Predicting S&P 500 Direction with Ensemble Methods

Christian Weißmeier Farkas Tallos

Statistical and Machine Learning (2025/26)

November 17, 2025

Agenda

1. Introduction and Data
2. Methodology
3. Model Results
   - Elastic Net
   - Random Forest
   - Gradient Boosting Machine
4. Trading Strategy
5. Conclusion

Motivation

"Stock returns are predictable, but not by much."

— John H. Cochrane, Asset Pricing (2005)

- Forecasting stock market direction is one of the most classical and challenging tasks in finance.
- Weak predictability can matter for portfolio allocation and risk management.

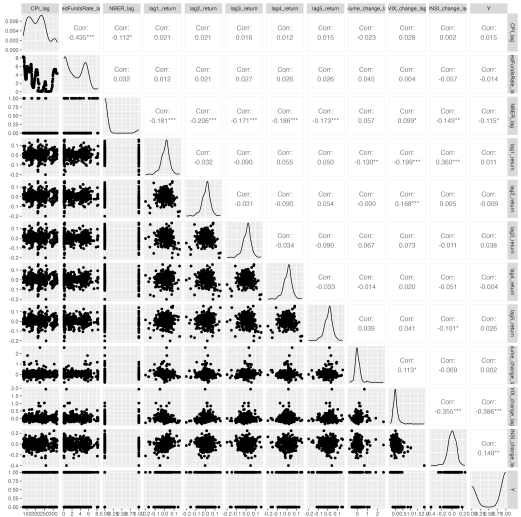
Our Set-Up:

- We focus on predicting whether the S&P 500 index goes **Up** or **Down** next month.
- We evaluate statistical learning methods in a time-series context.
- Introduce the role of regularization and nonlinear models.
- Compare linear vs. nonlinear classification models (Elastic Net vs. Random Forest vs. GBM).

Data Overview

- We created our own monthly dataset from S&P 500, FRED, and FRBSF data banks (1990–today).
- **Target:** Monthly S&P 500 market direction (**UP** or **DOWN**) in $t + 1$.
- **Predictors:**
  1. **Market data:** Lagged S&P 500 returns (up to five months) and trading volume changes.
  2. **Macroeconomic indicators:** CPI, Federal Funds Rate, NBER recession dummy.
  3. **Volatility:** Lagged changes in the VIX index.
  4. **Sentiment:** Daily News Sentiment Index.
- All features are lagged to avoid look-ahead bias.

Exploratory Analysis: Feature Relationships



Pairwise correlations and marginal distributions.

Why these predictors may matter:

- **Macro indicators** (CPI, Fed Funds Rate, NBER) affect discount rates and expected returns.
- **Lagged returns** reflect momentum and reversal effects.
- **Volatility** (VIX) captures uncertainty and risk sentiment.
- **Sentiment** (DNSI) reflects investor expectations and behavioral biases.

Empirical observation:

- Weak linear correlations $\Rightarrow$ low-signal, non-linear problem.
- Several predictors show significant **multicollinearity**.
- Motivates using **Elastic Net**, **Random Forests**, and **Gradient Boosting**.

Stationarity Considerations

- Most macro-financial time series are **non-stationary** in levels.
- We therefore use:
  - **Lagged returns** instead of prices.
  - **Changes** in VIX and sentiment rather than levels.
  - **Changes in macro variables** to preserve temporal causality.
- This transformation makes features approximately stationary, ensuring:
  - Stable model coefficients over time.
  - Valid cross-validation across time periods.

Hyperparameter Tuning: Time-Series Cross-Validation

Why not standard cross-validation?

- Random k-fold CV assumes i.i.d. data.
- Time-series data exhibit autocorrelation $\Rightarrow$ temporal dependence.
- Random shuffling lets the model see the future $\Rightarrow$ data leakage and over-optimism.

