deep learning approaches, are benchmarked against each other using the same methodology. Each model forecasts the next-day return direction of SPY, and its performance is evaluated using a set of financial metrics, with the Sortino ratio serving as the main benchmark for comparison.

In the second stage, the model that achieved the highest Sortino ratio over the full backtesting period is further examined using explainable AI techniques. The objective is to assess the contribution of individual input features and identify which types of data provide predictive signals about market movements that may not be fully reflected in current prices. This is done by systematically removing variables - both individually and in thematic buckets, re-running the backtests, and comparing changes in performance metrics. This approach offers a practical and data-driven measure of feature importance, as it captures the marginal impact of each variable on real-world trading outcomes under consistent historical conditions.

The remainder of this section is structured as follows: Section 4.2 presents the selected machine learning models and outlines the hyperparameter tuning process. Section 4.3 details the backtesting framework used for out-of-sample evaluation. Section 4.4 introduces the evaluation metrics, including the portfolio construction mechanism. Section 4.5 describes the XAI methodology employed to assess feature relevance.

4.2 Models and hyperparameter tuning

This study evaluates six supervised learning models, representing a diverse range of machine learning techniques: Lasso (Tibshirani 1996), Random Forest (Breiman 2001), LightGBM (Ke et al., 2017), Long Short-Term Memory networks (Hochreiter and Schmidhuber 1997), and two Feedforward Neural Networks (Rumelhart, Hinton, and Williams 1986), one with a single hidden layer and the other with two hidden layers. This selection spans linear models, tree-based ensembles, and deep learning architectures, including both feedforward and recurrent neural networks, ensuring comprehensive coverage of different model types. The neural networks are designed with a pyramidal architecture, wherein each hidden layer contains half the number of neurons as the previous layer. This architectural choice, along with the inclusion of two FNN configurations with varying depths, is informed by prior research. It has demonstrated that the depth of neural networks can significantly affect the model's performance in return forecasting as it impacts both its ability to capture complex data relationships and its susceptibility to overfitting (Gu et al., 2020).

Hyperparameter selection plays a critical role in unlocking the full potential of machine learning models, as it is essential to achieve an optimal fit to the data while avoiding overfitting to noise. This aspect of the study posed significant challenges, particularly in balancing multiple considerations. First, each model has a distinct set of hyperparameters that require tuning, thus it was not possible to simply set the same search space for all models. Second, to ensure fair comparison, it was imperative

to allocate a similar amount of computational resources across all models, avoiding biases that could arise from testing an excessive number of hyperparameter combinations for one model while testing a much smaller set for another. Finally, the optimization process was constrained by computational time limitations, exacerbated by the backtesting procedure, which involved performing 20 hyperparameter selection processes per backtest (as outlined in following section). Therefore, it was nevessary to find a solution capable of efficiently identifying the optimal hyperparameters within a reasonable timeframe.

To address these challenges, Bayesian Optimization was employed, as it offers solutions to most of the outlined problems. It is more computationally efficient than grid search, and studies have demonstrated its superior performance compared to random search (Turner et al., 2021). Additionally, Bayesian Optimization allowed for consistent control over computational time across models by limiting the number of optimization trials for each, ensuring a balanced evaluation. The Tree-structured Parzen Estimator (TPE) algorithm, implemented via the Optuna Python library (Akiba et al., 2019), was utilized for hyperparameter tuning. Each model underwent 50 optimization trials, with the search space for each model defined through expert judgment, tailored to the model's specific structure and complexity. The objective of each tuning run was to maximize the following accuracy-based reward function:

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{1}(\hat{y}_i = y_i), \quad y_i,\ \hat{y}_i \in \{-1, +1\},\ i = 1, \ldots, N. \tag{1}$$

This metric directly reflects the task objective - correctly classifying whether the SPY ETF will close higher or lower in the subsequent period without regard to the magnitude of the price change.

Table 2 summarizes the hyperparameter search spaces used for each model, reflecting their structural and algorithmic differences. For neural networks, the tuning focused on training dynamics and regularization, including learning rate, batch size, dropout rate, and L1 regularization. In contrast, Lasso required tuning only a single regularization parameter (C), resulting in a much simpler search space. Tree-based models involved more complex configurations: Random Forests required optimization over tree depth, the number of estimators, and node-splitting criteria, while LightGBM included a broader set of hyperparameters such as learning rate, number of leaves, and regularization terms.

Model	**Parameter**	**Search Range**	**Sampling**
Neural Networks	learning_rate	$[10^{-4}, 10^{-2}]$	Log-uniform
	batch_size	$\{16, 32, 64, 128, 256\}$	Categorical
	dropout_rate	$\{0, 0.1, \ldots, 0.5\}$	Categorical
	l1_reg	$[10^{-6}, 10^{-2}]$	Log-uniform
Lasso	C	$[10^{-5}, 10^{3}]$	Log-uniform
Random Forest	n_estimators	$[100, 500]$	Integer (uniform)
	max_depth	$[3, 25]$	Integer (uniform)
	min_samples_split	$[2, 15]$	Integer (uniform)
	min_samples_leaf	$[1, 10]$	Integer (uniform)
LightGBM	n_estimators	$[100, 1000]$	Integer (uniform)
	learning_rate	$[0.01, 0.3]$	Log-uniform
	max_depth	$[3, 20]$	Integer (uniform)
	num_leaves	$[7, 255]$	Integer (uniform)
	min_data_in_leaf	$[10, 100]$	Integer (uniform)
	feature_fraction	$[0.5, 1.0]$	Uniform
	bagging_fraction	$[0.5, 1.0]$	Uniform
	bagging_freq	$[1, 10]$	Integer (uniform)
	lambda_l1	$[0, 10]$	Uniform
	lambda_l2	$[0, 10]$	Uniform

Table 2: Unified hyperparameter search spaces for all models. Sampling strategies include uniform (continuous or integer), log-uniform (log scale), and categorical (discrete sets).

These differences illustrate why a uniform search strategy across models would have been inappropriate. Instead, model-specific search spaces were defined through expert judgment, balancing

comprehensiveness with efficiency. Sampling strategies, ranging from uniform to log-uniform and categorical, were selected to reflect each parameter's practical sensitivity and to limit the number of possible configurations. This careful design was essential to ensure a consistent and fair comparison: it allowed equal computational resources to be allocated across models despite their differing complexities, mitigating the risk of biased performance outcomes due to uneven optimization effort.

