Forecasting Stock Returns with Explainable Machine Learning Models

University of Warsaw, Faculty of Economic Sciences

Krzysztof Płachta   Robert Ślepaczuk

November 3, 2025

Overview of the Study

- The study is divided into two main parts, linking **predictive performance** with **economic interpretability**.
- **Part I – Model Performance Comparison:**
  - Evaluates and compares six machine learning models in forecasting next-day S&P 500 returns.
  - Tests their performance and robustness across a 20-year period.
- **Part II – Feature Importance Analysis:**
  - Analyzes which features drive or hinder model performance.
  - Provides insight into which types of market information offer a predictive edge.

Outline

K. Płachta & R. Ślepaczuk (2025)
https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5445537

Motivation

- **Case for ML models:**
  - ML models can discover non-linear dependencies and filter noise effectively.
  - Financial markets have a notoriously low signal-to-noise ratio, making ML a natural choice for predicting returns.
- **Challenges:**
  - Many ML models exist - how to select the most suitable one?
  - High complexity of ML models makes interpretability difficult.
  - Which data actually contains valuable predictive information?
- **Core objectives:**
  - Identify the ML model that best predicts next-day S&P 500 returns (binary prediction task).
  - Understand how individual features contribute to model performance.

Novelties and Contributions

Main contributions of the study:

- **Model comparison:** Comprehensive evaluation of six machine learning models for binary classification of next-day S&P 500 returns over a 20-year period (2005–2024).
- **XAI methodology:** Development of a novel, model-agnostic framework that links the contribution of individual features to the economic performance of machine learning models.
- **XAI insights:** Reveals which types of information were not efficiently incorporated by the market and provided a predictive edge for short-term movements in the U.S. stock market, as well as which features carried little or no informational value.

Research Hypotheses

- **H1:** ML-based trading strategies will 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.
- **H5:** Feature importance analysis should show that removing features will have negative or neutral impact on the models' performance.

Data and Feature Engineering

Data Overview

- **Custom dataset** constructed using publicly available sources (Yahoo Finance, Stooq)
- **Daily frequency**, covering the period Jan 2000 – Dec 2024 (6,289 observations)
- **Prediction target**: next-day directional return of SPY ($\{+1, \text{-}1\}$)
- **20 predictive features** drawn from five categories:
  - Equities, FX, Commodities, Fixed Income, Technical Indicators
- **Preprocessing steps**:
  - Log-returns computed for price series
  - Level variables (e.g., VIX, yields) standardized
  - All predictors lagged by one day to preserve out-of-sample integrity

Summary Statistics of Features

Bucket	Feature	Data Type	Mean	Std Dev	Min	Max
Technicals	vol_change	fract. change	0.06	0.37	-0.83	4.64
	1d_lag	log-rets	0.00	0.01	-0.12	0.14
	5d_mom	c. log-rets	0.00	0.02	-0.22	0.18
	21d_mom	c. log-rets	0.01	0.05	-0.40	0.22
	63d_mom	c. log-rets	0.02	0.08	-0.54	0.34
	252d_mom	c. log-rets	0.07	0.17	-0.64	0.57
Equities	SSE_CI	log-rets	0.00	0.01	-0.09	0.09
	HSI	log-rets	-0.00	0.01	-0.14	0.13
	Nik225	log-rets	-0.00	0.01	-0.13	0.10
	Rus2000	log-rets	0.00	0.02	-0.15	0.09
	VIX	level	0.20	0.08	0.09	0.83
Rates	13w_TBill	level	0.02	0.02	-0.00	0.06
	10y_TNote	level	0.03	0.01	0.00	0.07
	yield_spread	level	0.01	0.01	-0.02	0.04
FX	usdeur	log-rets	-0.00	0.01	-0.03	0.03
	usdjpy	log-rets	0.00	0.01	-0.04	0.05
	usdcny	log-rets	-0.00	0.01	-0.06	0.05
Commodities	WTI_c	log-rets	0.00	0.03	-0.29	0.22
	Gold_c	log-rets	0.00	0.01	-0.09	0.10
	NatGas_c	log-rets	0.00	0.04	-0.35	0.62

