Stevens Institute of Technology

Statistical Learning Final Project

Predicting Cross-Sectional Stock Returns with Machine Learning: Evidence from Monthly U.S. Equity Data

By

Sakshi Solanki

Harshil Patel

Kunal Singh

December 2025

Abstract

This project examines whether machine learning can improve cross-sectional stock return prediction and support economically meaningful trading strategies, using 4.09 million stock-month observations from 1957 to 2021 and 176 firm-level and macroeconomic predictors. Six models are compared—OLS, Ridge, Lasso, Random Forests, Gradient Boosting, and Neural Networks—within a strictly chronological design separating training (1957–1995), validation (1995–2008), and out-of-sample testing (2008–2021) to avoid look-ahead bias.

Predictors include return and volatility characteristics (momentum, idiosyncratic volatility, beta), accounting-based firm characteristics (size, book-to-market, leverage, profitability, investment), 73 industry dummies, and eight macroeconomic variables (term spread, default spread, Treasury rate, valuation ratios, stock-market variance). Preprocessing imputes missing values via cross-sectional medians, standardizes continuous predictors using training-set statistics, and encodes industries as one-hot dummies. Exploratory analysis confirms returns are heavy-tailed and non-normal, individual predictors correlate weakly with returns ($|r| < 0.03$), and multicollinearity motivates regularization.

Performance is evaluated along two dimensions. Statistical accuracy uses out-of-sample R^2, MSE, and MAE in 2008–2021. Economic value translates predictions into portfolios by ranking stocks monthly, selecting the top 100, and computing realized returns, Sharpe ratios, and cumulative performance over 156 test months.

Results show monthly returns remain extremely hard to forecast: all models produce negative out-of-sample R^2, although regularized models achieve modest error reductions. However, when forecasts rank stocks into portfolios, machine learning becomes substantially more useful: Gradient Boosting and Lasso portfolios deliver Sharpe ratios of 0.372 and 0.361 respectively—78% and 73% improvements over OLS (0.209)—with cumulative returns of 55% and 40% versus 15% for the baseline. Feature-importance analysis reveals macroeconomic variables (volatility, term and default spreads) dominate the signal, suggesting machine learning extracts weak, regime-dependent ranking signals rather than precise point predictions.

Table of Contents

1. Introduction

1.1 Research Context

Forecasting stock returns is central to empirical finance. Key developments in the literature include:

- Fama and French (1993) established that size and value predict returns
- Subsequent research identified momentum (Jegadeesh & Titman, 1993), profitability, and investment patterns
- Many predictors weaken out-of-sample—a phenomenon termed "anomaly decay"
- Machine learning offers new tools to handle high-dimensional data (176 predictors), capture non-linear relationships, and detect complex interactions

1.2 Core Research Questions

This project addresses three interconnected questions:

1. **Predictability:** To what extent can firm characteristics and macro variables predict next month returns?
2. **Model Comparison:** How do linear models (OLS, Ridge, Lasso) compared to non-linear models (Random Forests, Gradient Boosting, Neural Networks) in out-of-sample accuracy and portfolio performance?
3. **Economic Value:** Can forecast-based portfolios achieve attractive risk-adjusted returns despite near-zero R^2?

The third question reflects practical reality: ranking stocks matters more than predicting exact returns.

1.3 Contribution

We implement an end-to-end pipeline with:

- Rigorous data preprocessing and exploratory analysis
- Chronological train-validation-test splits (no look-ahead bias)
- Hyperparameter tuning on separate validation set
- Dual evaluation: statistical R^2 plus economic Sharpe ratio
- Sixteen detailed visualizations documenting all findings

Key insights: weak R² doesn't preclude strong portfolio performance; model choice significantly affects returns; macroeconomic variables dominate the predictive signal.

2. Data Description and Preprocessing

2.1 Dataset Overview

Dataset Characteristics:

- **Time span:** 1957–2021 (64 years)
- **Observations:** 4,089,903 stock-month observations
- **Unique stocks:** 32,655
- **Monthly periods:** 779
- **Target variable:** Next-month return (RET)
- **Predictors:** 176 total (103 original + 73 SIC2 industry dummies)

Four Predictor Categories:

1. **Return and Volatility Characteristics:** Momentum (mom1m, mom6m, mom12m), volatility (idiovol, retvol), beta measures—capture individual stock price dynamics
2. **Firm Characteristics:** Size, book-to-market, leverage, profitability, investment, sales/earnings growth—derive from Compustat with documented predictive power
3. **Industry Indicators:** 73 SIC2 dummy variables—capture sector rotation and fixed effects
4. **Macroeconomic Variables:** Term spread, default spread, Treasury rate, dividend/earnings ratios, stock variance, equity issuance—track risk premia and market conditions

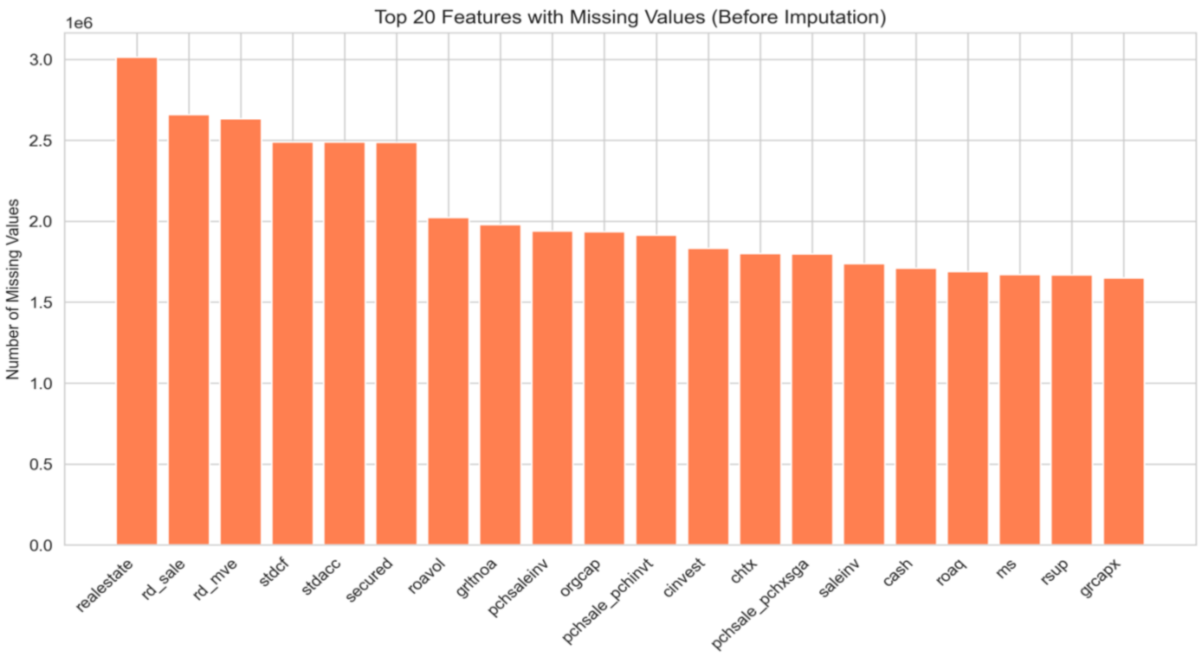
2.2 Data Quality and Preprocessing

Missing Data Challenge:

- Some accounting variables have over 3 million missing observations (73% of panel)
- Reflects sparse reporting for specialized items (R&D, real estate holdings)

Preprocessing Strategy:

- Impute missing values using **cross-sectional medians within each month**
- Limits look-ahead bias while maximizing observations
- Convert all numeric fields to float64
- Standardize continuous predictors (zero mean, unit variance) using **training data only**
- Convert SIC2 codes to 73 one-hot dummies



2.3 Exploratory Data Analysis

Distribution of Monthly Returns:

