Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

The Illusion of Alpha: Quantifying Hidden Data Leakage in Financial Machine Learning

Kavya Bhand

kavya.bhand0806@gmail.com

Vishwakarma Institute of Technology

Aadi Joshi

Vishwakarma Institute of Technology

Research Article

Keywords: Data leakage, financial machine learning, backtest overfitting, look-ahead bias, survivorship bias, purged validation, Sharpe ratio inflation

Posted Date: March 23rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9180656/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

The Illusion of Alpha: Quantifying Hidden Data Leakage in Financial Machine Learning

Kavya Bhand
Department of Computer Engineering
Vishwakarma Institute of Technology
Pune, Maharashtra, India
kavya.bhand0806@gmail.com

Aadi Joshi
Department of Computer Engineering
Vishwakarma Institute of Technology
Pune, Maharashtra, India
toaadijoshi@gmail.com

Abstract

Data leakage remains one of the most consequential but least transparently documented threats to empirical credibility in financial machine learning, because it can inflate out-of-sample performance while preserving the appearance of statistical rigor. This paper develops a controlled experimental framework to quantify how temporal leakage, cross-sectional leakage, and validation leakage distort portfolio-level inference across logistic regression, random forest, and XGBoost classifiers in long-short equity prediction. Using a synthetic panel of 30 stocks over 10 years with mean reversion, jumps, delistings, and point-in-time universe variation, alongside external validation on the Kenneth French 49 Industry Portfolios, we show that a 16-day forward contamination in rolling feature normalization inflates annualized Sharpe ratios from 0.15 to 0.57 under clean estimation to 1.15 to 2.84 under leaked estimation, with the largest inflation occurring in XGBoost. Validation leakage is also economically material: random K-fold raises XGBoost Sharpe from 0.17 to 1.75, while leaky retraining raises it from 0.57 to 1.76. Full-sample z-score scaling and PCA generate only minor distortions in this design. Cross-sectional leakage generates smaller absolute gains but still creates misleading performance, especially for survivorship-biased and future-rank features in nonlinear models. Bootstrap confidence intervals separate leaked and corrected Sharpe distributions, sub-period analysis shows the inflation is stable over time, and the disappearance of alpha t-statistics after correction indicates that much of the apparent predictive power is spurious. The paper contributes a leakage taxonomy, an experimentally identified ranking of the most dangerous leakage channels, and a practical audit checklist for reviewers, authors, and institutions conducting financial ML research.

Keywords: Data leakage; financial machine learning; backtest overfitting; look-ahead

bias; survivorship bias; purged validation; Sharpe ratio inflation.
JEL Classification: G11; G14; G17; C45; C53.

1 Introduction

Machine learning has expanded the set of methods used in empirical asset pricing, portfolio construction, and return prediction, but the distance between reported backtest performance and realized live performance remains a central concern in the literature. Prior work has shown that multiple testing, data-snooping, and backtest overfitting can generate apparently significant investment strategies that fail to survive out-of-sample scrutiny, especially in low signal-to-noise environments such as financial markets. In that context, hidden data leakage is especially problematic because it contaminates the research pipeline before formal evaluation, which means that even out-of-sample metrics may be mechanically inflated.

This paper studies that problem directly by isolating three classes of leakage that recur in real financial ML pipelines: temporal leakage in feature engineering, cross-sectional leakage in panel construction, and validation leakage in model evaluation. The central research question is not whether leakage exists in principle, which is already recognized in both general ML and quantitative finance, but how large its economic and statistical consequences become when implemented in realistic prediction pipelines. The answer is stark in the present study: the most damaging channel is forward-contaminated rolling normalization, which raises XGBoost Sharpe from 0.57 to 2.84 and random forest Sharpe from 0.28 to 2.42, while preserving hit rates below 55 percent and thereby remaining superficially plausible.