4.3 Backtesting framework

To evaluate model performance under realistic trading conditions, we employ an expanding-window, walk-forward backtesting framework that guarantees strictly out-of-sample forecasts, thereby eliminating look-ahead bias. The framework is applied uniformly to all six models described in Section 3.2.

We utilize twenty years of daily SPY ETF data, which encompasses both market crises and prolonged bull markets in order to ensure robustness across diverse market regimes. The first five years of observations (approximately 1260 trading days) were used exclusively for model training. In the first run, the data is split into a four-year training set and a one-year validation set for hyperparameter optimization. Once optimal hyperparameters are selected, the model is retrained on the full five-year window of in-sample data. That model is then used to obtain out-of-sample forecast for all trading days in the next calendar year - starting from 2005. Each forecast is done by passing data available for t = n, to make prediction about the direction of returns on SPY at t = n+1. Afterwards, the model is discarded and an additional year of data is added to the in-sample set, and the process is repeated on the expanded dataset. This process is summarized in the flowchart presented in Figure 1.

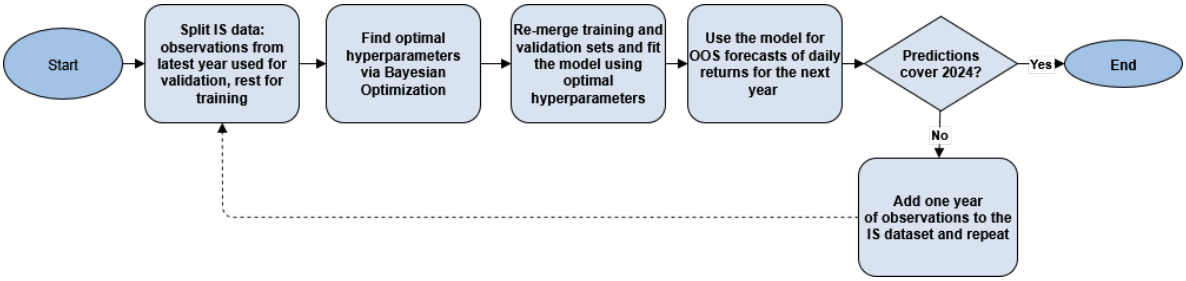

Figure 1: Flowchart of the expanding-window walk-forward backtesting procedure.

The annual re-estimation frequency balances model adaptability and computational cost. While more frequent updates (e.g., daily or monthly) might offer marginal gains, they would significantly increase runtime; less frequent updates, in contrast, may underperform in dynamic markets. No universally accepted update cadence exists, yet our choice aligns with prevailing practice and maintains tractable compute requirements. A similar trade-off governs the one-year validation window: shorter windows may overfit transient noise, whereas longer windows may fail to capture regime shifts promptly. This configuration reflects a trade-off supported by prior literature and ensures consistency throughout the study.

4.4 Evaluation Metrics

To assess the practical relevance of model forecasts, we implement a simplified trading strategy based on the directional predictions of each model. The strategy begins with an initial capital of $\omega_0 = 1$, which is fully allocated at each market close to either a long or short position in the SPY ETF, depending on the sign of the predicted return. To focus purely on the predictive quality of the signals, the strategy assumes zero transaction costs, including slippage and bid–ask spreads. While this assumption may lead to performance estimates that exceed what would be achievable in real-world settings, the deviation is likely limited. The trading frequency is relatively low, keeping cumulative transaction costs modest, and high-frequency execution near the close would minimize slippage. More importantly, the objective of this study is not to propose a deployable trading strategy, but rather to compare the effectiveness of different forecasting models and to explore their decision-making processes under controlled conditions.

Under these assumptions, the evolution of the strategy's equity value is governed by the following dynamic:

$$\omega_{t+1} = \omega_t \cdot (1 + \text{pred}_t \cdot r_{\text{SPY},t}), \tag{2}$$

where $\text{pred}_t \in [-1, 1]$ denotes the model's directional prediction for day t, and $r_{\text{SPY},t}$ is the realized daily return of the SPY index.

For each model, a separate equity curve is constructed based on its directional predictions. These strategies are then evaluated using a comprehensive set of performance metrics that capture not only cumulative returns but also risk-adjusted performance, drawdowns, and directional accuracy. The same metrics are also computed for a buy-and-hold strategy, which serves as a benchmark.

While multiple evaluation criteria are reported, model rankings are based exclusively on the Sortino ratio (Sortino and Price, 1994), which measures excess return relative to downside deviation. This metric is selected because it more accurately reflects risk-adjusted performance from an investor's perspective: unlike the Sharpe (Sharpe, 1994) or Information ratios, it penalizes only downside volatility, ignoring positive deviations that are typically desirable. An alternative approach would be to focus purely on predictive accuracy, measured by hit rate. However, this wouldn't capture the economic value of the forecasts. A model that occasionally misses minor fluctuations but successfully anticipates larger, more consequential price moves would be more valuable to investors than one that excels at minor directional accuracy alone.

Lastly, although the primary evaluation is based on the full 20-year out-of-sample period, we also segment the analysis into shorter subperiods to assess the consistency of strategy performance across different market environments. The specific performance metrics and their corresponding formulas are provided below.

- **Annual Returns Compounded:**

$$\text{ARC} = \left(\frac{\omega_T}{\omega_0}\right)^{1/T} - 1, \tag{3}$$

  where T is the total number of years in the backtest period, in this case 20.

- **Maximum Drawdown (MDD):**

$$\text{MDD} = \min_t \left(\frac{\omega_t}{\max_{s \le t} \omega_s} - 1\right), \tag{4}$$

- **Annualized Volatility:**

$$\sigma_{\text{ann}} = \sqrt{252} \cdot \text{StdDev}(r_t^{\text{strat}}). \tag{5}$$

- **Sharpe Ratio:**

$$\text{Sharpe} = \frac{\text{ARC} - \bar{r}_f}{\sigma_{\text{ann}}}, \tag{6}$$

  where $\bar{r}_f$ is the average yield on 10-year U.S. Treasury notes.

- **Sortino Ratio:**

$$\text{Sortino} = \frac{\text{ARC} - \bar{r}_f}{\sigma^-_{\text{ann}}}, \tag{7}$$

  with σ^-_{ann} representing the annualized downside deviation computed from negative daily returns.

- **Information Ratio (IR):**

$$\text{IR} = \frac{\text{ARC}}{\sigma_{\text{ann}}}, \tag{8}$$

- **Hit Rate:**

$$\text{Hit Rate} = \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}[p_t = \text{sign}(r_t)], \tag{9}$$

  representing the proportion of correct directional forecasts.