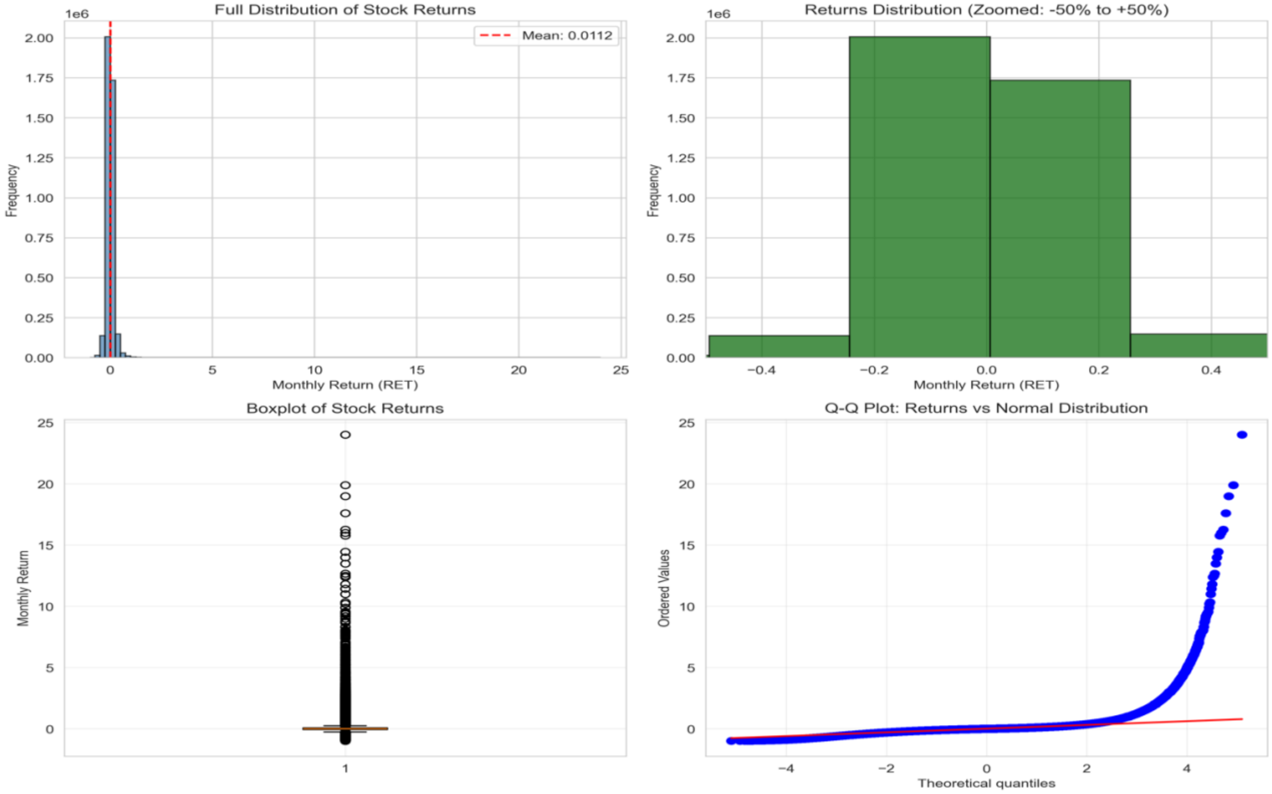


Figure reveals:

- Heavy right skewness with extreme outliers (range: -90% to +2000%)
- Non-normal Q-Q plot departure from 45° line
- High kurtosis indicating significant tail risk
- Sample mean: 1.12% monthly (not representative due to outliers)

Time-Series Behavior:

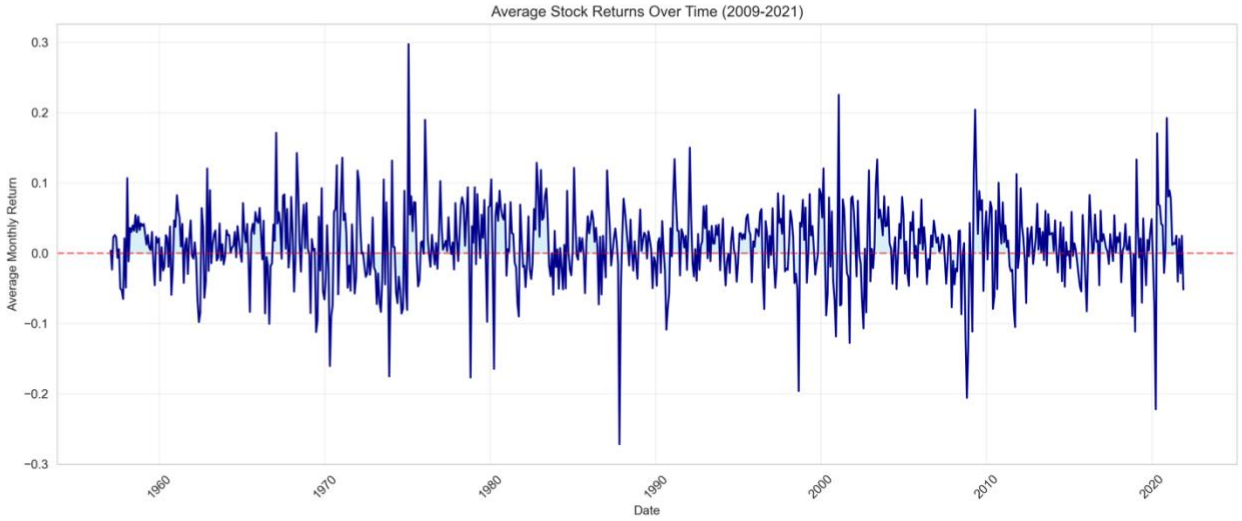


Figure shows:

- Pronounced mean-reversion and regime changes (1957-2021)
- Volatility clustering around major crises: 1987 crash, 2000 Dotcom burst, 2008 financial crisis, 2020 COVID crash
- Long-run positive drift during bull markets (especially post-2009)
- Time-varying expected returns driven by economic conditions

Feature-Return Correlations:

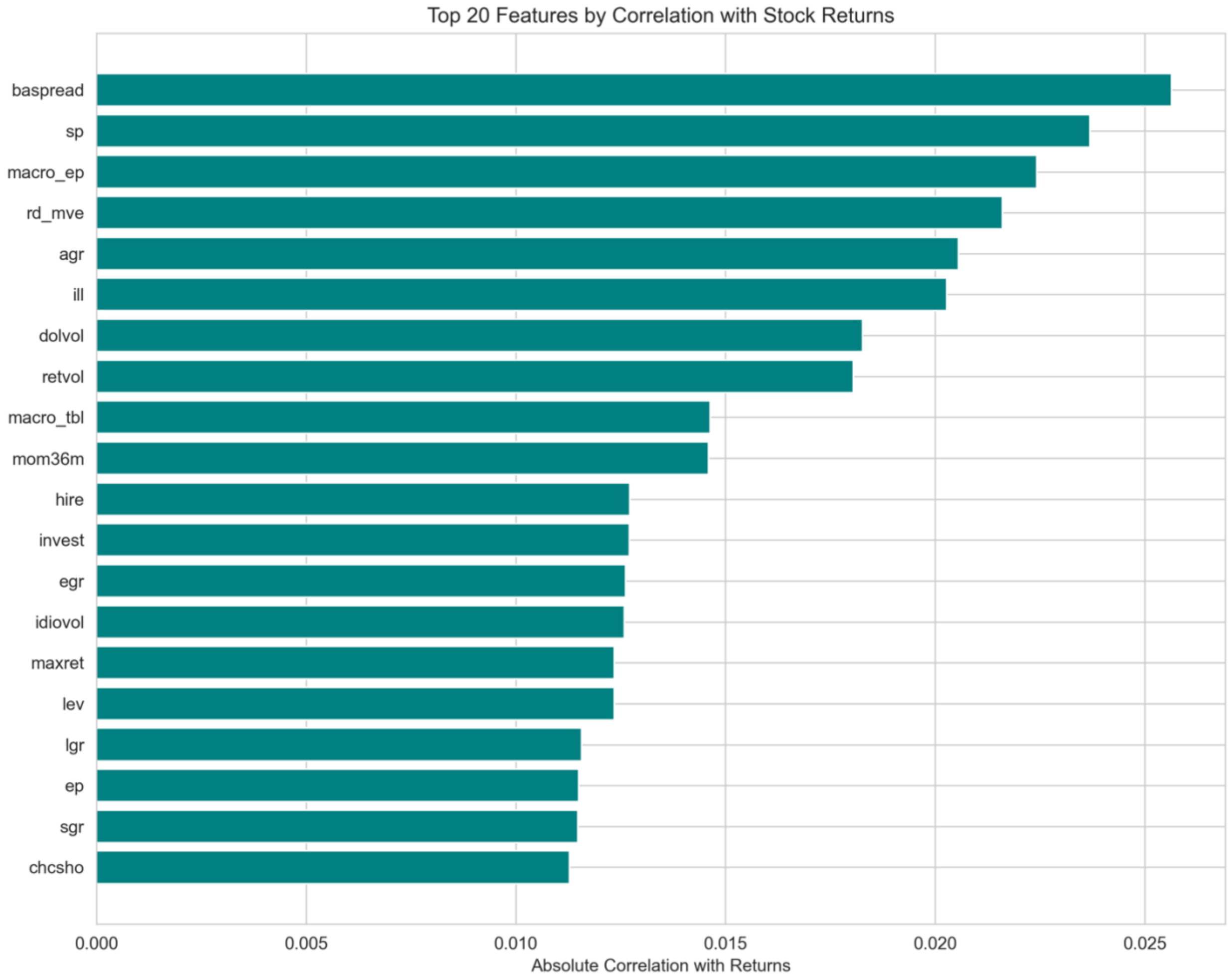
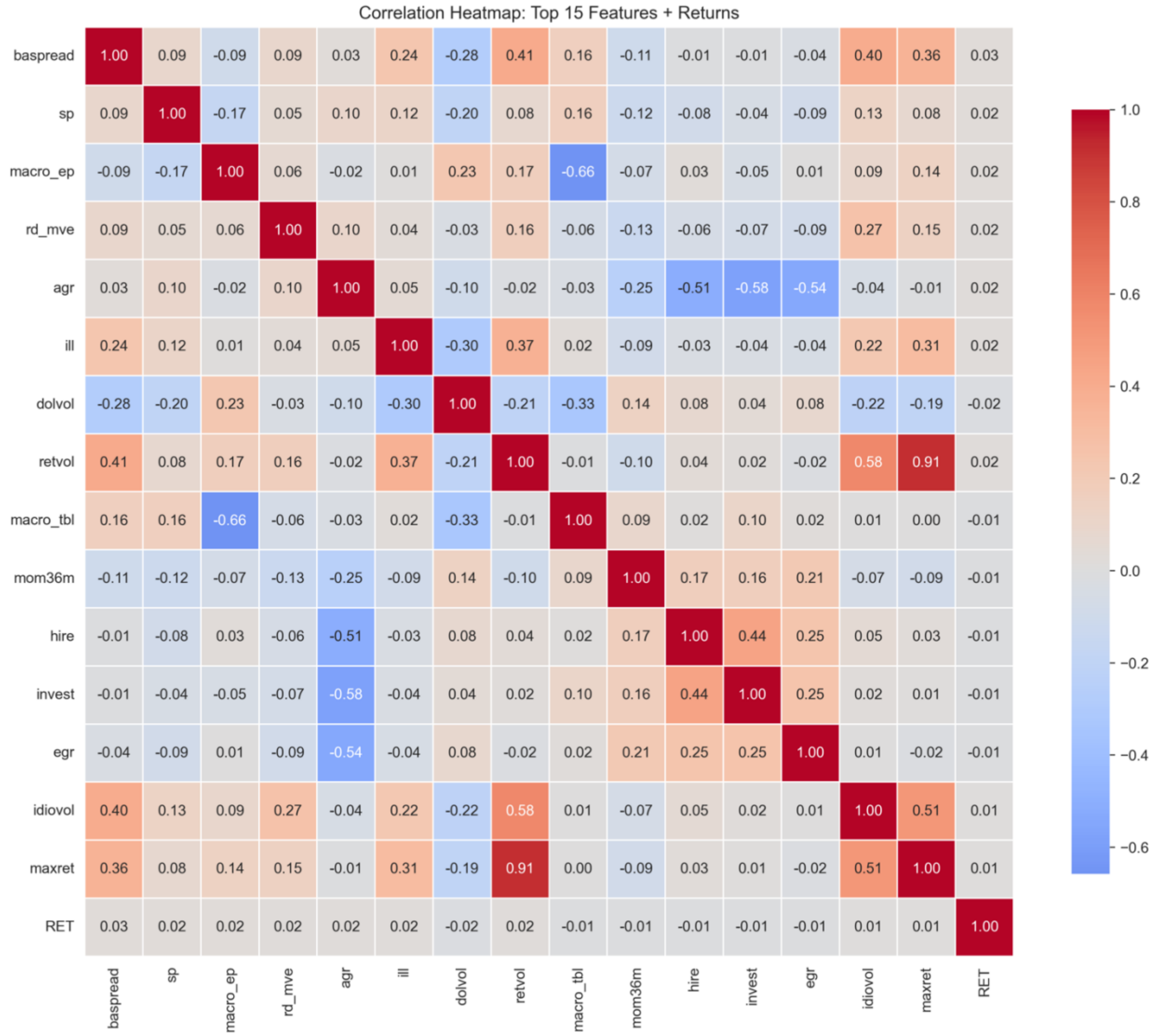