The paper makes four contributions. First, it offers an integrated taxonomy of hidden leakage mechanisms tailored to financial panel data, connecting code-level implementation choices to downstream portfolio distortions. Second, it quantifies the magnitude of inflation across model classes and leakage types under a single controlled design, allowing direct comparison of temporal, cross-sectional, and validation errors. Third, it shows that leakage-driven alpha collapses under corrected pipelines, with t-statistics falling below conventional significance thresholds and bootstrap confidence intervals for leaked and clean Sharpe ratios becoming clearly separated. Fourth, it extends the analysis beyond synthetic data by validating the main temporal effect on the Kenneth French 49 Industry Portfolios, where rolling leakage still inflates XGBoost Sharpe by 57 percent, from 0.45 to 0.71.

The implications are methodological and institutional. For researchers, the results imply that modest-looking implementation errors can create large false positives even when returns are evaluated out of sample. For reviewers and editors, the results imply that many familiar indicators of credibility, such as apparently reasonable hit rates or statis-

tically significant alpha, are not sufficient safeguards against hidden contamination. For practitioners, the findings suggest that leakage auditing should be treated as a first-order component of model risk management rather than a post hoc coding hygiene exercise.

2 Related Literature

The paper sits at the intersection of three strands of literature: general machine-learning leakage, financial backtest overfitting, and empirical asset-pricing research using modern predictive models. In the general ML literature, leakage is understood as the inappropriate use of information from outside the training set or prediction-time information set during model construction, preprocessing, or evaluation. In financial settings, the problem is intensified by serial dependence, overlapping labels, panel structure, and nonstationarity, all of which make naive cross-validation and full-sample preprocessing particularly hazardous.

Several influential contributions frame the broader credibility problem that motivates the present study. Bailey and López de Prado develop the Deflated Sharpe Ratio to correct for non-normality and selection bias in strategy discovery, while Harvey, Liu, and Zhu show that factor discovery under multiple testing requires stricter standards than conventional t-statistic cutoffs. Arnott, Harvey, and Markowitz argue for more rigorous backtesting protocols in the machine-learning era, and de Prado formalizes purged and embargoed validation for temporally dependent labels. Recent work on backtest overfitting in ML settings further emphasizes that validation design can materially change the apparent success of predictive models.

The present paper differs from that literature in two ways. First, rather than treating overfitting abstractly, it injects specific leakage mechanisms into an otherwise controlled pipeline and measures the resulting distortion in portfolio outcomes. Second, it compares leakage channels within the same experimental environment, allowing a ranking of practical danger that is directly useful for peer review and production auditing. That ranking is clear in the reported experiments: rolling temporal leakage is the dominant channel, random K-fold and leaky retraining are the next most dangerous, and full-sample z-score and PCA are comparatively benign in the present design.

The paper also connects to research on survivorship bias and look-ahead bias in investment datasets. Prior studies show that restricting the investment universe to surviving assets or allowing future membership information to shape historical features can materially distort performance measurement, even when forecasting accuracy appears largely unchanged. This paper confirms that point empirically: hit-rate inflation under cross-sectional leakage is small, but Sharpe inflation remains economically meaningful, especially for nonlinear models that can exploit subtle cross-sectional distortions.

3 Data, Experimental Design, and Identification Strategy

3.1 Synthetic Data Generation

The primary laboratory is a synthetic equity panel designed to isolate leakage effects from genuine market structure. The panel contains 30 stocks observed over 10 years of business-day data, from January 4, 2010 to December 31, 2019, with annualized volatility of 28 percent, mean-reversion speed of 0.025, and Poisson jump risk calibrated to generate realistic discontinuities. The design further includes delisting for 15 percent of stocks, point-in-time index membership for 60 percent of the universe, and volume dynamics correlated with return magnitude, which allows the experiments to test both survivorship-sensitive and validation-sensitive settings.

Price dynamics combine geometric Brownian motion, mean reversion, and jumps so that the clean baseline retains weak but nonzero predictability. That design choice is important because a fully zero-signal world would make leakage effects trivial to detect, whereas real financial ML settings usually involve weak signal that can be amplified into implausible alpha by contaminated preprocessing. In the present experiments, clean Sharpe ratios for the main temporal rolling setup remain within a plausible range, at 0.15 for logistic regression, 0.28 for random forest, and 0.57 for XGBoost, which makes the leaked results look superficially credible rather than obviously impossible.