4.5 XAI

Following the identification of the best-performing model, we conducted an explainability analysis to evaluate the relative importance of individual input features. To this end, we adopted a model-agnostic approach, whereby features were systematically removed from the input set, and the model was retrained and backtested to assess changes in key performance metrics. This analysis was conducted in two formats: (i) feature-wise, where each of the 20 input variables was excluded individually, resulting in 20 separate backtests; and (ii) category-wise, where entire feature groups-as defined in the data section-were removed, producing 5 additional backtests.

An important consideration in this study is the effect of dataset modification on hyperparameter selection. Since the backtesting framework involves re-optimizing hyperparameters whenever a feature or feature group is removed, changes in model performance may reflect not only the importance of the excluded feature but also differences in the resulting model configuration. This is a natural consequence of data-dependent optimization procedures.

Moreover, the machine learning models employed in this research inherently involve stochastic elements, such as random weight initialization in neural networks and bootstrap resampling in ensemble methods. These elements introduce variability into the training process that will persist when the training data is altered even when random seeds are fixed. As such, observed changes in performance metrics reflect a combination of the direct impact of feature removal and the broader sensitivity of the models to stochastic training dynamics.

Despite this inherent variability, the adopted methodology remains informative. Rather than attempting to isolate deterministic feature effects in a static model, the objective is to assess the robustness of feature contributions across realistic, data-driven model retraining cycles. This approach more accurately reflects the conditions under which predictive models are developed and deployed in practice. While the magnitude of performance changes may vary, consistent patterns in these results offer meaningful insights into which features contribute substantively to predictive accuracy and risk-adjusted returns.

5 Results

5.1 Comparison of models' performances

Table 3 presents a comparative analysis of all evaluated models based on key performance indicators: return, risk, and directional accuracy. The results indicate that non-neural network-based models consistently outperformed the passive buy-and-hold (b&h) strategy across most performance metrics, particularly in terms of risk-adjusted returns.

Model	ARC (%)	Cum. Ret. (%)	MDD (%)	# of Trades	St. Dev. (%)	Sharpe ratio	Sortino ratio	IR	Hit Rate (%)	Hit Rate $> 1\sigma$ (%)	Hit Rate $> 2\sigma$ (%)
rf	12.13	885.7	-33.2	1008	19.05	0.48	0.61	0.64	53.90	50.3	46.89
lgbm	11.19	733.8	-32.6	1550	19.06	0.44	0.57	0.59	52.55	51.59	49.79
lasso	10.58	646.2	-36.3	279	19.06	0.40	0.49	0.55	54.64	50.26	43.57
b&h	10.30	609.8	-55.2	1	19.06	0.39	0.46	0.54	55.04	50.15	41.91
nn2	8.91	450.5	-51.7	930	19.06	0.32	0.39	0.47	53.82	49.43	44.40
nn1	5.21	175.8	-59.6	583	19.06	0.12	0.15	0.27	53.07	48.51	45.64
lstm	1.06	23.3	-73.1	165	19.07	-0.10	-0.12	0.06	52.87	48.20	45.23

Table 3: Performance comparison of trading strategies based on machine learning model forecasts. The table reports key performance metrics including Annualized Return Compounded (ARC), cumulative returns, maximum drawdown (MDD), number of trades executed, standard deviation of returns, Sharpe and Sortino ratios, Information Ratio (IR), and hit rates, including thresholds above one and two standard deviations. Results are benchmarked against a buy-and-hold strategy (b&h).

Among the evaluated models, Random Forest (rf) delivered the strongest overall performance,

achieving a Sortino ratio of 0.61, which is over 30% higher than the benchmark Sortino of 0.46. LightGBM followed closely with a Sortino ratio of 0.57, while Lasso regression was the third and final model to outperform the benchmark, exceeding it by 6.5% and achieving a Sortino ratio of 0.49. This result aligns with the conclusions of Navani and Giri (2024), who also found LightGBM and Random Forest consistently outperform the traditional buy-and-hold strategy in both profit generation and drawdown reduction, especially under volatile conditions. In contrast, all neural network-based models significantly underperformed. The best among them, the two-layer feedforward neural network (nn2), lagged behind the benchmark by 15.2%, while the worst performer, the Long Short-Term Memory (lstm) model, delivered negative excess returns. This finding is surprising, because it contradicts both hypothesis H4 and prior empirical findings, such as Gu et al. (2020). Potential explanations for this underperformance are discussed later.

A more nuanced picture emerges when examining performance over shorter subperiods. Table 4 (presented in the Appendix) reports key metrics for all models over subsequent four-year intervals. Since 2009, the buy-and-hold (b&h) strategy has consistently yielded strong returns, with ARC ranging from 13.55% to 15.90%, and delivered the highest risk-adjusted return during 2009–2012. In contrast, model performances exhibited greater variability; for instance, Random Forest (RF) achieved a remarkable 22.29% ARC between 2017 and 2020, followed by a sharp decline to 2.44% in 2021–2024.

Models that ultimately outperformed the benchmark generated strong returns during the initial 2005–2008 period, which encompasses the global financial crisis. Notably, in the most recent period (2021–2024), the two-layer feedforward neural network (nn2) was the best-performing model, suggesting that neural networks may retain potential in highly nonlinear or evolving market regimes. These results support the hypothesis that machine learning models are particularly effective during market stress, even if their performance is less consistent during bull markets.