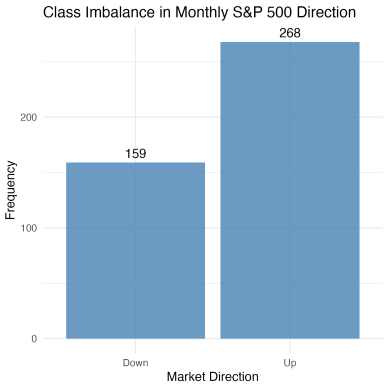
Our method: Rolling-window time-series CV

- Construct a hyperparameter grid:
  - Elastic Net: α and λ grid.
  - Random Forest: `mtry`, `min.node.size`, `max.depth`, `sample.fraction`.
- Use a chronological window structure:
  - Initial training window: 60 months (5 years).
  - Validation horizon: 12 months (1 year).
  - Fixed-length rolling window moving forward in time.
- For each fold:
  - Fit model only on past data and predict the next 12 months.
  - Record accuracy and AUC on the validation slice.

Result:

- Select the hyperparameters achieving the **highest mean AUC across all rolling folds**.
- Prevents look-ahead bias and mimics real-time forecasting.

Class Imbalance, Threshold Adjustment & AUC



Class imbalance in monthly S&P 500 direction

- Dependent variable `UP_DOWN` is imbalanced.
- A naive classifier that would always predict "Up" would achieve high accuracy.
- A simple threshold of 0.5 reduces the accuracy.
  $\Rightarrow$ **Accuracy alone is misleading.**

Threshold adjustment (Youden's J):

$$t^* = \arg\max_t \left\{ \text{Sensitivity}(t) + \text{Specificity}(t) - 1 \right\}$$

- We tune t^* on the **training set** using the ROC curve.
- Then apply this fixed threshold to the **test set** for true OoS evaluation.

AUC is robust here:

- ROC/AUC depends on the *ranking* of predicted probabilities, not on class proportions.
  $\Rightarrow$ **AUC is invariant to class imbalance** and a suitable metric for our setting.

Elastic Net Logistic Regression

Model Intuition:

- We model the probability of an **Up** move using a logistic function:

$$P(Y_t = 1 \mid X_t) = \frac{1}{1 + e^{-(\beta_0 + X_t \beta)}}$$

- Coefficients β are maximum likelihood estimates under a regularization penalty to prevent overfitting and perform variable selection.

Elastic Net Regularization:

$$\min_{\beta} \left[-\ell(\beta) + \lambda \left((1-\alpha) \frac{\|\beta\|_2^2}{2} + \alpha \|\beta\|_1 \right) \right]$$

- $\ell(\beta)$: log-likelihood of the logistic model.
- λ: overall penalty strength controlling coefficient shrinkage.
- α: mixes the two types of regularization:
  - Ridge (L2): smooth shrinkage.
  - Lasso (L1): sets some coefficients exactly to zero.

Elastic Net: Out-of-Sample Performance

Predictive performance (test set):

- Accuracy: **64.3%**
- AUC: **0.79** (strong probability ranking ability)
- Threshold tuned : $t^* = 0.6743 \rightarrow$ The model avoids overly optimistic Up predictions

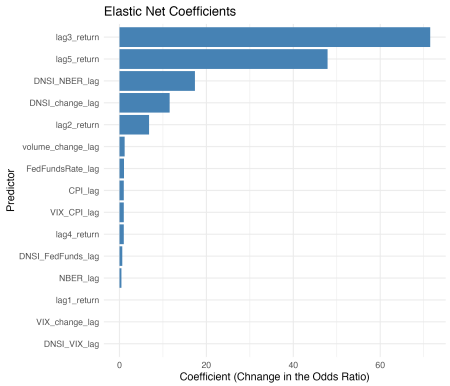
Confusion matrix:

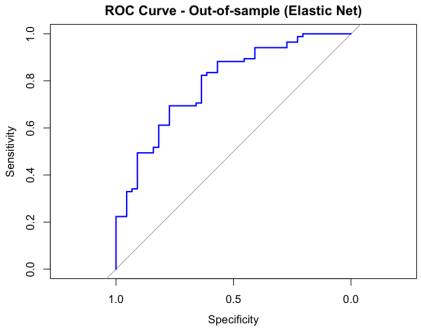
	Actual 0	Actual 1
Predicted 0	36	38
Predicted 1	8	47

Interpretation:

- **Specificity** (correct Down predictions): $36/44 \approx$ **82%** $\Rightarrow$ we correctly flag most Down months.
- **Sensitivity** (correct Up predictions): $47/85 \approx$ **55%** $\Rightarrow$ we still capture the majority of Up months.
- The model trades a small loss in raw accuracy for a much better **balance** between detecting costly Down markets and still identifying Up markets, which is desirable in an asset-management context.