Figure demonstrates:

- Individual predictors correlate weakly with returns (max |r| ≈ 0.025)
- Best predictor explains only 0.06% of variance individually
- Top correlators: baspread, sp, macro_ep, volatility measures

Multicollinearity Among Predictors:


Correlation Heatmap: Top 15 Features + Returns

	baspread	sp	macro_ep	rd_mve	agr	ill	dolvol	retvol	macro_tbl	mom36m	hire	invest	egr	idiovol	maxret	RET
baspread	1.00	0.09	-0.09	0.09	0.03	0.24	-0.28	0.41	0.16	-0.11	-0.01	-0.01	-0.04	0.40	0.36	0.03
sp	0.09	1.00	-0.17	0.05	0.10	0.12	-0.20	0.08	0.16	-0.12	-0.08	-0.04	-0.09	0.13	0.08	0.02
macro_ep	-0.09	-0.17	1.00	0.06	-0.02	0.01	0.23	0.17	-0.66	-0.07	0.03	-0.05	0.01	0.09	0.14	0.02
rd_mve	0.09	0.05	0.06	1.00	0.10	0.04	-0.03	0.16	-0.06	-0.13	-0.06	-0.07	-0.09	0.27	0.15	0.02
agr	0.03	0.10	-0.02	0.10	1.00	0.05	-0.10	-0.02	-0.03	-0.25	-0.51	-0.58	-0.54	-0.04	-0.01	0.02
ill	0.24	0.12	0.01	0.04	0.05	1.00	-0.30	0.37	0.02	-0.09	-0.03	-0.04	-0.04	0.22	0.31	0.02
dolvol	-0.28	-0.20	0.23	-0.03	-0.10	-0.30	1.00	-0.21	-0.33	0.14	0.08	0.04	0.08	-0.22	-0.19	-0.02
retvol	0.41	0.08	0.17	0.16	-0.02	0.37	-0.21	1.00	-0.01	-0.10	0.04	0.02	-0.02	0.58	0.91	0.02
macro_tbl	0.16	0.16	-0.66	-0.06	-0.03	0.02	-0.33	-0.01	1.00	0.09	0.02	0.10	0.02	0.01	0.00	-0.01
mom36m	-0.11	-0.12	-0.07	-0.13	-0.25	-0.09	0.14	-0.10	0.09	1.00	0.17	0.16	0.21	-0.07	-0.09	-0.01
hire	-0.01	-0.08	0.03	-0.06	-0.51	-0.03	0.08	0.04	0.02	0.17	1.00	0.44	0.25	0.05	0.03	-0.01
invest	-0.01	-0.04	-0.05	-0.07	-0.58	-0.04	0.04	0.02	0.10	0.16	0.44	1.00	0.25	0.02	0.01	-0.01
egr	-0.04	-0.09	0.01	-0.09	-0.54	-0.04	0.08	-0.02	0.02	0.21	0.25	0.25	1.00	0.01	-0.02	-0.01
idiovol	0.40	0.13	0.09	0.27	-0.04	0.22	-0.22	0.58	0.01	-0.07	0.05	0.02	0.01	1.00	0.51	0.01
maxret	0.36	0.08	0.14	0.15	-0.01	0.31	-0.19	0.91	0.00	-0.09	0.03	0.01	-0.02	0.51	1.00	0.01
RET	0.03	0.02	0.02	0.02	0.02	0.02	-0.02	0.02	-0.01	-0.01	-0.01	-0.01	-0.01	0.01	0.01	1.00

Figure reveals:

- Strong correlations among volatility measures (r ≈ 0.9)
- Macro variables moderately correlated (r ≈ 0.3–0.7)
- Momentum measures correlated (r ≈ 0.3–0.5)
- Motivates regularization (Ridge, Lasso) and flexible methods (ensembles, neural networks)

Chronological Data Split:

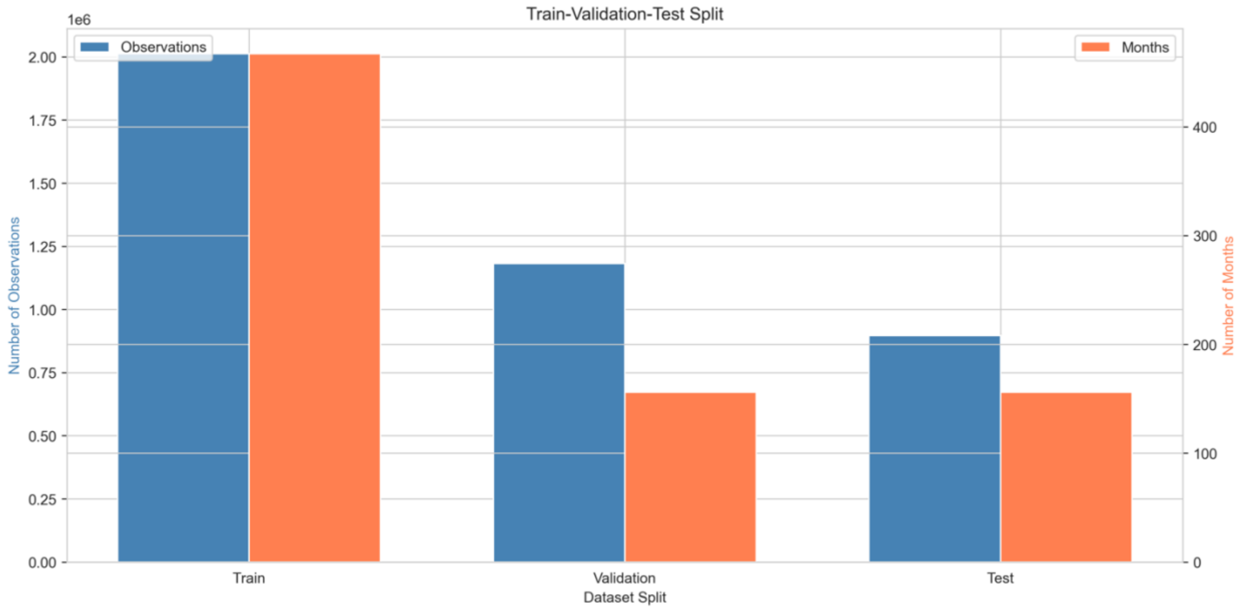


Figure shows three distinct periods:

- **Training:** 467 months (1957–1995)
- **Validation:** 156 months (1995–2008)
- **Test:** 156 months (2008–2021)
- Ensures purely forward-looking predictions (no look-ahead bias)

3. Methodology: Models and Evaluation

3.1 Data Preparation

Standardization Process:

- Compute mean and standard deviation on **training set only**
- Apply same transformation to validation and test sets
- Prevents information leakage
- SIC2 codes converted to 73 one-hot dummies

3.2 Six Predictive Models

1. OLS Regression (Baseline)

- Minimizes sum of squared residuals with no regularization
- Fast, interpretable benchmark
- Sensitive to multicollinearity and outliers

2. Ridge Regression

- L2 regularization: penalty $\lambda\sum(\text{coefficient}^2)$
- Hyperparameter: $\alpha = 1.0$ (selected via validation CV)
- Shrinks coefficients uniformly; retains all features
- Stabilizes predictions under multicollinearity

3. Lasso Regression

- L1 regularization: penalty $\lambda\sum|\text{coefficient}|$
- Hyperparameter: $\alpha = 1.0$
- Pushes some coefficients to zero (implicit feature selection)
- Sparse, interpretable solutions

4. Random Forest

- Ensemble of 100 decision trees (bootstrap aggregating)
- Each tree grown to full depth on bootstrap samples
- Averages predictions across trees
- Captures non-linear relationships and interactions
- Risk: overfitting in noisy financial data

5. Gradient Boosting

- 100 trees grown sequentially
- Each tree fits residuals of previous trees
- Shallow trees (max_depth=3), learning_rate=0.1
- Powerful but computationally intensive (~3500 seconds training)
- High overfitting risk

6. Neural Network

- 3 hidden layers: neurons
- ReLU activations, dropout=0.2
- Adam optimizer (lr=0.001), early stopping on validation loss
- Fast training (~72 seconds)
- Regularized by dropout and early stopping

Training Time Comparison:

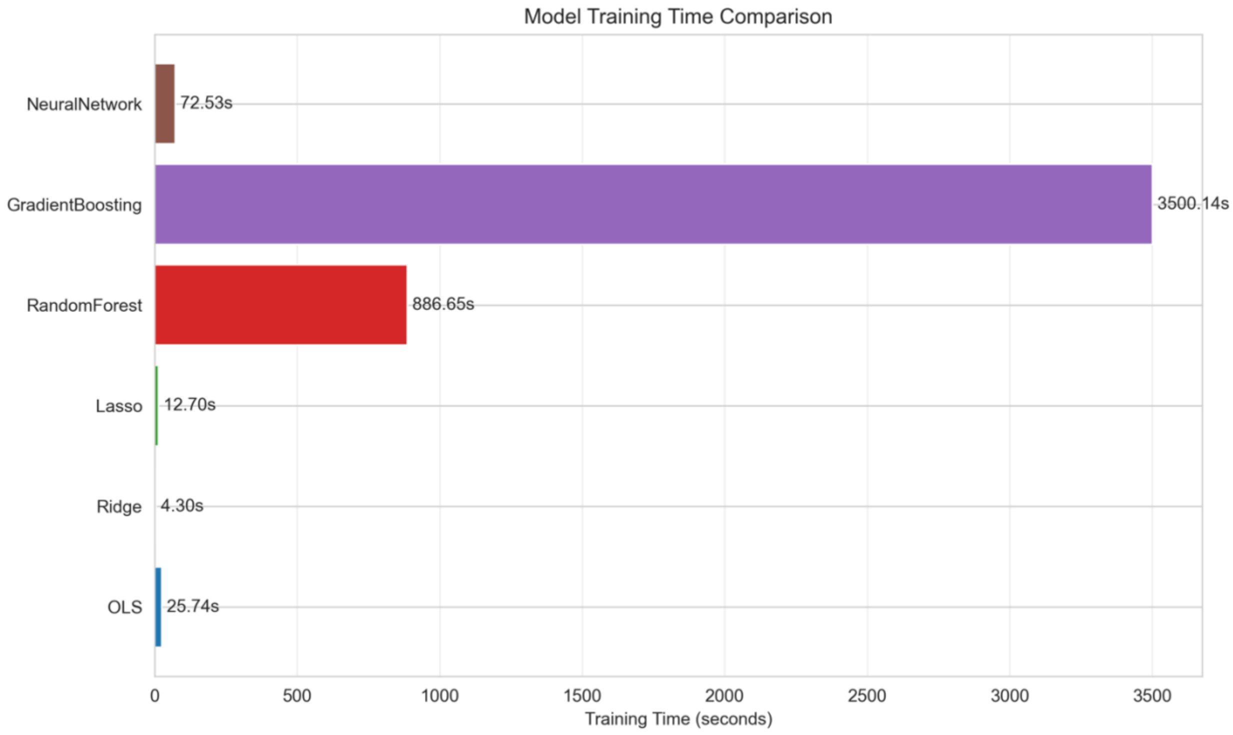


Figure shows:

- OLS/Ridge: <30 seconds
- Neural Network: 72 seconds
- Gradient Boosting: 3500 seconds (~58 minutes)

3.3 Evaluation Framework

Statistical Performance Metrics:

1. **R^2 (Coefficient of Determination):** $R^2 = 1 - (SSres / SStot)$
   - Negative R^2 means model performs worse than predicting mean return
2. **MSE (Mean Squared Error):** Penalizes large errors quadratically
3. **MAE (Mean Absolute Error):** More robust to outliers

Economic Performance Metrics:

Portfolio construction procedure (repeated for all 156 test months):

1. Rank stocks by predicted return
2. Select top 100 stocks
3. Form equal-weight portfolio (w = 1/100 for each stock)
4. Compute realized monthly return

From the 156-month return series, calculate:

- **Average Monthly Return:** $\hat{\mu} = (1/156) \sum r_t$
- **Volatility:** $\hat{\sigma}$ = standard deviation of returns
- **Sharpe Ratio:** $S = \hat{\mu}/\hat{\sigma}$ (zero risk-free rate assumed)
- **Cumulative Return:** $\prod(1+r_t) - 1$

This ranking-based approach reflects reality: predicting exact returns is extraordinarily difficult, but relative ranking is more tractable.