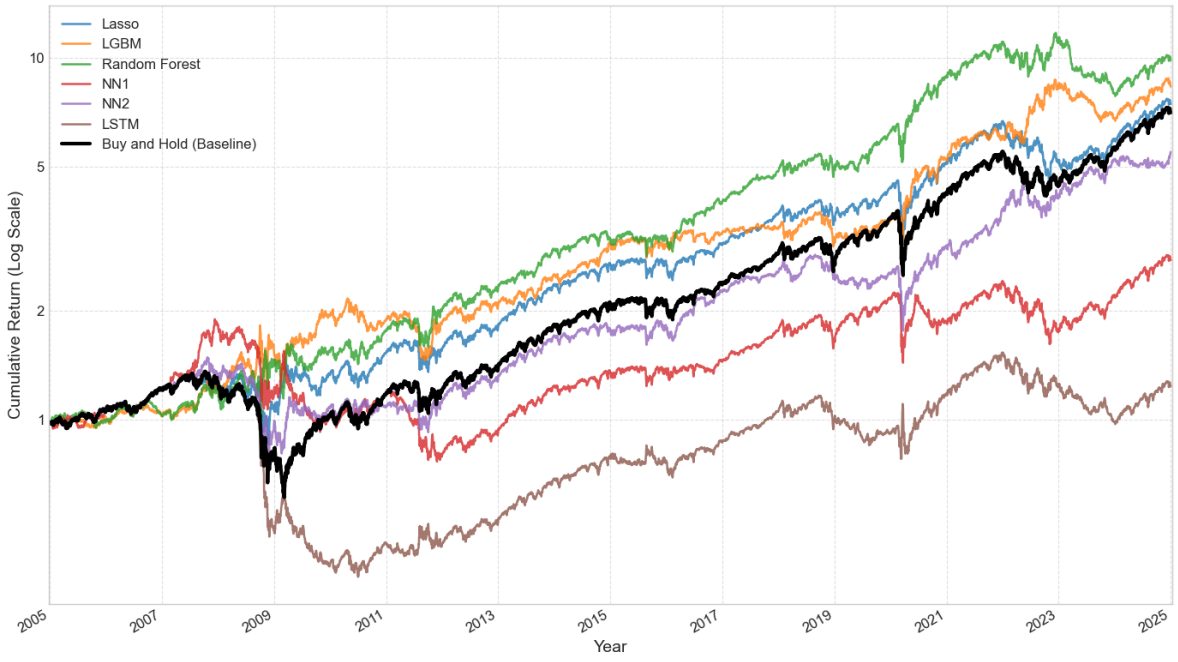

Figure 2: Cumulative returns of trading strategies derived from model-generated signals alongside the passive buy-and-hold benchmark over the 2005–2024 period.

These temporal performance patterns are further illustrated by the equity curves in Figure 2, which display cumulative returns of all models throughout the study period. Model returns generally track the buy-and-hold strategy, except during periods of market turbulence where divergences emerge. Two notable findings stand out: first, the clear outperformance of Random Forest, which achieved the highest cumulative returns throughout almost the entire period since 2012. Second, the convergence of Lasso regression towards the buy-and-hold benchmark. Although Lasso initially outperformed, its advantage gradually diminished over time, with a pronounced convergence observed in 2023 and 2024, which shows how simple strategies lose the edge.

The benefits of actively managed ML strategies become more pronounced when considering downside risk. The top three models - RF, LightGBM, and Lasso exhibited significantly lower maximum drawdowns than the buy and hold approach. From this group, Lasso had the largest MDD of 36%, which is still far lower than the MDD of the benchmark strategy, which reached 55%.

Reducing drawdowns offers two crucial advantages to investors. First, it provides greater flexibility regarding investment horizons, as reduced capital loss allows investors to avoid being locked into long recovery periods. Second, because recovery from drawdowns is non-linear, the impact on long-term performance is amplified. For example, a strategy that incurs a 26% drawdown (as did RF in 2005-2008) requires a 35% gain to return to break-even, while strategy with a 50.76% drawdown (B&H in 2005-2008), would require over a 100% gain. These differences highlight that drawdown reduction alone can significantly improve long-term capital accumulation, even if volatility remains similar across strategies. It is worth noting that all models in this study were fully invested at all times, hence their volatility closely mirrors that of the underlying index.

Performance differences are also reflected in the hit rate, which captures the directional accuracy of predictions and indirectly informs the skewness of return distributions. Across the full sample, the buy and hold strategy recorded a hit rate of 55%, representing the proportion of trading days with positive SPY returns. However, this ratio declines to 50.15% for returns exceeding one standard deviation, and further to 41.91% for returns exceeding two standard deviations.

These patterns reveal several important insights. Firstly, the most substantial returns-particularly those exceeding two standard deviations-tend to occur during market turmoil, making downside prediction accuracy especially valuable. All models that outperformed the benchmark were more accurate than buy and hold in adapting to these extreme movements. Secondly, although large returns occur less frequently, movements exceeding one standard deviation are far more common, and their directionality is nearly evenly split. The ability to correctly anticipate such frequent, moderate moves was a distinguishing feature of the outperforming models. In contrast, all neural network-based models underperformed the passive approach in this category, and their marginally higher hit rates for extreme moves did not compensate for this shortfall.

Two main themes emerge from these findings. First, when appropriately selected and implemented, machine learning models can provide meaningful improvements in return and risk-adjusted performance, even under relatively simple model and trading logic. Over the full backtest period, for Random Forest based strategy this translated into over 200 percentage points in additional cumulative return relative to the benchmark, while also significantly reducing drawdowns and recovery times.

Second, the underperformance of neural networks, which may initially appear counterintuitive given their success in similar return prediction applications, highlights important limitations. Their poor results are likely attributable to insufficient data, suboptimal hyperparameter selection, or both. In contrast, models such as RF and LightGBM are more robust in low-data regimes due to their ensemble structures (e.g., bagging and boosting). On the other hand, Lasso benefited from its simplicity and strong regularization, which mitigated the risk of overfitting, which resulted in more conservative signal generation and a markedly lower number of trades over the backtest period. This restrained trading frequency led Lasso to closely follow the general market trend, effectively anchoring its performance to the buy and hold approach while still capturing some predictive gains. This balance between simplicity and selective trading likely explains its outperformance relative to more complex but higher-variance models like neural networks.

Overall, these results underscore the potential of machine learning models in financial forecasting, particularly when volatility, regime shifts, and drawdown control are central to strategy design.

5.2 Feature Importance Analysis

The contribution of individual variables to model performance was assessed using a feature elimination approach applied to the best-performing model identified in the previous section - Random Forest. This analysis aimed to identify which features most influenced risk-adjusted returns. Two complementary strategies were employed. First, each of the 20 features was excluded from the dataset, followed by re-estimation of the model, including re-running Bayesian hyperparameter optimization, and computing out-of-sample performance over the full backtest horizon. Second, the same procedure was applied to predefined feature groups ("buckets") representing broader thematic categories: fixed income, foreign exchange, commodities, equity indices, and technical indicators.