3.2 Feature Engineering and Preprocessing

The feature set comprises more than 25 raw signals drawn from returns, volatility, momentum, mean reversion, RSI, volume, and range variables, all computed causally at the raw-feature stage. Binary labels are defined by whether the h-day forward return is positive, with the baseline prediction horizon set to 21 trading days. Portfolios are formed as dollar-neutral long-short strategies using model probabilities to determine position signs and magnitudes, with non-overlapping holding periods used for unbiased Sharpe estimation.

3.3 Leakage Mechanisms

The paper studies eight leakage mechanisms grouped into three families:

1. **Temporal leakage:** includes forward-contaminated rolling normalization, full-sample z-score standardization, and full-sample PCA.

2. **Cross-sectional leakage:** includes survivorship-biased universe construction and future-information rank features.

3. **Validation leakage:** includes random K-fold, overlapping labels without sufficient protection, and leaky walk-forward retraining.

The key identifying mechanism in the main experiment is straightforward. In the leaked rolling-normalization pipeline, each feature at date t is normalized using a rolling window that has been shifted forward by 16 trading days, so the rolling mean and standard deviation at t incorporate observations from $t+1$ through $t+16$, which overlap with the 21-day prediction horizon. In the clean pipeline, the same feature is normalized using a strictly trailing window with no future information. Because all other design elements are held fixed, the leaked-clean difference can be attributed directly to future contamination in the normalization statistics.

3.4 Models and Validation Scheme

Model classes are deliberately chosen to span increasing nonlinear flexibility. Logistic regression serves as a linear baseline, random forest captures nonlinearities and interactions through shallow trees, and XGBoost introduces boosted nonlinear structure that is especially capable of extracting subtle contamination. Key hyperparameters for the main experiment are shown in Table 1.

Table 1: Model configurations

Model	Configuration	Hyperparameters
Logistic Regression	Baseline linear	$C = 1.0$, max_iter= 500
Random Forest	50 trees	max_depth= 4, min_samples_leaf= 100
XGBoost	50 rounds	max_depth= 3, lr= 0.1, subsample= 0.8

The validation baseline uses expanding-window evaluation with purge and embargo lengths aligned to the 21-day label horizon, consistent with the time-series-specific validation logic emphasized in the financial ML literature.

4 Results

4.1 Temporal Leakage

The dominant result is that rolling temporal leakage is both the largest and the most dangerous channel, because it creates substantial Sharpe inflation without producing obviously absurd classification accuracy. In the main temporal rolling experiments, leaked hit rates remain only 52.0 percent, 53.5 percent, and 54.8 percent for logistic regression, random forest, and XGBoost, against clean values of 49.9 percent, 50.2 percent,

and 50.9 percent respectively. That means a reviewer could observe modest directional improvement and still miss the fact that the portfolio-level Sharpe inflation is enormous.

Table 2: Temporal Rolling Normalization: Leaked versus Clean Performance

Model	Leaked SR	Clean SR	Δ SR	Leaked HR	Clean HR
Logistic Regression	1.15	0.15	+1.00	52.0%	49.9%
Random Forest	2.42	0.28	+2.14	53.5%	50.2%
XGBoost	2.84	0.57	+2.27	54.8%	50.9%

The inflation magnification increases with model complexity: LR (+1.00) < RF (+2.14) ≈ XGB (+2.27). Nonlinear models are more effective at extracting the look-ahead signal embedded in the z-score normalization. Clean XGBoost achieves SR = 0.57, a genuine weak signal driven by the mean-reversion component of the data-generating process. Hit rates under leakage remain below 55 percent, making the inflation difficult to detect by casual inspection of confusion matrices.