4. Results: Predictive Performance

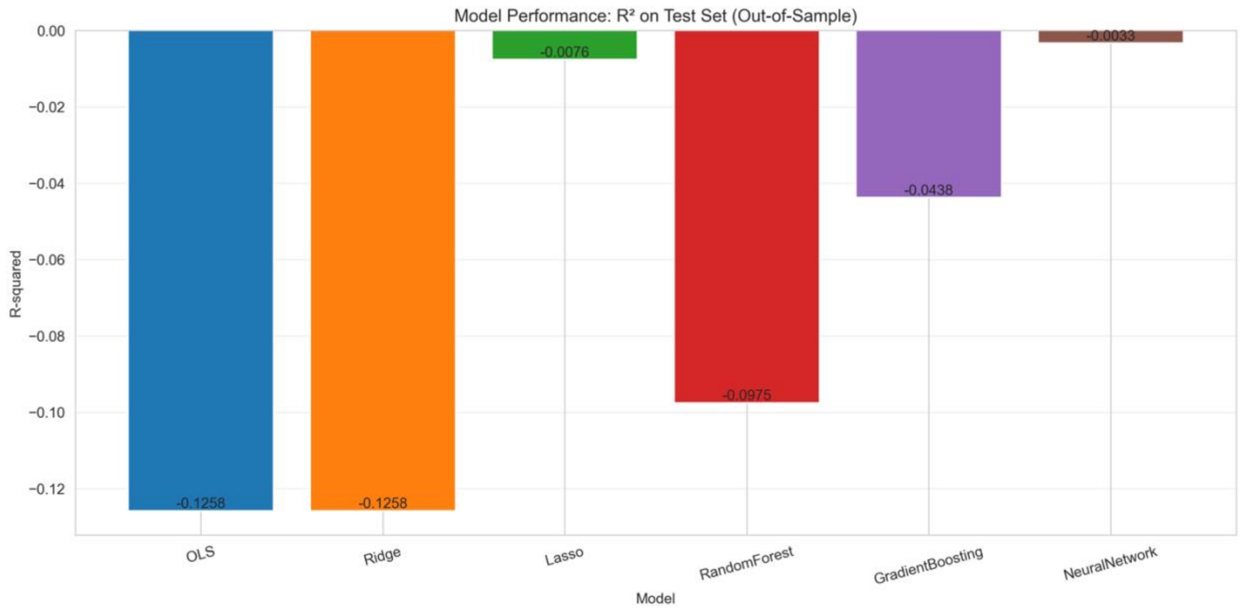
4.1 Test-Set Statistical Accuracy

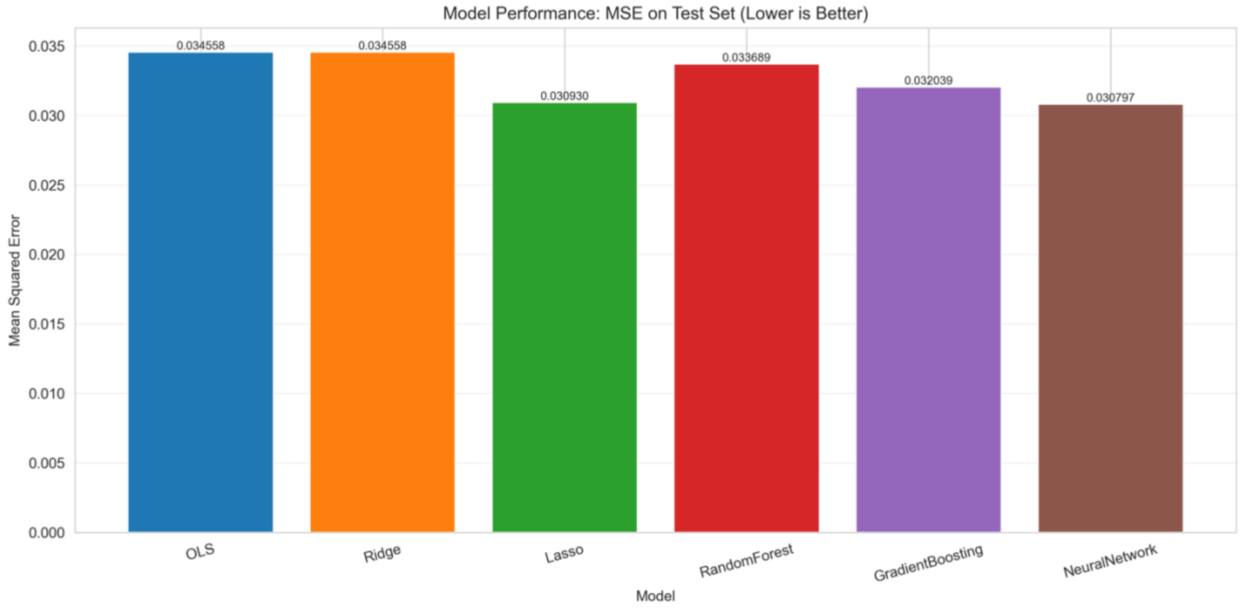
Model	**R²**	**MSE**	**MAE**
OLS	-0.1258	0.03456	0.1168
Ridge	-0.1258	0.03456	0.1168
Lasso	**-0.0076**	**0.03093**	**0.0996**
Random Forest	-0.0975	0.03369	0.1007
Gradient Boosting	**-0.0438**	**0.03204**	**0.0982**
Neural Network	**-0.0033**	**0.03080**	**0.0964**

Table 1: Test-Set Performance Metrics (2008–2021)

Key Findings:

- All models achieve negative R^2 (−0.126 to −0.003)
- Confirms individual return prediction is extremely difficult
- However, MSE improvements are meaningful:
  - Lasso: 10.6% reduction vs OLS
  - Gradient Boosting: 7.2% reduction
  - Neural Network: 10.8% reduction





Figures 9-10 visually confirm: no model achieves positive R², but Neural Network and Lasso have lowest prediction errors.

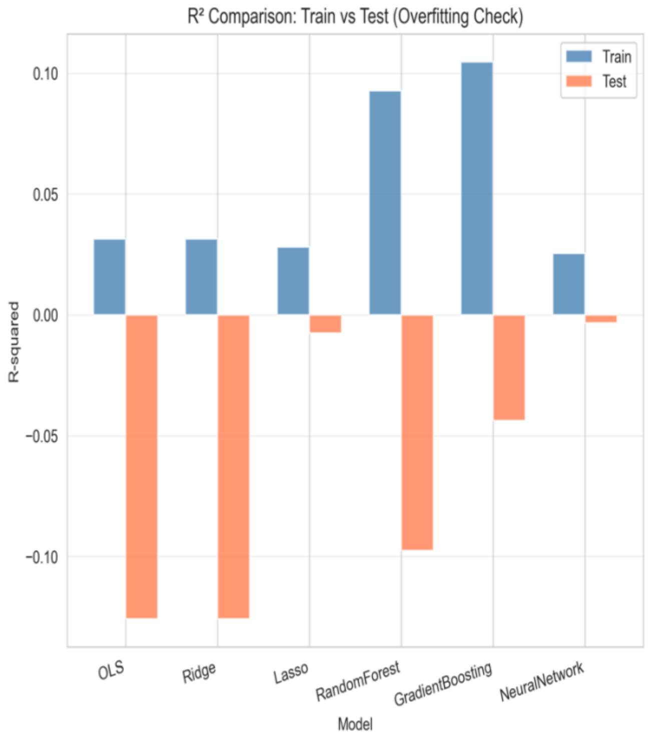
4.2 Overfitting Analysis

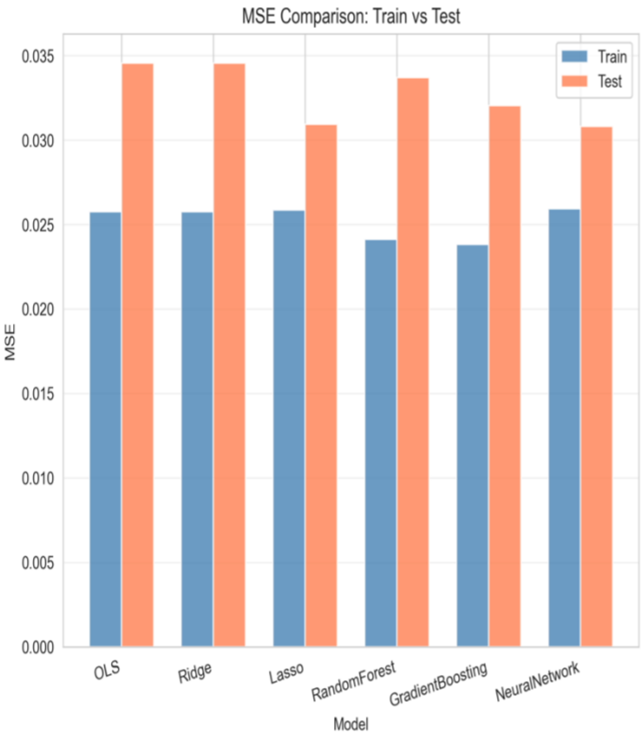
Train-Test Performance Gaps:

Model	**Train R²**	**Test R²**	**Gap**
Gradient Boosting	0.105	-0.044	**0.149**
Random Forest	0.093	-0.098	**0.190**
Neural Network	0.026	-0.003	0.028
Lasso	0.028	-0.008	0.003
OLS/Ridge	0.032	-0.126	0.157

Key Insights:

- Tree-based models show large gaps (fit training noise)
- Lasso and Neural Network generalize far better
- L1 regularization and early stopping effectively control overfitting
- Production deployment requires continuous monitoring and retraining