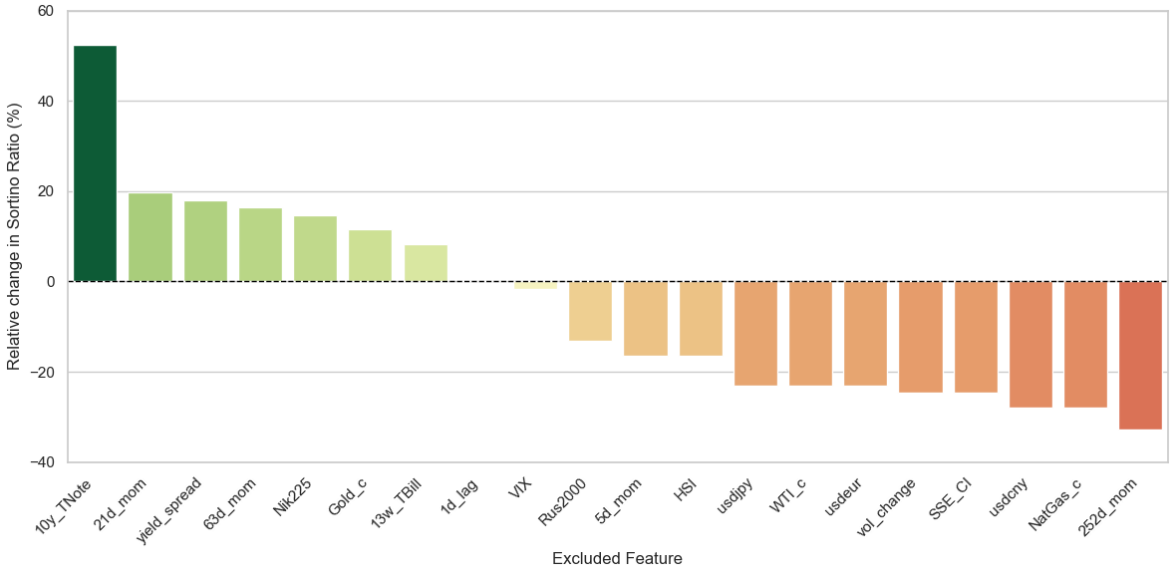

Figure 3: Impact of individual feature elimination on model performance, measured by the change in Sortino ratio relative to model trained on full feature set.

The results of the individual feature elimination study are shown in Figure 3. A key finding is that removing certain features improved model performance. In several instances, the Sortino ratio increased, indicating that some features degrade model effectiveness. One explanation is that specific market variables exhibit regime-dependent relationships with returns that the model fails to capture even despite annual retraining on shifted validation window. Another possibility is that some features have limited predictive relevance for SPY returns, introducing noise rather than signal. However, this effect appears limited, as Random Forest inherently reduces the influence of irrelevant variables as evidenced by the neutral impact of removing the lagged return.

It is important to note that the results are partly influenced by the stochastic nature of Bayesian optimization. Since the full hyperparameter tuning process was re-run in each iteration, some variation in performance may result from random factors. While this may affect the magnitude, and in borderline cases the direction, of performance changes, the consistency of both individual and bucket-level results suggests that changes to the feature set are the primary drivers.

A particularly striking result is that removing the 10-year Treasury yield increased the Sortino ratio by over 50%, boosting the ARC from 12.1% to approximately 17.3%. Other fixed income features also negatively impacted model performance, albeit to a lesser extent. This aligns with prior research (Guo et al., 2011; Yildirim, 2017), which found that at lower frequencies yields are often reactive to equity markets. As such, they may contribute more noise than predictive signal. Although Random Forest should mitigate purely noisy features, the negative contribution of the 10-year yield may reflect regime-dependent effects. Hu, Jin, and Pan (2023) show that at high frequencies, bond movements can precede movements in equities and exhibit regime-dependent relationships. If the model captures such patterns but lacks signals for regime shifts, it may generate false predictions based on outdated dynamics. Further inspection shows that features whose removal enhanced performance often relate to fixed income, medium-term technical indicators, and gold. These variables, while economically intuitive, may be reactive (e.g., gold) or too sluggish to capture daily dynamics (e.g., medium-horizon technicals).

Conversely, certain features consistently improved model performance. Notably, the shortest-term (5-day) and longest-term (252-day) momentum indicators were associated with higher risk-adjusted returns, suggesting their ability to capture distinct dimensions of market behavior. The 252-day measure likely proxies for prevailing market regimes, while the 5-day indicator reflects short-term momentum effects. Additionally, variables related to foreign exchange and energy commodities contributed positively, likely capturing global macroeconomic shifts and geopolitical events influencing SPY returns. Trading volume, particularly surges in volume, also exhibited a positive effect, consistent with prior research highlighting its informational value during periods of market stress (e.g., Blume, Easley, and

O'Hara, 1994; Chordia, Roll, and Subrahmanyam, 2001).

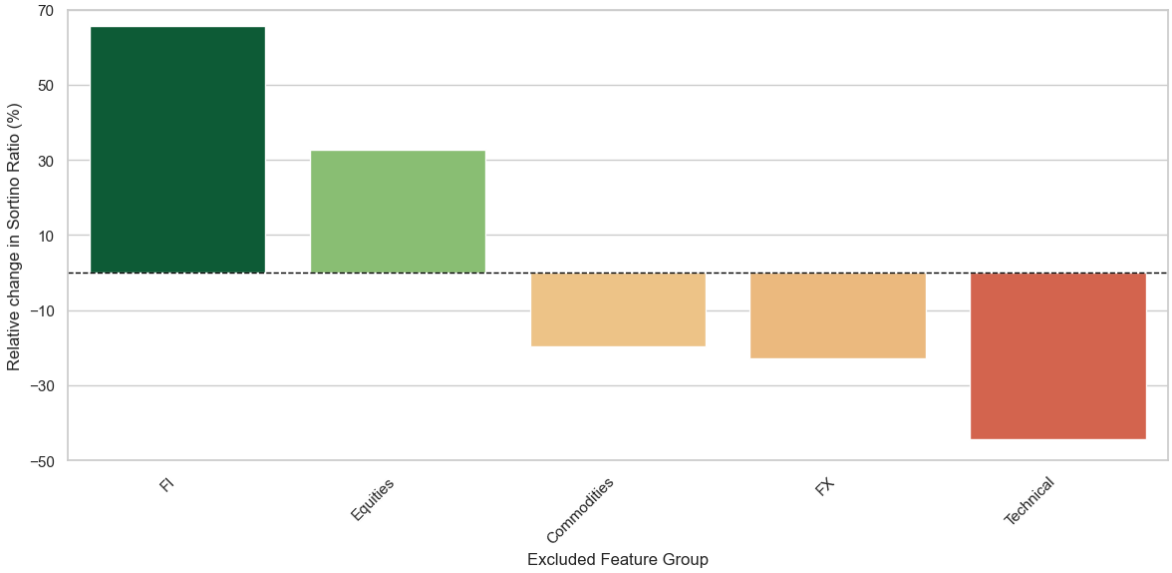

Figure 4: Impact of feature bucket elimination on model performance, measured by the change in Sortino ratio relative to model trained on full feature set.