Part I - Model Performance Comparison

Selected Models and Objective Function

- Six supervised learning models covering linear, tree-based, and deep learning approaches:
  - Lasso (linear model with regularization)
  - Random Forest (tree ensemble)
  - LightGBM (gradient-boosted trees)
  - LSTM (recurrent neural network)
  - Feedforward Neural Network (1 hidden layer)
  - Feedforward Neural Network (2 hidden layers)
- Neural networks use pyramidal architectures: each hidden layer has half the neurons of the previous layer.
- **Reward Function:** Accuracy score

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

Hyperparameter Search Spaces

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

Hyperparameter Tuning

- **Requirements:**
  - Ensure fair tuning across models with different hyperparameters
  - Limit compute cost (620+ optimization cycles across all backtests)
  - Maximize validation accuracy
- **Solution:** Bayesian Optimization (TPE algorithm via Optuna)
  - More efficient than grid/random search (Turner et al., 2021)
  - Enables consistent trial budget across models
- **Search Design:**
  - 50 trials per model for balanced compute
  - Model-specific search spaces defined via expert judgment

Backtesting Framework

- **Forecasting scheme:** Expanding window with annual re-estimation
  - 5-year initial training, 1-year validation
  - Roll forward one year; repeat training and tuning
- **Evaluation:** Daily predictions are strictly out-of-sample during test period
- **Trading rule:** Fully long or short SPY at close based on predicted sign
  - No transaction costs assumed

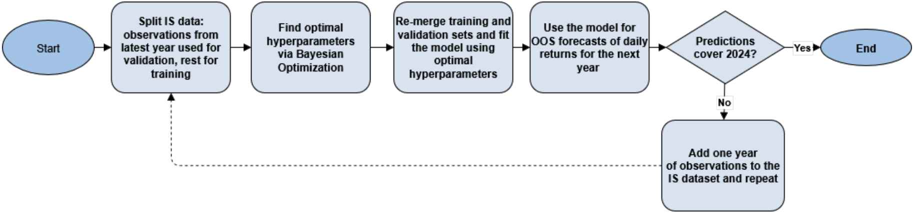


Figure: Flowchart of the backtesting procedure

Evaluation Metrics

Equity line (per-day update):

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

- Simulated portfolio fully long or short SPY based on model prediction

Reported metrics:

- Annualized Return Compounded (ARC)
- Cumulative return
- Maximum Drawdown (MDD)
- Number of trades
- Annualized Standard Deviation (ASD)
- Sharpe Ratio
- Sortino Ratio
- Hit Rate
- Hit Rate on high-volatility days (returns exceeding 1σ or 2σ)

Backtest Results

Advantages of ML strategies

- RF and LGBM achieved materially higher Sortino ratios (+33% and +24% relative to buy-and-hold strategy)
- Successful models also reduced maximum drawdown by approximately 40%

Limitations of ML strategies

- Neural networks underperformed significantly
- Inconsistent performance across time
- No meaningful reduction in volatility

Model	ARC (%)	Cum. Ret. (%)	MDD (%)	# of Trades	ASD (%)	Sharpe ratio	Sortino ratio	Hit rate (%)	Hit $> 1\sigma$ (%)	Hit $> 2\sigma$ (%)
RF	12.13	885.7	-33.2	1008	19.05	0.48	0.61	53.90	50.30	46.89
LGBM	11.19	733.8	-32.6	1550	19.06	0.44	0.57	52.55	51.59	49.79
Lasso	10.58	646.2	-36.3	279	19.06	0.40	0.49	54.64	50.26	43.57
B&H	10.30	609.8	-55.2	1	19.06	0.39	0.46	55.04	50.15	41.91
NN2	8.91	450.5	-51.7	930	19.06	0.32	0.39	53.82	49.43	44.40
NN1	5.21	175.8	-59.6	583	19.06	0.12	0.15	53.07	48.51	45.64
LSTM	1.06	23.3	-73.1	165	19.07	-0.10	-0.12	52.87	48.20	45.23