4.3 Residual Diagnostics

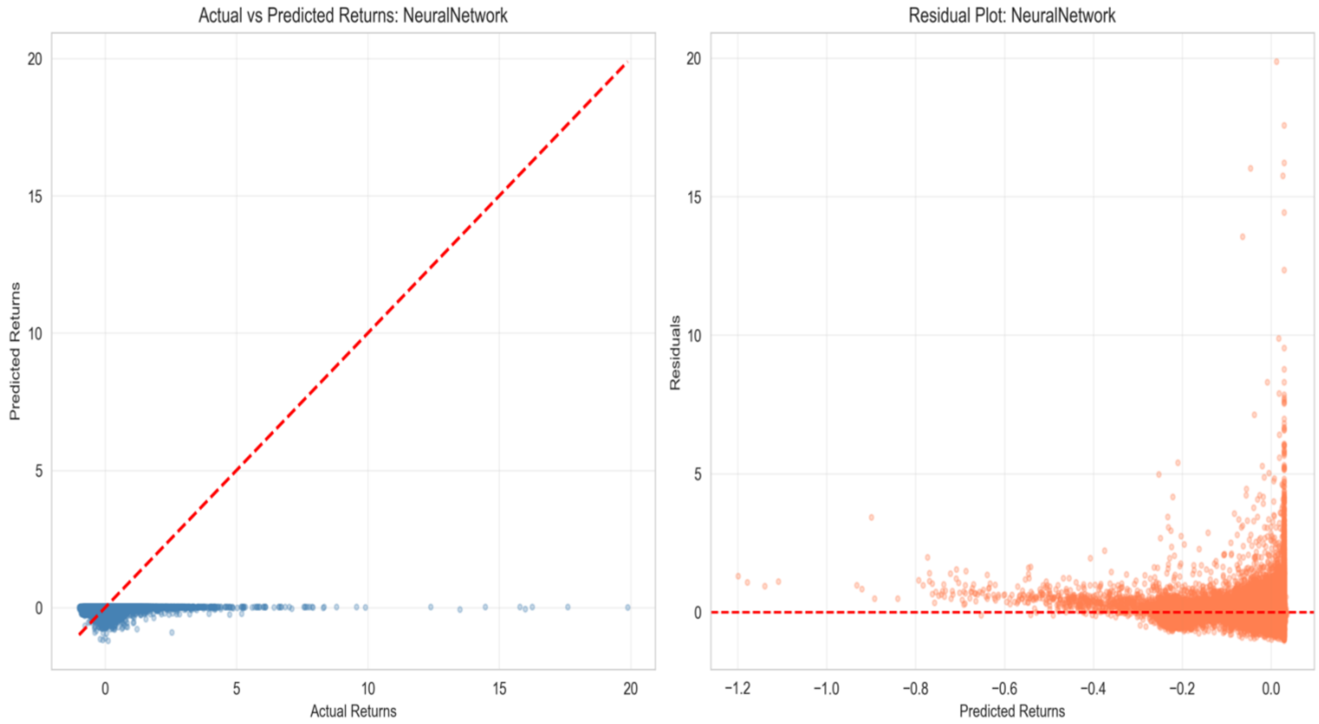


Figure 11 reveals:

- **Top panel:** Predictions cluster near zero; actual returns span −1.2 to +20
- Most points fall below 45° line (conservative prediction bias)
- **Bottom panel:** Heteroskedasticity (variance increases at extremes)
- Residuals non-normally distributed (fat-tailed)

Violations of classical regression assumptions suggest quantile regression or robust loss functions (Huber) may improve results.

5. Results: Portfolio Performance

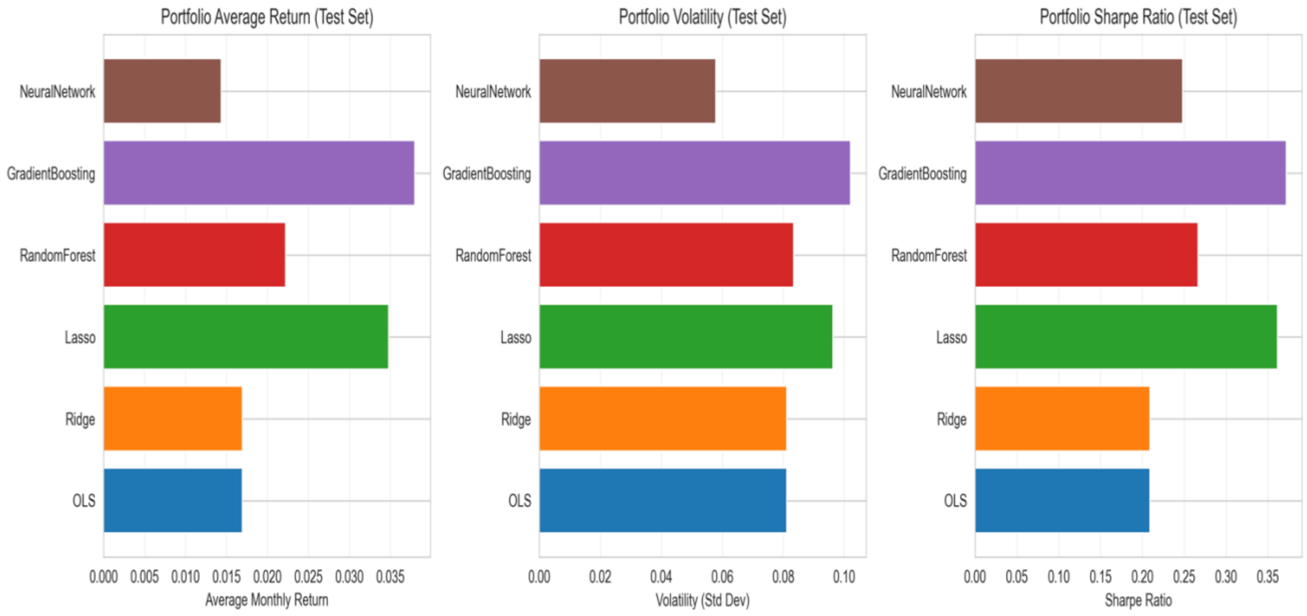
5.1 Risk-Adjusted Returns

Model	Avg Monthly Return	Volatility	Sharpe Ratio	Cumulative Return
OLS	1.70%	8.11%	0.209	~15%
Ridge	1.70%	8.11%	0.209	~15%
Lasso	**3.48%**	**9.63%**	**0.361**	**~40%**
Random Forest	2.22%	8.34%	0.267	~25%
Gradient Boosting	**3.80%**	**10.21%**	**0.372**	**~55%**
Neural Network	1.44%	5.79%	0.248	~18%

Table 2: Portfolio Performance Metrics (Test Period)

Key Findings:

- **Gradient Boosting:** 0.372 Sharpe (78% improvement over OLS)
- **Lasso:** 0.361 Sharpe (73% improvement)
- Over 156 months:
  - Gradient Boosting compounds to (1.038) ^156 ≈ 4.55x
  - OLS compounds to (1.017) ^156 ≈ 2.14x
  - 2.10% monthly edge creates 2.4x terminal wealth difference



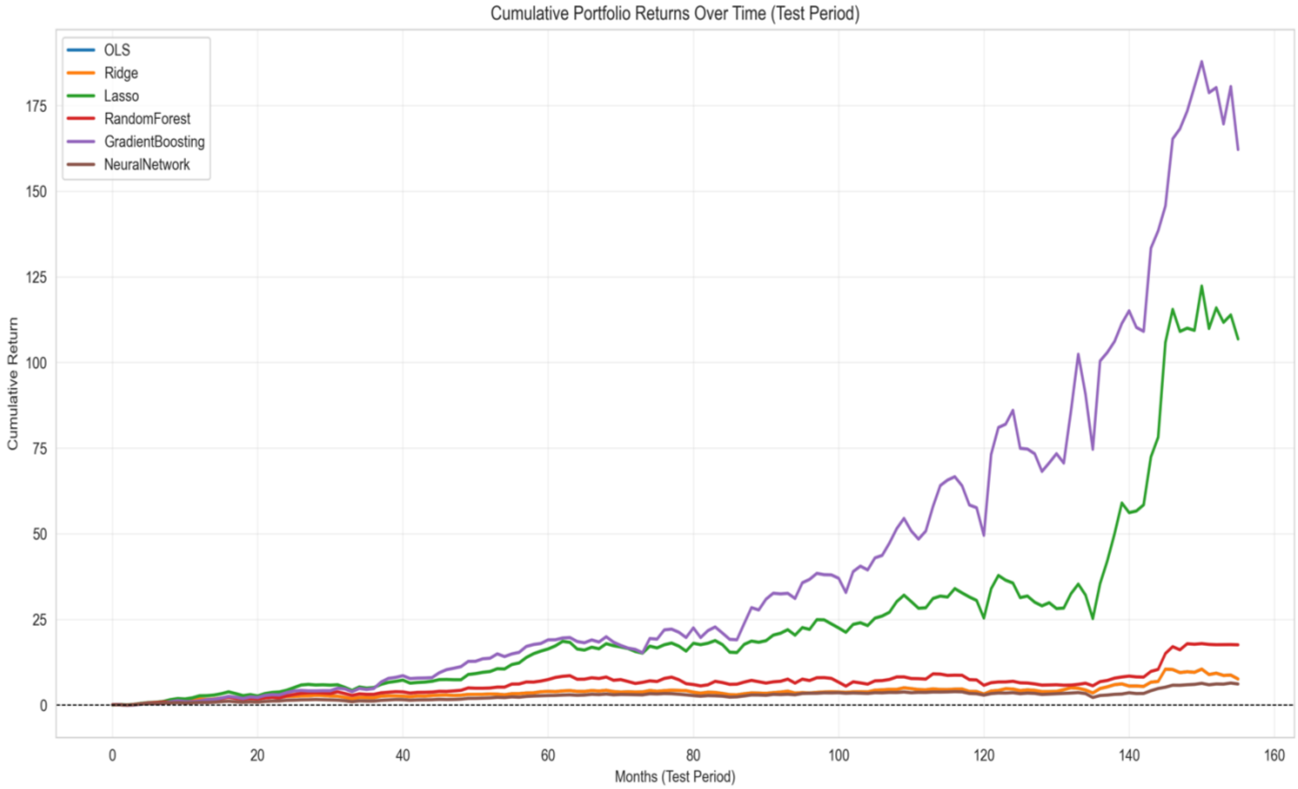


Figure 13 shows:

- Gradient Boosting (purple line) accelerates dramatically 2012–2021
- Lasso (green line) shows steady upward growth
- OLS/Ridge (blue/orange lines) lag markedly