The bucket-level elimination results (Figure 4) broadly corroborate the individual feature findings, with a few differences. Removing the fixed income bucket led to a significant rise in the Sortino ratio, while removing the commodity and FX buckets reduced performance. Interestingly, eliminating the entire technical-indicator bucket produces the greatest decline in model effectiveness, indicating that the negative impact of medium-term momentum is more than offset by the positive contributions of other technical measures. Perhaps most surprising, however, is that despite the Nikkei index emerging as the only individual equity index with a negative effect, exclusion of all equity indices enhances overall model performance. This suggests complex interactions among equity predictors, whereby collective removal may eliminate cross-asset noise that individual perturbations fail to isolate.

Further insights can be obtained by examining the cumulative return trajectories associated with the exclusion of individual feature buckets, which are presented in Figure 5. Notably, the cumulative returns of the model excluding the fixed income bucket consistently outperformed all other specifications across the majority of the sample period. This effect is particularly pronounced during episodes of heightened market stress, including the Global Financial Crisis, the COVID-19 pandemic, and the 2022–23 period of monetary policy tightening in the United States. These findings provide additional evidence that, while interest rates exert a well-established influence on equity markets, fixed income indicators do not constitute effective predictive signals for lower-frequency strategies. Interestingly, a similar pattern emerges when excluding the equity indices bucket; however, this specification fails to capture the substantial return acceleration observed during the most recent rate-hiking cycle. This suggests that, akin to fixed income variables, cross-asset information embedded in international equity indices may already be efficiently incorporated into S&P 500 price dynamics. Finally, the exclusion of technical indicators produces the most pronounced deterioration in performance, underscoring their critical role in the forecasting model. This result stands in contrast to the Efficient Market Hypothesis, which posits that past price information should be fully reflected in current market valuations. The persistent outperformance observed when technical indicators are included highlights their continued empirical relevance, despite theoretical arguments to the contrary.

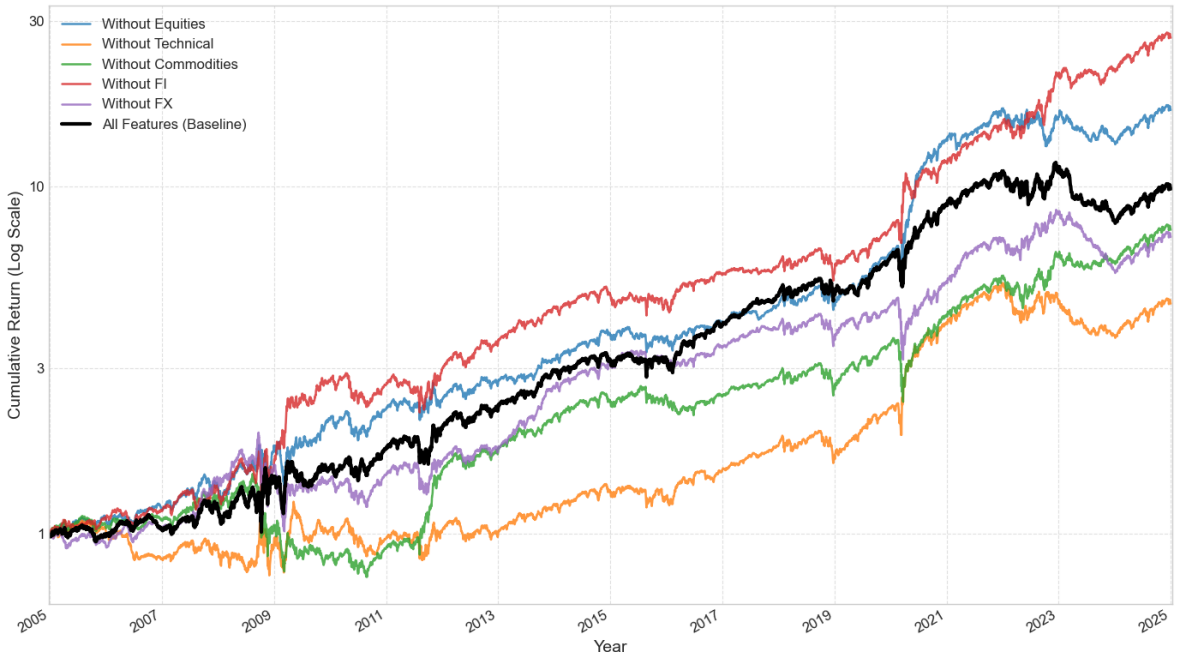

Figure 5: Cumulative returns of trading strategies derived from models with sequential feature bucket exclusions, benchmarked against the full-feature specification, over the 2005–2024 period.

In summary, the feature elimination results show that not all predictors enhance model performance—some degrade it. Short- and long-term momentum, volume surges, and select macro features boost returns, while interest rate and equity index inputs may introduce noise. Crucially, these findings emerge under realistic re-training and hyperparameter tuning, emphasizing their relevance for model development. Taken together, the individual and group-level results provide a data-driven path to reduce dimensionality, improving both interpretability and out-of-sample robustness in directional equity forecasting.

6 Conclusion

This study investigated the extent to which machine learning models can generate economically valuable forecasts of the daily return direction of the S&P 500 ETF (SPY), using a broad and diverse set of features including macroeconomic indicators, cross-asset signals, and technical variables. The research was guided by four hypotheses: (RH1) that ML models can produce forecasts with sufficient accuracy to yield superior trading performance over buy and hold approach; (RH2) that ML-based, active strategies should especially outperform during market crises; (RH3) that ML-based, active strategies should reduce risk; and (RH4) that LSTM, which is tailor-made for time-series prediction should outperform other models.

The empirical evidence provides partial support for H1. Certain models, especially tree-based approaches such as Random Forest and LightGBM, achieved superior risk-adjusted returns (measured by the Sortino ratio) compared to the buy-and-hold benchmark over the full backtest period. However, this outperformance was moderate, inconsistent, and largely concentrated in specific episodes. Outside of crisis periods, active strategies frequently failed to outperform, and at times underperformed the benchmark on both total and risk-adjusted returns. This suggests that while machine learning methods can enhance performance, their advantage is conditional and not uniformly robust across all market conditions.