Models' Performance Across Time

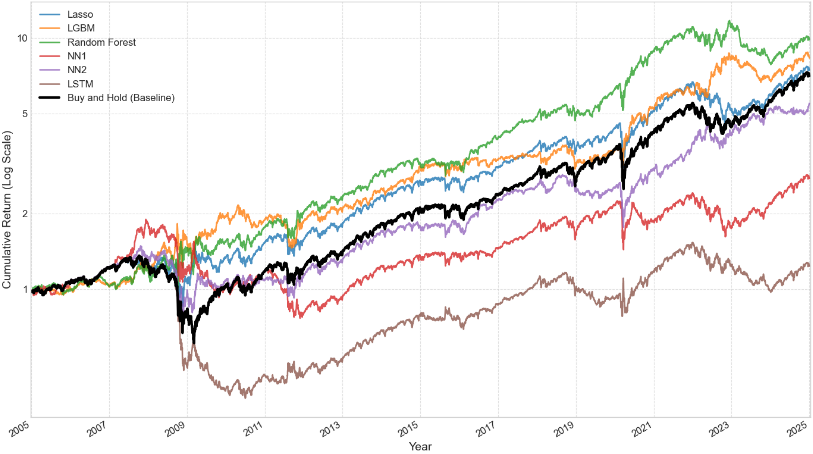


Figure: Cumulative returns of model-based strategies vs. buy-and-hold.

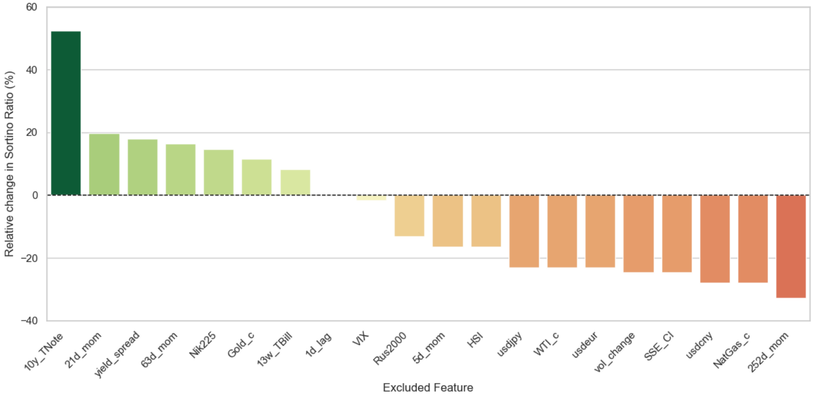
Part II - Feature Importance Analysis

Explainability Framework (XAI)

- After identifying the best model, we analyzed the **importance of input features**.
- We used a **model-agnostic retraining approach**:
  - Remove features or feature groups
  - Re-train and re-backtest
- Two levels of analysis:
  1. **Feature-wise:** Remove 1 of 20 features $\rightarrow$ 20 backtests
  2. **Category-wise:** Remove 1 of 5 feature groups $\rightarrow$ 5 backtests

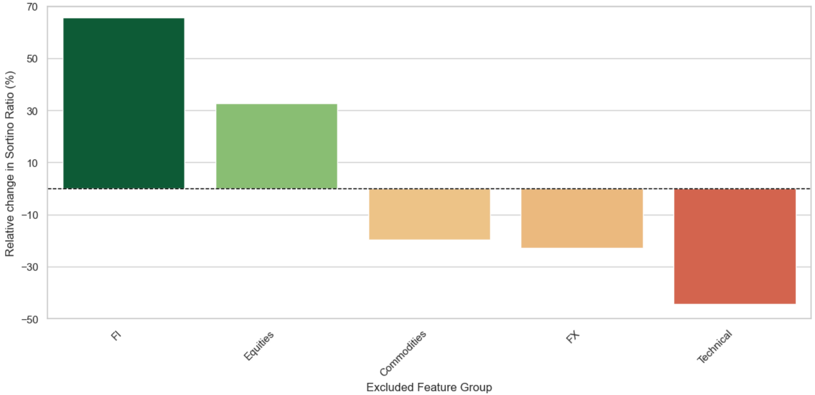
Feature Importance Analysis Results: Individual Features

- Inclusion of some features decreased model performance.
- Removing the 10-year Treasury yield increased Sortino ratio by over 50%.
- Technical indicators had the most mixed effect on performance.



Feature Importance Analysis Results: Buckets

- Results are consistent with individual feature elimination for Fixed Income, FX and Commodities.
- Effect of removal of Equity bucket was reverse to that observed in individual feature elimination.