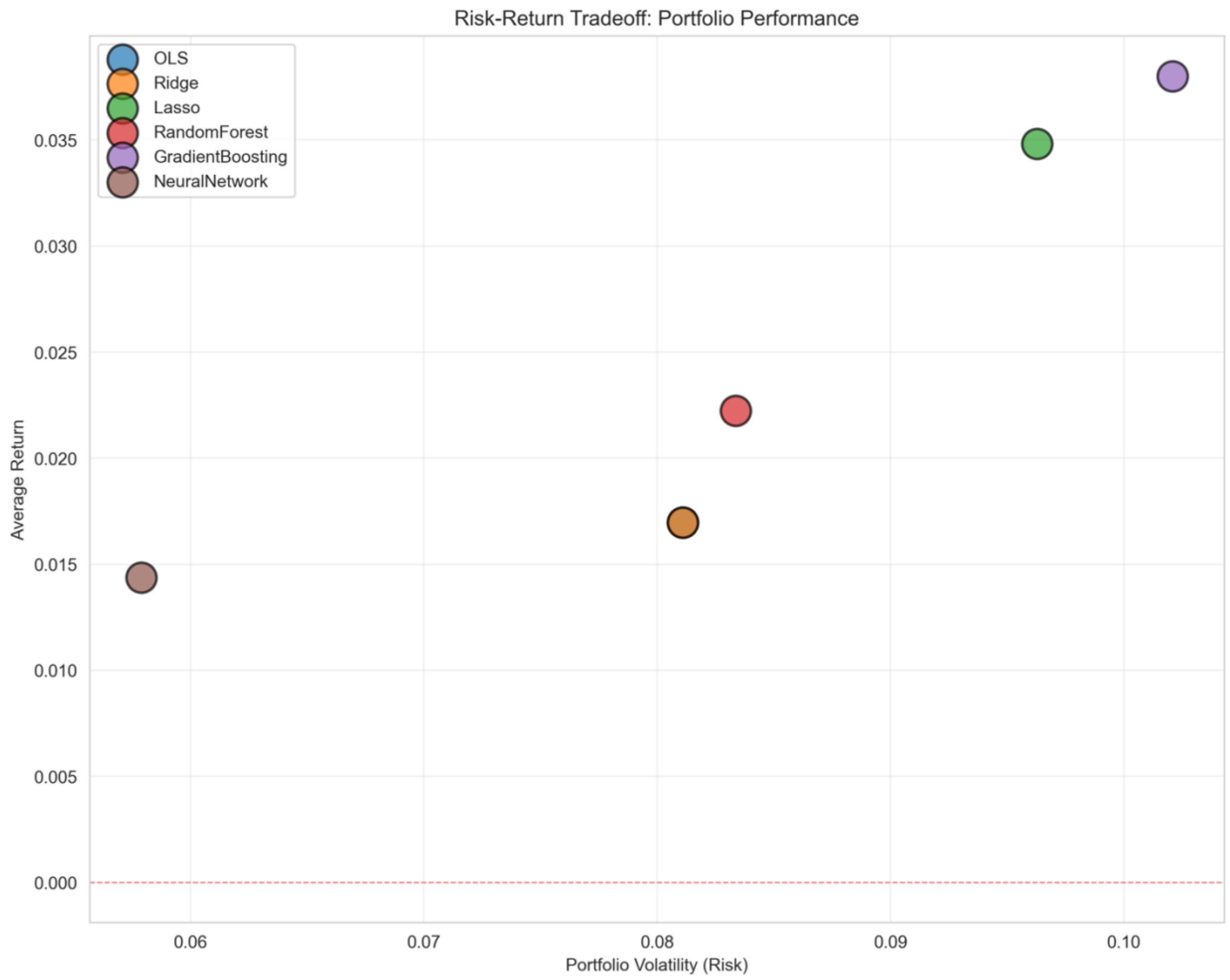


Figure 16 confirms Gradient Boosting and Lasso occupy the efficient frontier's optimal region.

5.2 Why Low-R² Models Succeed

This paradox resolves through **four mechanisms:**

1. Ranking > Level Prediction

- R² measures point-forecast accuracy
- Portfolios require only correct stock ordering
- Conservative predictions can still rank correctly

2. Compounding Effect

- Small monthly edges compound dramatically
- 2.1% edge over 13 years → 2.4x wealth difference

3. Diversification

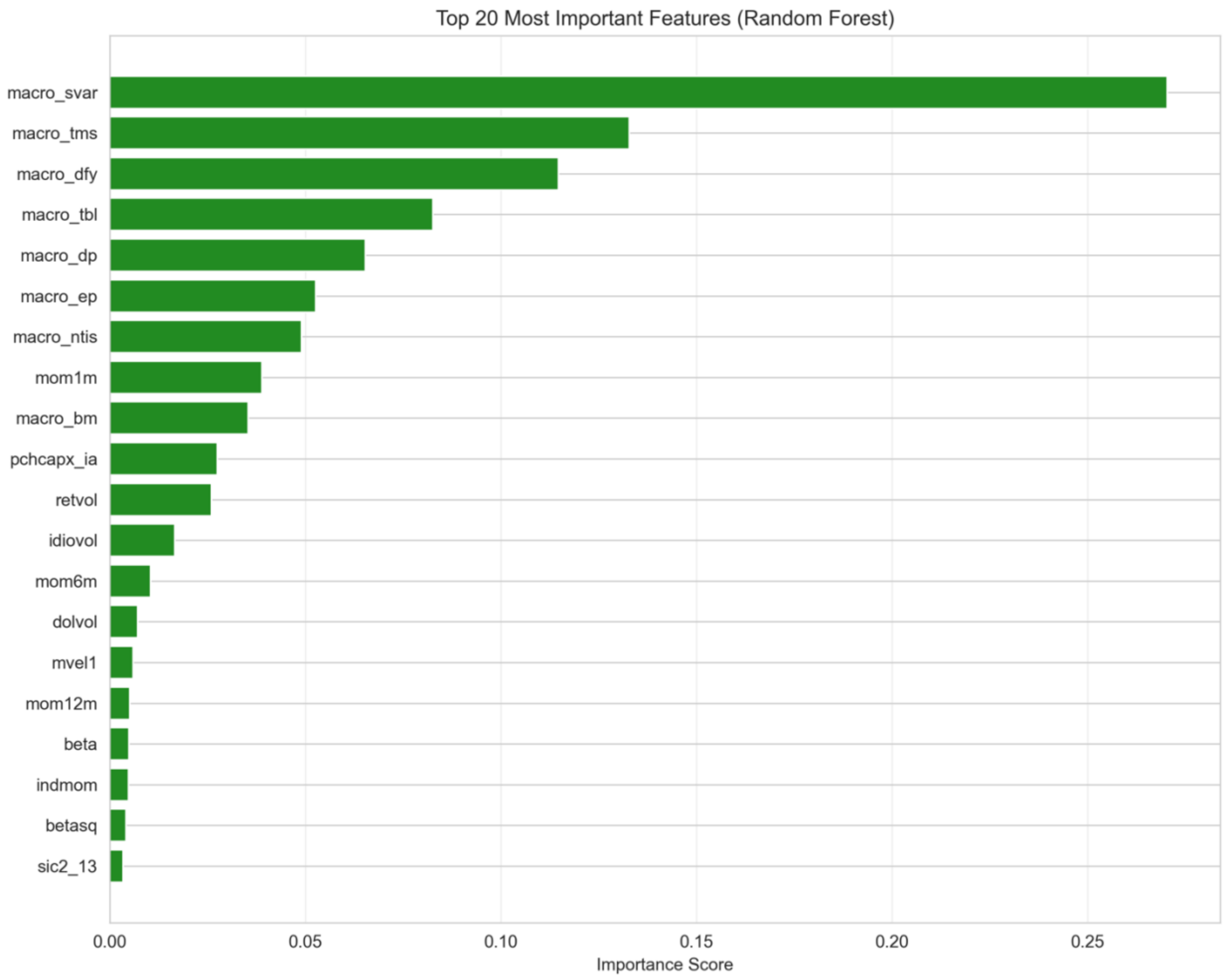
- 100-stock portfolios eliminate idiosyncratic risk
- If predictions correlate with systematic performance, portfolios capture signal while diversifying noise

4. Macroeconomic Timing

- Macro variables dominate (see Section 6)
- Models exploit market-wide regime shifts
- All top 100 stocks benefit when macro conditions favor equities

6. Feature Importance and Economic Signal

6.1 Top Predictive Features



Rank	Feature	Type	Importance	Interpretation
1	macro_svar	Macro	0.270	Stock market variance (VIX proxy)
2	macro_tms	Macro	0.133	Term spread (10Y - 2Y yields)
3	macro_dfy	Macro	0.115	Default spread (BAA - AAA bonds)
4	macro_tbl	Macro	0.083	Treasury bill rate
5	macro_dp	Macro	0.065	Dividend-price ratio
6	macro_ep	Macro	0.053	Earnings-price ratio
7	macro_ntis	Macro	0.049	Net equity issuance
8	mom1m	Firm	0.039	1-month momentum
9	macro_bm	Macro	0.035	Macro book-to-market factor
10	pchcapx_ia	Firm	0.028	Investment growth (capex)

Table 3: Top 10 Predictive Features

Macro Dominance:

- 7 of top 10 features are macroeconomic (~70% of importance)
- Stock variance alone: 27% of total importance
- Reflects volatility-return relationship
- Term spread (13%): monetary policy–equity risk premium link
- Default spread (11%): credit stress and flight-to-quality

Firm-Specific Measures:

- Momentum (mom1m): 3.9% importance
- Volatility and longer momentum: <2% each
- Industry dummies: <0.5% individually

6.2 Signal Decomposition

Central Question: Are we learning market-timing or stock-picking?