The results lend stronger, though still not definitive, support to H2. All active models that surpassed the benchmark on a risk-adjusted basis did so largely thanks to their stronger performance during the 2008 market crash. This suggests that machine learning models may be especially adept at navigating environments characterized by heightened volatility and structural dislocations. Nevertheless, their edge appears to have diminished in more recent crises. During the COVID-19 downturn,

for instance, the buy and hold strategy fared better than most active approaches, raising questions about the continued effectiveness of these models in increasingly efficient markets. Therefore, while the historical evidence indicates that machine learning strategies can outperform during certain types of market stress, the lack of consistent superiority across all downturns tempers the strength of this conclusion.

H3 also receives partial support. The active ML strategies that outperformed in returns also exhibited materially lower drawdowns, indicating superior downside protection and improved capital preservation during turbulent periods. However, overall portfolio volatility remained broadly comparable to the passive benchmark. This suggests that the primary risk management benefit of ML-based strategies lies in mitigating tail risk, rather than reducing day-to-day return fluctuations.

Finally, the results clearly reject H4. Contrary to expectations, neural networks, including LSTM, consistently underperformed other models and the benchmark across most performance metrics. This underperformance likely stems from insufficiently large dataset – as neural networks are known to require large amounts of data for effective training. Alternatively, the issue resulted from suboptimal parameter tuning – which also highlights another challenge of using complex models.

Overall, the findings suggest that machine learning can enhance trading performance, but its success is highly dependent on model choice, feature selection, and market regime. In this application simpler, tree-based models displayed the most consistent risk-adjusted outperformance, primarily by reducing drawdowns during market stress. Deep learning models failed to deliver superior performance in this context. Future research should explore more adaptive strategies, including dynamic model selection, regime-switching frameworks, and hybrid approaches that incorporate economic theory or structural signals alongside machine learning forecasts.

In addition to evaluating trading performance, the study also offers insights through a detailed feature importance analysis. This component sheds light on which input variables were most influential in driving model predictions, revealing that cross-asset signals and selected technical indicators consistently ranked among the most informative features. This analysis not only enhances interpretability, a common concern in machine learning applications, but also underscores the potential value of transparent and explainable AI in quantitative finance. The methodology and findings in this section represent a novel contribution and open avenues for future research into model explainability and feature engineering.

In conclusion, this study contributes to a more balanced understanding of machine learning's role in financial prediction. While ML models can extract valuable signals under certain conditions, their benefits are nuanced and context-dependent. Realistic expectations and careful strategy design are essential for harnessing machine learning effectively in financial markets.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T. & Koyama, M. (2019), 'Optuna: A next-generation hyperparameter optimization framework', *arXiv (Cornell University)* .

BLUME, L., EASLEY, D. & O'HARA, M. (1994), 'Market statistics and technical analysis: The role of volume', *The Journal of Finance* **49**, 153–181.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1540-6261.1994.tb04424.x*

Breiman, L. (2001), 'Random forests', *Machine Learning* **45**, 5–32.

Bussmann, N., Giudici, P., Marinelli, D. & Papenbrock, J. (2020), 'Explainable machine learning in credit risk management', *Computational Economics* **57**.

Chen, K., Zhou, Y. & Dai, F. (2015), 'A lstm-based method for stock returns prediction: A case study of china stock market', *2015 IEEE International Conference on Big Data (Big Data)* .
URL: *https://ieeexplore.ieee.org/abstract/document/7364089*

Chordia, T., Roll, R. W. & Subrahmanyam, A. (2000), 'Market liquidity and trading activity', *SSRN Electronic Journal* .

Cont, R. (2001), 'Empirical properties of asset returns: stylized facts and statistical issues', *Quantitative Finance* **1**, 223–236.

Fama, E. (1970), 'Efficient capital markets: A review of theory and empirical work', *The Journal of Finance* **25**, 383–417.

Fischer, T. & Krauss, C. (2018), 'Deep learning with long short-term memory networks for financial market predictions', *European Journal of Operational Research* **270**, 654–669.

Gu, S., Kelly, B. & Xiu, D. (2020), 'Empirical asset pricing via machine learning', *The Review of Financial Studies* **33**, 2223–2273.

Guo, K., Zhou, W.-X., Cheng, S.-W. & Sornette, D. (2011), 'The us stock market leads the federal funds rate and treasury bond yields', *PLoS ONE* **6**, e22794.

Hochreiter, S. & Schmidhuber, J. (1997), 'Long short-term memory', *Neural Computation* **9**, 1735–1780.
URL: *https://direct.mit.edu/neco/article-abstract/9/8/1735/6109/Long-Short-Term-Memory?redirectedFrom=fulltext*

Hu, G. X., Jin, Z. & Pan, J. (2023), 'Comovements in global markets and the role of u.s. treasury', *SSRN Electronic Journal* .

Huang, W., Nakamori, Y. & Wang, S.-Y. (2005), 'Forecasting stock market movement direction with support vector machine', *Computers Operations Research* **32**, 2513–2522.

Jegadeesh, N. & Titman, S. (1993), 'Returns to buying winners and selling losers: Implications for stock market efficiency', *The Journal of Finance* **48**, 65–91.
URL: *https://www.jstor.org/stable/2328882*

Kara, Y., Acar Boyacioglu, M. & Baykan, K. (2011), 'Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the istanbul stock exchange', *Expert Systems with Applications* **38**, 5311–5319.
URL: *https://www.sciencedirect.com/science/article/pii/S0957417410011711*

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. & Liu, T.-Y. (2017), 'Lightgbm: A highly efficient gradient boosting decision tree'.
URL: *https://proceedings.neurips.cc/paper*$_f$*iles/paper*/2017/$file$/6449f44a102fde848669bdd9eb6b76$fa-$
Paper.pdf

Krauss, C., Do, X. A. & Huck, N. (2017), 'Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the sp 500', *European Journal of Operational Research* **259**, 689–702.

Lo, A. W. & MacKinlay, A. C. (1990), 'When are contrarian profits due to stock market overreaction?', *Review of Financial Studies* **3**, 175–205.

Lundberg, S. & Lee, S.-I. (2017), 'A unified approach to interpreting model predictions'.
URL: *https://arxiv.org/abs/1705.07874v2*

Navani, J. & Giri, N. (2024), 'Can ai models outperform the traditional buy-and-hold strategy?', *Journal of Electrical Systems* **20**, 1211–1219.

Passalis, N., Kanniainen, J., Gabbouj, M., Iosifidis, A. & Tefas, A. (2021), 'Forecasting financial time series using robust deep adaptive input normalization', *Journal of Signal Processing Systems* **93**, 1235–1251.