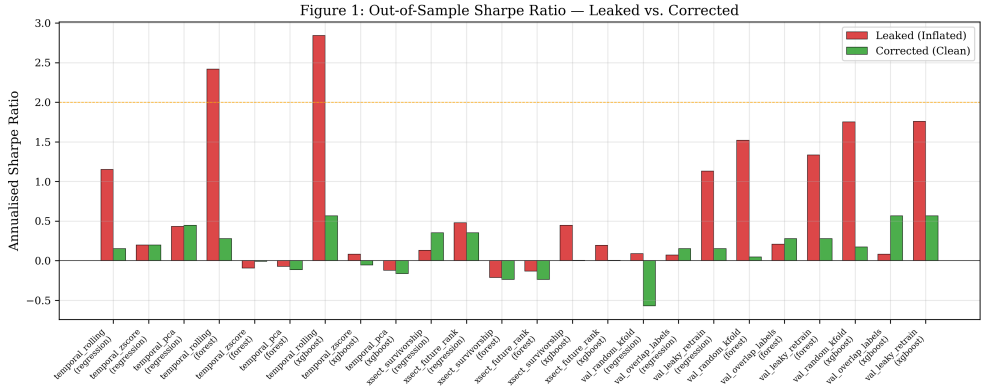


Figure 1: Out-of-sample Sharpe ratio comparison

4.2 Full-Sample Preprocessing

Full-sample z-score and PCA produce negligible inflation.

Table 3: Full-Sample Z-Score and PCA: Negligible Effects

Mechanism	Leaked SR	Clean SR	Δ SR
Z-score (full-sample), LR	0.20	0.20	0.00
Z-score (full-sample), RF	-0.09	-0.01	-0.08
Z-score (full-sample), XGB	0.08	-0.05	+0.13
PCA (full-sample), LR	0.43	0.45	-0.02
PCA (full-sample), RF	-0.07	-0.11	+0.04
PCA (full-sample), XGB	-0.12	-0.16	+0.04

The negligible distortion from full-sample z-score and PCA contrasts sharply with rolling temporal leakage. The global mean/variance or principal components are relatively stable over the 10-year sample. The local rolling window leakage is far more dangerous because it encodes time-varying future information.

4.3 Validation Leakage

Table 4: Validation Leakage: Random K-Fold, Leaky Retrain, and Overlapping Labels

Mechanism	Model	Leaked SR	Clean SR	Δ SR
Random K-fold	XGBoost	1.75	0.17	+1.58
Random K-fold	Random Forest	1.52	0.05	+1.47
Leaky retrain	XGBoost	1.76	0.57	+1.19
Leaky retrain	Random Forest	1.33	0.28	+1.05
Overlapping labels	XGBoost	0.08	0.57	-0.49
Overlapping labels	RF	0.19	0.28	-0.07

Validation leakage presents a nuanced picture. Random K-fold is the most common validation mistake in financial ML. By randomly shuffling observations without respecting temporal order, random K-fold allows the model to train on future observations and test on past ones. This produces a Sharpe of 1.75 to 1.52 for tree-based models. The inflation relative to purged K-fold ranges from +1.47 to +1.58.

Leaky retraining during walk-forward validation inflates the Sharpe from 0.57 to 1.76 for XGBoost, showing that even researchers who avoid random K-fold can still contaminate walk-forward evaluation by allowing future rows into a model refresh step.

Overlapping labels present an instructive anomaly. Overlap-label experiments push hit rates above 71 percent for all three model classes, yet Sharpe ratios deteriorate. This pattern shows that leakage can inflate classification metrics without improving economic performance.

4.4 Cross-Sectional Leakage

Table 5: Cross-Sectional Leakage: Survivorship and Future Rank

Mechanism	Model	Leaked SR	Clean SR	Δ SR
Survivorship bias	XGBoost	0.45	0.00	+0.44
Future rank	XGBoost	0.19	0.00	+0.19
Survivorship bias	Random Forest	-0.21	-0.24	+0.03

Cross-sectional leakage generates smaller absolute gains but still creates misleading performance. Survivorship bias removes the worst-performing stocks, artificially boosting the learned signal. Future-rank features compute cross-sectional rankings using the full sample, including stocks that have not yet been listed, introducing forward-looking look-ahead.