Elastic Net: Selected Predictors and Probability Ranking





- Coeff. represent changes in the **odds** of an "Up" month.
- Elastic Net highlights variables with consistent signal.
- Momentum is a major driver of Up probabilities.
- Sentiment matters more during recession periods.

- **AUC = 0.79**: strong discriminative ability despite low-signal, noisy financial data.
- ROC curve well above diagonal ⇒ model ranks Up vs. Down months effectively.

Random Forest: Out-of-Sample Performance

Predictive performance (test set):

- Accuracy: **71.3%**
- AUC: **0.753** (strong non-linear discriminative ability)
- RF produces well-calibrated probability scores without threshold tuning

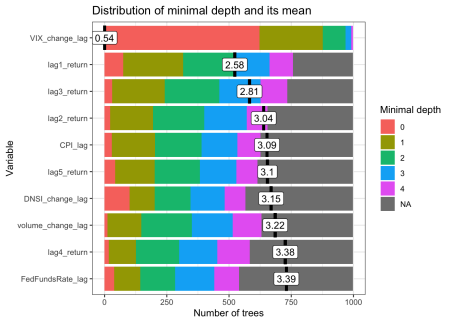
Confusion matrix:

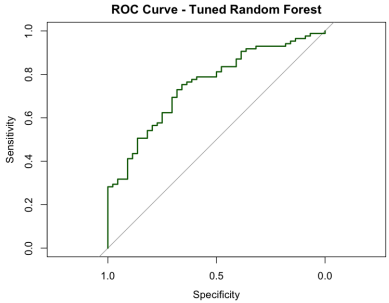
	Actual 0	Actual 1
Predicted 0	29	22
Predicted 1	15	63

Interpretation:

- **Specificity** (correct Down predictions): $29/44 \approx$ **66%**. RF detects a meaningful fraction of negative weeks.
- **Sensitivity** (correct Up predictions): $63/85 \approx$ **74%**. RF identifies most positive-return weeks.
- **Overall**: RF captures important **nonlinear return patterns**, outperforming Elastic Net in accuracy and maintaining a strong AUC.

Random Forest: Minimal Depth, Interpretation, and AUC





Explanation and Interpretation of Minimal Depth

- Min. depth measures how **early** a variable is used in the tree.
- **VIX_change_lag** is the dominant signal
- **Lagged returns** are meaningful momentum predictors.
- Overall: RF relies mainly on **volatility shocks** and **recent price dynamics**.

AUC

- **AUC = 0.753** $\Rightarrow$ strong discriminative ability and reliable ranking of Up vs. Down weeks.

Takeaway

- Random Forest effectively captures nonlinear interactions between **volatility movements** and **short-term momentum**.

Gradient Boosting Machine (GBM)

Model Intuition:

- A powerful **boosting** ensemble method, as discussed in class.
- It builds trees **sequentially**, not in parallel like Random Forest .
- Each new, simple tree (a "weak learner") is trained on the (pseudo-)residuals of the previous trees' errors.
- It's a "slow" learner that "boosts" the signal over many iterations, fitting an additive model.
- We use `distribution = "bernoulli"` for binary classification, which optimizes the deviance (log-loss).

Tuned Hyperparameters (from Time-Series CV):

- `shrinkage = 0.01` (The learning rate ν): A small value provides **regularization**, forcing the model to learn "slowly". This is ideal for noisy financial data, as it prevents overfitting and generally gives better results.
- `n.trees = 200` (The number of iterations M): This is the optimal stopping point found by CV. It is directly balanced with the small shrinkage; a slow learner ($\nu = 0.01$) requires more iterations ($M = 200$) to fit the signal.
- `interaction.depth = 1` (Tree size J): This was a key finding. Our CV selected a "stump" (a tree with one split), which restricts the entire GBM to a purely **additive model**. This suggests the predictive signal in our data is best captured by main effects, not by complex interactions.