Persio, L. D. & Honchar, O. (2016), 'Artificial neural networks approach to the forecast of stock market price movements', *International Journal of Economics and Management Systems* **01**, 158–162.

Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986), 'Learning representations by back-propagating errors', *Nature* **323**, 533–536.

Sharpe, W. F. (1994), 'The sharpe ratio', *The Journal of Portfolio Management* **21**, 49–58.

Sortino, F. A. & Price, L. N. (1994), 'Performance measurement in a downside risk framework', *The Journal of Investing* **3**, 59–64.

Takeuchi, L. & Lee, A. (2013), 'Applying deep learning to enhance momentum trading strategies in stocks'.
URL: *https://cs229.stanford.edu/proj2013/TakeuchiLee-ApplyingDeepLearningToEnhanceMomentumTradingStrateg*

Tibshirani, R. (1996), 'Regression shrinkage and selection via the lasso', *Journal of the Royal Statistical Society: Series B (Methodological)* **58**, 267–288.

Turner, R., Eriksson, D., McCourt, M., Kiili, J., Laaksonen, E., Xu, Z. & Guyon, I. (2021), 'Bayesian optimization is superior to random search for machine learning hyperparameter tuning: Analysis of the black-box optimization challenge 2020'.
URL: *https://arxiv.org/abs/2104.10201*

Yildirim, H. (2017), 'The interactions between sp500 and 10-years treasury bond returns in the us', *International Journal of Academic Value Studies (Javstudies)* **3**.

Appendix

Model	Period	CAGR (%)	Cum. Ret. (%)	B&H Cum. Ret. (%)	MDD (%)	# Trades	St. Dev. (%)	Sharpe ratio	Sortino ratio	Hit Rate (%)	Hit $> 1\sigma$ (%)	Hit $> 2\sigma$ (%)
rf	05-08	10.0	46.5	-19.3	-26.0	282	23.3	0.24	0.31	52.6	51.1	53.1
rf	09-12	11.9	56.8	71.8	-21.9	156	20.8	0.44	0.59	54.5	51.9	48.3
rf	13-16	14.9	74.4	70.2	-14.8	176	12.8	0.99	1.34	54.6	54.7	48.2
rf	17-20	22.3	123.5	80.4	-21.7	197	20.2	1.00	1.10	55.7	51.9	43.7
rf	21-24	2.4	10.1	66.0	-33.2	199	16.5	-0.04	-0.06	52.1	48.2	48.2
lgbm	05-08	11.7	55.3	-19.3	-31.9	387	23.3	0.31	0.41	51.4	51.8	57.4
lgbm	09-12	7.3	32.6	71.8	-32.6	286	20.8	0.22	0.29	53.4	51.5	43.3
lgbm	13-16	12.3	59.1	70.2	-11.9	173	12.8	0.79	1.07	53.8	54.7	46.5
lgbm	17-20	12.6	60.7	80.4	-19.7	361	20.2	0.52	0.64	51.6	51.9	45.8
lgbm	21-24	12.2	58.3	66.0	-23.6	347	16.5	0.55	0.80	52.5	54.0	51.7
lasso	05-08	2.1	8.8	-19.3	-36.3	101	23.3	-0.09	-0.11	53.7	48.2	51.0
lasso	09-12	13.3	64.7	71.8	-21.4	80	20.8	0.51	0.67	54.8	51.5	51.6
lasso	13-16	14.1	69.4	70.2	-13.0	13	12.8	0.92	1.23	54.8	53.5	44.8
lasso	17-20	14.2	70.3	80.4	-33.7	21	20.2	0.60	0.64	55.9	50.0	37.5
lasso	21-24	9.6	44.4	66.0	-31.6	67	16.5	0.39	0.55	54.0	50.7	44.6
bh	05-08	-5.1	-18.9	-19.3	-50.8	0	23.3	-0.41	-0.47	53.8	45.3	46.8
bh	09-12	14.5	71.8	71.8	-27.1	0	20.8	0.57	0.73	55.7	51.9	48.3
bh	13-16	14.2	70.2	70.2	-13.0	0	12.8	0.94	1.24	54.9	53.5	44.8
bh	17-20	15.9	80.4	80.4	-33.7	0	20.2	0.69	0.72	56.8	50.0	35.4
bh	21-24	13.6	66.0	66.0	-24.5	1	16.5	0.63	0.87	54.0	53.7	46.4
nn2	05-08	-0.9	-3.7	-19.3	-51.7	131	23.3	-0.23	-0.27	53.9	46.8	40.4
nn2	09-12	7.7	34.2	71.8	-20.1	139	20.8	0.24	0.32	52.8	50.2	51.6
nn2	13-16	14.9	74.0	70.2	-12.9	99	12.8	0.99	1.31	54.8	54.3	44.8
nn2	17-20	7.8	35.0	80.4	-38.4	291	20.2	0.28	0.31	53.0	48.1	35.4
nn2	21-24	16.1	81.3	66.0	-21.9	274	16.5	0.79	1.18	54.6	52.2	51.7
nn1	05-08	4.5	19.3	-19.3	-43.7	245	23.3	0.01	0.01	53.2	50.4	48.9
nn1	09-12	-6.3	-22.9	71.8	-50.5	155	20.8	-0.44	-0.58	51.4	46.8	45.0
nn1	13-16	12.0	57.0	70.2	-11.9	81	12.8	0.76	1.02	53.5	54.3	43.1
nn1	17-20	6.6	29.1	80.4	-36.1	79	20.2	0.22	0.25	53.1	44.9	41.6
nn1	21-24	10.3	48.0	66.0	-33.0	27	16.5	0.44	0.60	54.1	52.2	42.8
lstm	05-08	-15.9	-49.9	-19.3	-65.3	13	23.3	-0.87	-0.92	53.3	42.5	42.5
lstm	09-12	0.9	3.7	71.8	-44.7	4	20.8	-0.09	-0.13	50.7	49.8	48.3
lstm	13-16	13.5	66.1	70.2	-18.5	3	12.8	0.88	1.21	55.9	48.8	50.0
lstm	17-20	8.2	37.1	80.4	-33.1	145	20.2	0.30	0.35	53.4	50.6	41.6
lstm	21-24	1.1	4.3	66.0	-36.6	3	16.5	-0.13	-0.18	51.0	51.1	44.6

Table 4: Performance comparison of trading strategies over 4-year subperiods.