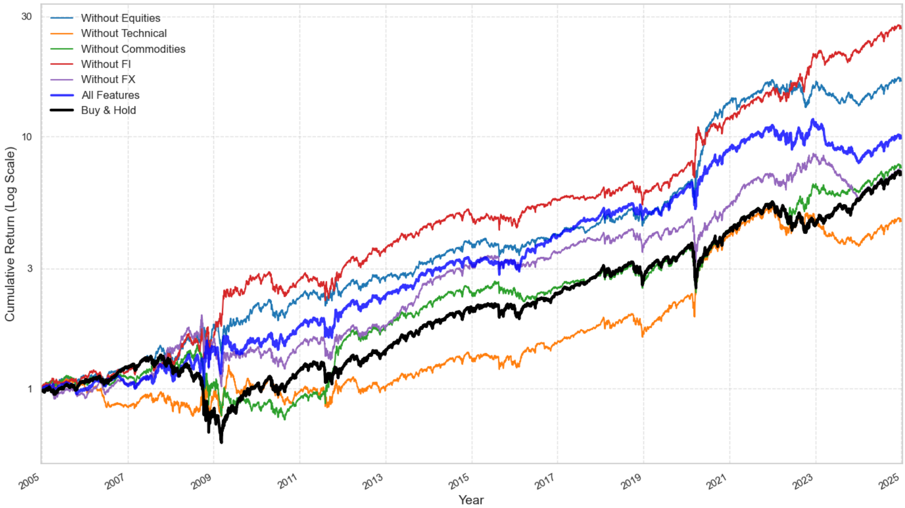


Detailed look at buckets elimination results

- **Excluding the Fixed Income bucket** produced the strongest performance across all metrics, except for the overall hit rate, which remained slightly below the buy-and-hold benchmark.
- **Lower drawdowns translated into higher cumulative returns**, indicating that improved downside control, rather than prediction frequency, was the key driver of outperformance.

Model	ARC (%)	Cum. Ret. (%)	MDD (%)	# of Trades	ASD (%)	Sharpe	Sortino	Hit (%)	$>1\sigma$ (%)	$>2\sigma$ (%)
FI	17.90	2587.5	-24.1	1168	19.04	0.79	1.01	54.29	52.42	52.70
Equities	15.10	1563.3	-27.8	951	19.05	0.64	0.81	54.11	51.70	48.13
All	12.13	885.7	-33.2	1007	19.05	0.48	0.61	53.90	50.36	46.89
Comm.	10.62	651.9	-49.5	922	19.06	0.41	0.49	54.15	49.95	45.23
b&h	10.30	609.8	-55.2	1	19.06	0.39	0.46	55.04	50.15	41.91
FX	10.36	616.8	-48.1	951	19.06	0.39	0.47	54.01	51.18	46.06

Impact of Bucket Removal Across Time



Conclusion

Main Takeaways

- **Model choice matters** – performance varies substantially across algorithms, with tree-based models proving the most effective for this task.
- **Feature selection is critical** – the predictive value of inputs differs widely across categories and time periods.
- **Explainable AI adds value** – using XAI to assess feature relevance enables targeted model refinement and can significantly enhance performance.
- **Fixed Income signals underperform** – yield- and rate-based variables provide little or negative predictive power for daily equity returns.

Research Hypotheses Answered

- **H1:** ML-based strategies outperform buy-and-hold.
  Answer: Partially supported – Tree-based models (RF, LGBM) improved risk-adjusted returns, but outperformance was episodic.
- **H2:** Outperformance is stronger during crises/high volatility.
  Answer: Supported in part – Strong gains during 2008 crash, weaker during COVID-19 downturn.
- **H3:** ML reduces overall portfolio risk.
  Answer: Partially supported – Lower drawdowns observed, but overall volatility similar to benchmark.
- **H4:** Neural networks achieve highest risk-adjusted performance.
  Answer: Rejected – Neural networks, including LSTM, consistently underperformed other models.
- **H5:** Removing predictors uniformly degrades performance.
  Answer: Rejected – while this was true in most cases, in a few examples removing features drastically improved performance.

Thank You

Thank you for your attention!

In case of questions or comments, please reach out via email or LinkedIn (Krzysztof Płachta).

`k.plachta2@student.uw.edu.pl`

Scan the QR code to access the full paper on SSRN