GBM: Out-of-Sample Performance and Importance

Predictive performance (test set):

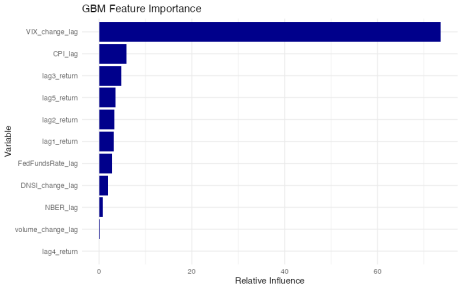
- Accuracy: **0.674**
- AUC: **0.783**
- Log-Loss: **0.5531**
- Tuned Threshold: **0.6962** (via training ROC)

Confusion matrix:

	Actual 0	Actual 1
Predicted 0	33	31
Predicted 1	11	54

Interpretation:

- **Specificity**: $33/(33+11) \approx$ **75%**
- **Sensitivity**: $54/(54+31) \approx$ **64%**
- The additive GBM model shows a good balance, effectively capturing both "Up" and "Down" markets.

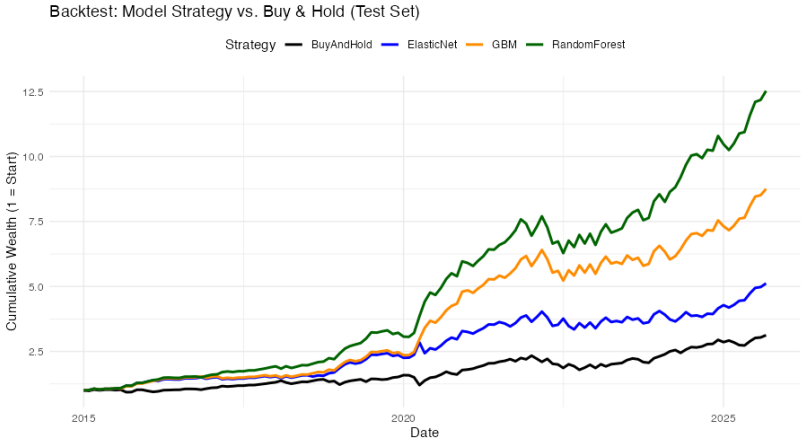


Feature Importance Interpretation:

- The model is **dominated** by a single predictor: `VIX_change_lag` (73.6% relative influence).
- The best-tuned model was additive (`interaction.depth = 1`), so it only finds main effects.
- Other macro (`CPI_lag`) and momentum (`lag3_return`) features provide minor corrective adjustments to the main volatility signal.

Strategy Backtest

- We can simulate a trading strategy to see if the statistical edge (AUC > 0.75) translates into a practical, usable result.
- This simulation uses the unseen test set and assumes no transaction costs. At the start of each month t, use the model's prediction.
- If P(Up) $>$ Tuned Threshold $\Rightarrow$ **Go Long** S&P 500 for the month. If P(Up) $<$ Tuned Threshold $\Rightarrow$ **Go Short** S&P 500 for the month.



Strategy Backtest

Motivation:

- We can quantify the visual performance from the backtest plot using standard financial metrics.
- **Annualized Sharpe Ratio:** The key metric for risk-adjusted return (Return / Volatility). Higher is better.
- **Maximum Drawdown (MaxDD):** The largest peak-to-trough loss. Measures "pain" or tail risk; smaller is better.

Strategy Performance Metrics (Test Set)

Strategy	Annualized Return	Annualized Volatility	Annualized Sharpe	Max. Drawdown
BuyAndHold	11.92	15.91	0.750	24.168
ElasticNet	16.47	15.56	1.058	17.000
RandomForest	24.77	14.61	1.696	18.434
GBM	21.44	15.05	1.425	18.434

- The Random Forest and GBM models generated the highest risk-adjusted returns (Sharpe Ratios), both significantly outperforming the "Buy and Hold" strategy.
- Impressively, all model-based strategies had a **lower Maximum Drawdown** than "Buy and Hold," suggesting they were successful at mitigating major losses.
- This confirms the models provide not just higher returns, but **smarter, risk-controlled** returns.