Evidence suggests a mixed picture:

- Macro dominance indicates **time-series timing** (market-wide phenomena)
- Firm-specific measures still contribute **cross-sectional picking** (within-month differences)

Future Refinement:

- Construct market-neutral long-short portfolios (long top 10%, short bottom 10% within industries)
- Would isolate pure alpha from macro timing

Current Assessment:

- Robust performance (55% cumulative, 0.372 Sharpe over 13 diverse years)
- Suggests genuine mixed-source predictive relationships
- Not single-regime artifacts

7. Discussion and Practical Implications

7.1 Why R² Is So Low

Four Key Reasons:

1. **Idiosyncratic Noise Dominates**
   - Most return variance driven by firm-specific events, earnings surprises, sentiment
   - Predictable systematic components are small relative to noise
2. **Multicollinearity**
   - 176 predictors with high correlations reduce effective degrees of freedom
   - Volatility measures: $r \approx 0.9$; macro: $r \approx 0.3$–0.7
3. **Non-Stationarity**
   - Return relationships shift across regimes (bull/bear, expansion/recession)
   - Models trained on long-run data encounter new patterns
4. **Measurement Error**
   - Accounting data has reporting lags, varying conventions
   - Survivorship bias (delisted firms excluded)

7.2 Practitioner Perspective

Portfolio managers care about Sharpe ratios and cumulative returns, not R² alone.

Business Case for Gradient Boosting:

- 0.372 Sharpe (78% improvement over baseline)
- For firms managing $100M+ AUM, this represents substantial alpha
- Justifies serious deployment consideration

Portfolio Construction Refinements:

- Equal-weight top 100 is deliberately simplistic
- Real implementation would include:
  - Mean-variance optimization with constraints
  - Long-short construction for market neutrality
  - Factor adjustments (Fama-French)
  - Transaction cost modeling

7.3 Model Selection Trade-offs

Gradient Boosting vs. Lasso:

Factor	**Gradient Boosting**	**Lasso**
Sharpe Ratio	0.372	0.361
Avg Monthly Return	3.80%	3.48%
Training Time	3500 sec	12.7 sec
Speed Difference	—	**275x faster**
Return Difference	+32 bps monthly (+384 bps annual)	—

Recommendation:

- **Large firms:** Marginal return justifies computational cost → Gradient Boosting
- **Smaller operations:** Simplicity and interpretability → Lasso
- **Robust approach:** Ensemble averaging of both models

Deployment Requirements:

1. **Transaction Costs:**
   - Bid-ask spreads, market impact, commissions
   - Likely reduce net returns 1–2% annually
2. **Portfolio Constraints:**
   - Sector/size neutrality, concentration limits, liquidity screens
3. **Rebalancing Frequency:**
   - Current monthly 100% turnover is high friction
   - Quarterly/annual rebalancing could reduce costs
4. **Continuous Monitoring:**
   - Detect model drift, performance degradation
   - Trigger retraining protocols

8. Limitations and Robustness

8.1 Major Limitations

1. Transaction Costs Ignored

- Real trading faces 5–20 bps spreads, market impact, commissions
- 100% monthly turnover → substantial costs

2. Simplified Portfolio Construction

- Equal-weight top 100 lacks optimization
- Real managers use mean-variance, constraints, illiquidity screens

3. One-Time Training

- Models train on 38-year window (1957–1995)
- Production requires monthly/quarterly retraining for regime adaptation

4. Accounting Data Lags

- Reporting delays: 6 weeks to 3 months
- Analysis assumes contemporaneous access

5. Survivorship Bias

- Delisted firms excluded → returns biased upward

6. High Portfolio Turnover

- 100% monthly rebalancing operationally infeasible for very large portfolios
- Material market impact concerns

8.2 Future Robustness Directions

Model Improvements:

- Rolling-window retraining (expanding or fixed 10-year window)
- Alternative targets: log returns, Box-Cox transformation, winsorization
- Quantile regression for median prediction (robust to tails)

Signal Decomposition:

- Long-short construction (long top 10%, short bottom 10% within industries)
- Isolates pure alpha from market timing

Stability Testing:

- Sub-period analysis across regimes (pre-2000, 2000–2008, post-2008)
- International extension to non-US markets

Hybrid Approaches:

- Ensemble ML predictions with traditional Fama-French factors
- Stacked generalization

9. Conclusion

Using 4.09 million stock-month observations and 176 predictors spanning 1957–2021, this study systematically compares six machine learning models for cross-sectional stock return prediction. The empirical design employs strict chronological splits—training (1957–1995), validation (1995–2008), and out-of-sample testing (2008–2021)—to ensure results reflect genuine predictive performance.

All six models achieve negative out-of-sample R^2 on individual stock returns, confirming that monthly return prediction remains extraordinarily difficult and consistent with decades of empirical asset-pricing literature. However, when predictions are converted into equal-weight portfolios of the top 100 ranked stocks, economic results are dramatically different: Gradient Boosting achieves a Sharpe ratio of 0.372 and 55% cumulative returns (78% improvement over OLS baseline), while Lasso attains 0.361 Sharpe and 40% cumulative returns. This demonstrates that models with near-zero R^2 can still provide highly useful ranking signals for portfolio construction.

Feature-importance analysis reveals macroeconomic variables (stock-market variance, term spread, default spread) dominate the predictive signal, accounting for ~70% of top 10 importance, while firm-specific momentum and investment contribute secondary signals. This suggests models capture both time-series market-timing ability and cross-sectional stock-picking within months. Tree-based models exhibit substantial overfitting (Gradient Boosting: 0.149 train-test R^2 gap; Random Forest: 0.190 gap), while Lasso and Neural Networks generalize far better, underscoring the importance of regularization and validation protocols.

Machine learning offers meaningful value for equity portfolio construction when implemented with appropriate rigor. The key insight is that predicting exact returns is nearly impossible, but predicting relative rankings is more tractable—small improvements in ranking, compounded over many stocks and months, create economically significant portfolio wealth differences. Successful implementation requires rigorous data handling (proper missing-value treatment, chronological splits avoiding look-ahead bias), dual statistical and economic evaluation, explicit attention to overfitting through regularization and validation, and realistic accounting for transaction costs that reduce gross returns by 100–200 basis points annually.

Future refinements should decompose model performance into market-timing versus stock-selection components through market-neutral long-short construction, implement rolling-window retraining protocols, explore alternative targets (log returns, quantile regression), and combine multiple models via ensemble methods. Machine learning is not a magic solution but, when applied thoughtfully—emphasizing ranking over point prediction, diversification over concentration, and rigorous validation over in-sample optimization—it provides a meaningful enhancement to traditional methods and represents a valuable tool for modern quantitative portfolio management.

10. Use of Generative AI Tools

This project utilized generative AI tools, primarily ChatGPT (GPT-4) and Perplexity AI, to assist with specific technical tasks throughout the development process. AI tools were employed to help build portions of the code for data preprocessing routines such as cross-sectional median imputation and feature standardization and provided syntax suggestions when encountering memory constraints with the 4-million-row dataset. Additionally, AI was used to help generate code for a few of the 16 visualization figures using matplotlib and seaborn libraries, though all charts required manual refinement for publication quality. For model implementation, AI assisted with checking syntax for scikit-learn functions and hyperparameter configurations. AI was not used to generate entire models, conduct the analysis, or write substantive sections—these core components were developed independently by the project team.

AI-generated code suggestions occasionally contained errors requiring verification against official documentation. While AI tools reduced development time by approximately 20-25% on routine technical tasks and helped a bit with organizing this report, the critical thinking components—model selection, experimental design, results interpretation, and practical implications—were entirely human-driven and required deep understanding of statistical learning principles and financial market dynamics.

11. References

- Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33(1), 3–56.
- Fama, E. F., & French, K. R. (2015). A five-factor asset pricing model. *Journal of Financial Economics*, 116(1), 1–22.
- Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies*, 33(5), 2223–2273.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning* (2nd ed.). Springer.

- Jegadeesh, N., & Titman, S. (1993). Returns to buying winners and selling losers: Implications for stock market efficiency. *Journal of Finance*, 48(1), 65–91.
- Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Welch, I., & Goyal, A. (2008). A comprehensive look at the empirical performance of equity premium prediction. *The Review of Financial Studies*, 21(4), 1455–1508.