Final Model Selection

Full Model Performance Comparison (Test Set)

Model	Test_AUC	Test_Accuracy	Test_LogLoss	Test_Brier
Elastic Net (Tuned)	0.7904	0.6434	0.5154	0.1735
Gradient Boosting (Tuned)	0.7828	0.6744	0.5531	0.1846
Random Forest (Tuned)	0.7532	0.7132	0.5644	0.1911

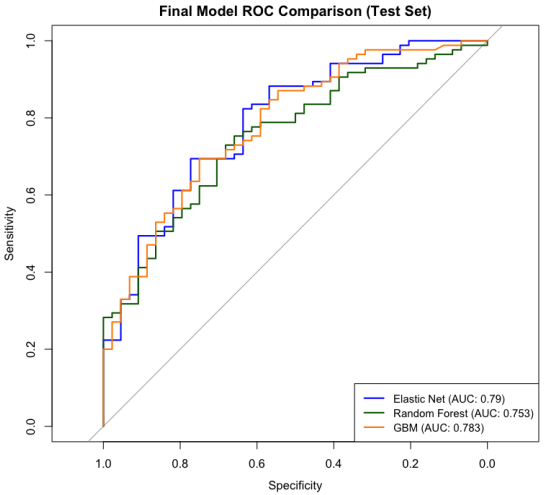
Model Selection Discussion:

- We face a classic trade-off between pure **ranking ability** and **strategy profitability**.
- **Best Ranking Model (AUC):** The **Elastic Net** (AUC=0.790) is the most reliable model for pure probabilistic discrimination.
- **Best Strategy Model (Sharpe):** The **Random Forest** (Sharpe=1.696) generated the most profitable risk-adjusted returns in our backtest.

Final Model Decision:

- For a pure **risk-management** or probability model, the **Elastic Net** is the winner.
- For a pure **trading strategy**, the **Random Forest** is the winner.
- This key insight - that the non-linear patterns (RF) were more profitable than the linear ones (ENet) - is the main finding of our analysis.

Final Model Comparison



Final Model Performance on Hold-Out Test Set			
Model	Test_AUC	Test_Accuracy	Test_LogLoss
Elastic Net (Tuned)	0.7904	0.6434	0.5154
Gradient Boosting (Tuned)	0.7828	0.6744	0.5531
Random Forest (Tuned)	0.7532	0.7132	0.5644

Final performance metrics on the hold-out test set

- All models performed significantly better than chance (all AUC greater than 0.75).
- The linear **Elastic Net** had the highest Test AUC (0.790), narrowly beating the GBM (0.783).
- The **Random Forest** had the highest Test Accuracy (0.713), but the lowest AUC.
- This suggests the regularized linear model was best at ranking probabilities, while the RF was best at classification after threshold tuning.

Conclusion & Final Insights

What We Did (Summary):

- We built, tuned, and tested three models (GLM, Bagging, Boosting) to predict S&P 500 direction.
- We used a robust **rolling-window CV** to respect the time-series data and avoid look-ahead bias.
- We addressed **class imbalance** by tuning with AUC and finding an optimal probability threshold.

What We Found (The Key Insight):

- **Monthly market direction is predictable** (all models had Test AUC > 0.75).
- The best linear model (Elastic Net) had the best ranking ability (AUC $= 0.790$).
- The best non-linear model (Random Forest) generated the most profit (Sharpe $= 1.696$).
- **Insight:** This implies that while linear signals (momentum, VIX) are the most consistent predictors, the **non-linear interactions** (e.g., VIX + sentiment) captured by the Random Forest, while perhaps less frequent, lead to more explosive, profitable moves.

Implications & Limitations:

- A simple long/short strategy based on the Random Forest model's non-linear signals would have significantly outperformed a "Buy and Hold" strategy on a risk-adjusted basis.
- **Limitations:** This backtest is idealized. It does not include transaction costs or slippage, and it assumes the model's relationships will remain stable in the future.

Thank you!

Questions & Discussion

Appendix: Full Probabilistic Performance

Motivation:

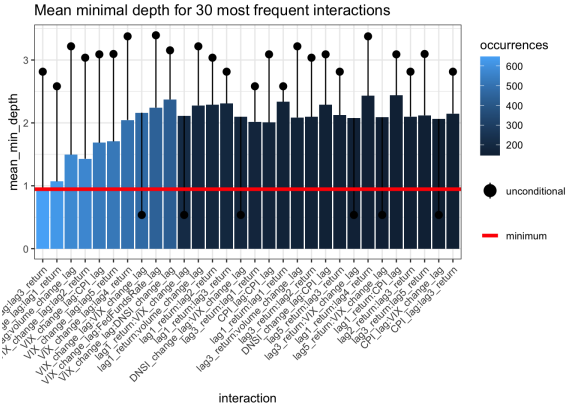
- Proper scoring rules like Brier Score and Log-Loss assess the *quality of the probabilities* themselves, not just the final classification.
- **Log-Loss** (Deviance) heavily penalizes confident wrong answers (e.g., predicting 0.9 when the answer is 0). This is the metric our GBM optimized.
- **Brier Score** is the Mean Squared Error of the probability, rewarding well-calibrated forecasts.

Full Model Performance Comparison (Test Set)

Model	Test_AUC	Test_Accuracy	Test_LogLoss	Test_Brier
Elastic Net (Tuned)	0.7904	0.6434	0.5154	0.1735
Gradient Boosting (Tuned)	0.7828	0.6744	0.5531	0.1846
Random Forest (Tuned)	0.7532	0.7132	0.5644	0.1911

Appendix: RF Interactions

- Our key finding was that the non-linear RF model (Sharpe=1.696) was more profitable than the linear Elastic Net (Sharpe=1.058).
- **Why?** The RF model captures *interaction effects* that the additive ENet and GBM (J=1) models miss.

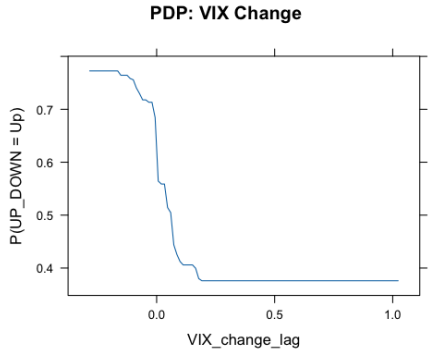


- This plot shows the 30 most frequent interactions. The strongest interactions are between **momentum** variables (e.g., `lag1_return:lag3_return`) and between **volatility and momentum** (e.g., `VIX_change_lag:lag1_return`).
- **Conclusion:** This proves the RF model is capturing complex, non-linear rules (e.g., the effect of momentum depends on the VIX). This non-linearity is the source of its superior trading performance.

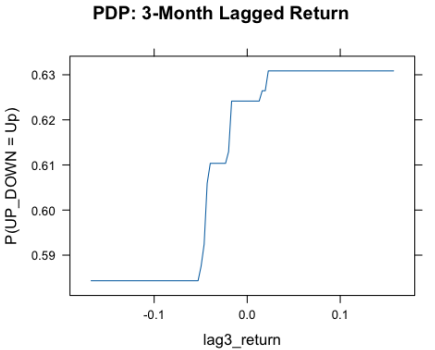
Appendix: GBM Partial Dependence Plots

Motivation:

- As discussed in class, Partial Dependence Plots (PDPs) are used to interpret the model by showing the marginal effect of a feature on the prediction, holding other features constant.
- This shows *how* our most important variables influence the probability of an "Up" month.



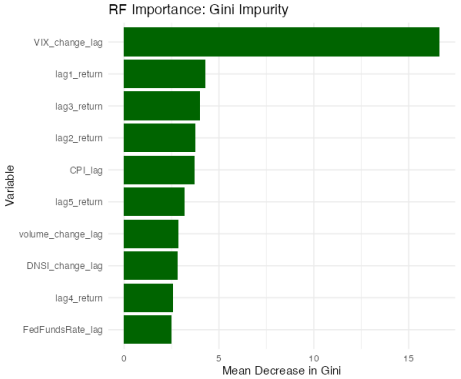
Insight: The probability of an "Up" month is highest (around 0.7) when the VIX has a small *negative* change (slight calming). A large VIX spike (large positive change) dramatically *decreases* the probability of an "Up" month.

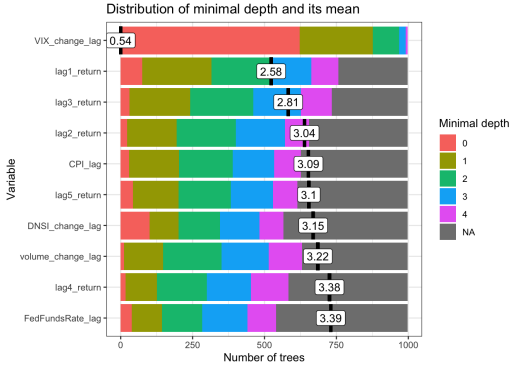


Insight: The model found a momentum effect. The probability of an "Up" month increases as the 3-month lagged return increases from negative to positive, consistent with financial theory.

Appendix: RF Importance (Minimal Depth vs. Gini)

- **Gini Importance** (left) measures the average gain in node purity from splitting on a variable. It is fast but can be biased towards continuous or high-cardinality features.
- **Minimal Depth** (right, from presentation) measures how early a variable is used to split. Variables that split early are generally more important. This is often considered more robust.





- Both metrics agree on the top 2 predictors: **VIX_change_lag** and **lag1_return**.
- This consistency gives us high confidence that these are the most powerful and reliable signals in our non-linear model.

Appendix: Hyperparameter Tuning Grids (Elastic Net)

Elastic Net (GLM) Grid:

- `alpha (Mixing Parameter)`: {0, 0.1, 0.2, ..., 0.9, 1.0}
  - Tests the **entire spectrum** of regularization.
  - `alpha = 0` is pure Ridge (good for multicollinearity).
  - `alpha = 1` is pure Lasso (good for variable selection).
  - In between is the Elastic Net, which balances both. Your optimal $\alpha = 0.1$ suggests a model that is 90 percent Ridge and 10 percent Lasso.
- `lambda (Penalty Strength)`: 40 log-spaced values from 10^{-4} to 10.
  - A log-spaced grid is critical because the penalty's effect is not linear.
  - This wide range tests models from a standard (un-penalized) GLM (low λ) up to a null (intercept-only) model (high λ).

Total combinations tested per CV fold: 11 (alphas) $\times$ 40 (lambdas) = 440

Insight:

- The tuning process is thorough. It selected $\alpha = 0.1$, confirming that our data benefits from a Ridge-dominant model to handle multicollinearity, while still using a small amount of Lasso for variable selection.

Appendix: Hyperparameter Tuning Grids

Random Forest Grid:

- `mtry (vars. per split)`: {3, 4, 6, 7, 11}
  - Based on $p = 11$ predictors. Grid extends from the defaults ($\sqrt{p} \approx 3$, $p/3 \approx 4$) up to $p = 11$ to find the optimal level of tree de-correlation.
- `min.node.size (leaf size)`: {2, 5, 10, 20}
  - Controls tree depth and regularization.
- `max.depth`: {5, 10, 15, 20, Unlimited (NA)}
  - Another regularization control on tree complexity.
- `sample.fraction`: {0.6, 0.7, 0.8, 0.9}

Total combinations tested via Time-Series CV: 400

Gradient Boosting (GBM) Grid:

- `n.trees` (M): {100, 200, 300}
  - The number of boosting iterations (weak learners).
- `interaction.depth` (J): {1, 2, 3}
  - Tests an **additive model (J=1)** against models with two-way (J=2) and three-way (J=3) interactions.
- `shrinkage` (ν): {0.01, 0.1}
  - The learning rate. Tests a slow, regularized rate (0.01) vs. a faster, standard rate (0.1).

Total combinations tested via Time-Series CV: 18

References

- Cochrane, John H. (2005). *Asset Pricing*. Princeton University Press.
- Federal Reserve Economic Data (FRED). *St. Louis Fed*. Retrieved 2025.
- Federal Reserve Bank of San Francisco. *Daily News Sentiment Index*. Retrieved 2025.
- Yahoo Finance. *S&P 500 ($\hat{GSPC}$) & CBOE VIX ($\hat{VIX}$) Data*. Retrieved 2025.
- Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software, 33*(1), 1-22.
- Wright, M. N. & Ziegler, A. (2017). ranger: A Fast Implementation of Random Forests... *Journal of Statistical Software, 77*(1), 1-17.
- Ridgeway, G. (2007). Generalized Boosted Models: A guide to the gbm package.