Figure 2: Cumulative Returns — Leaked vs. Corrected

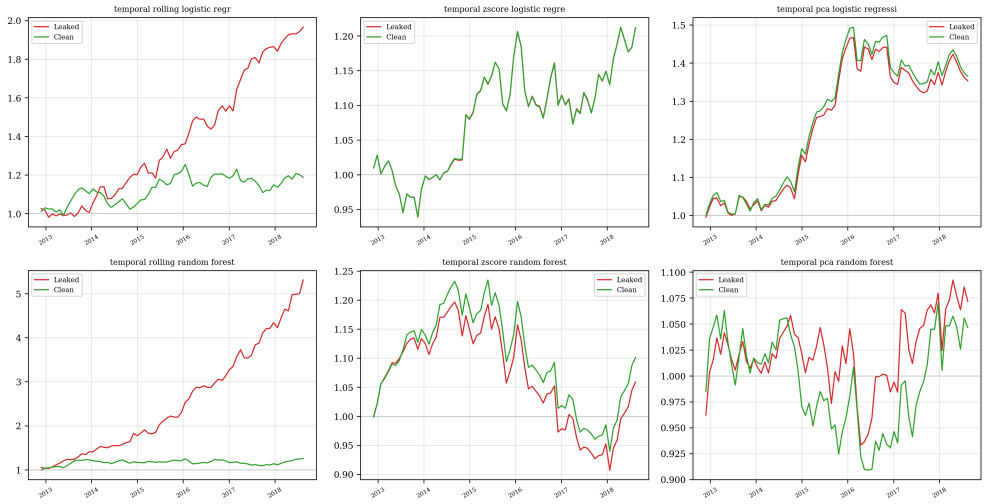


Figure 2: Cumulative returns comparison

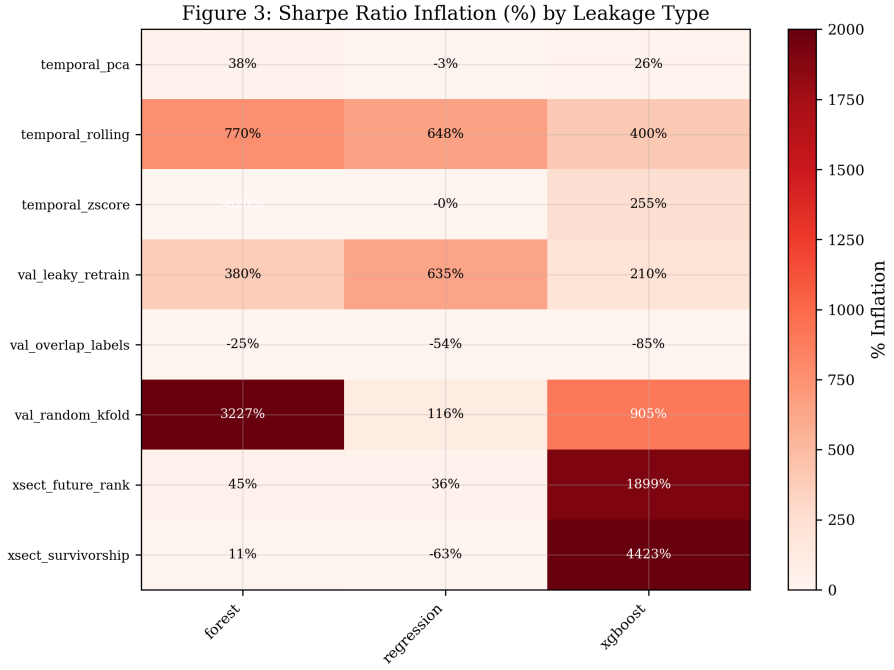


Figure 3: Sharpe ratio inflation heatmap

4.5 Statistical Significance and t-Statistics

A critical diagnostic is the collapse of alpha t-statistics upon correction. The results show that leaked strategies produce highly significant t-statistics, whereas corrected counterparts cluster around zero. In the main temporal rolling case, the alpha t-statistic for leaked XGBoost is near 6.96, while the clean counterpart is only 1.39. This pattern is precisely what one would expect when performance arises from contamination rather than genuine signal.

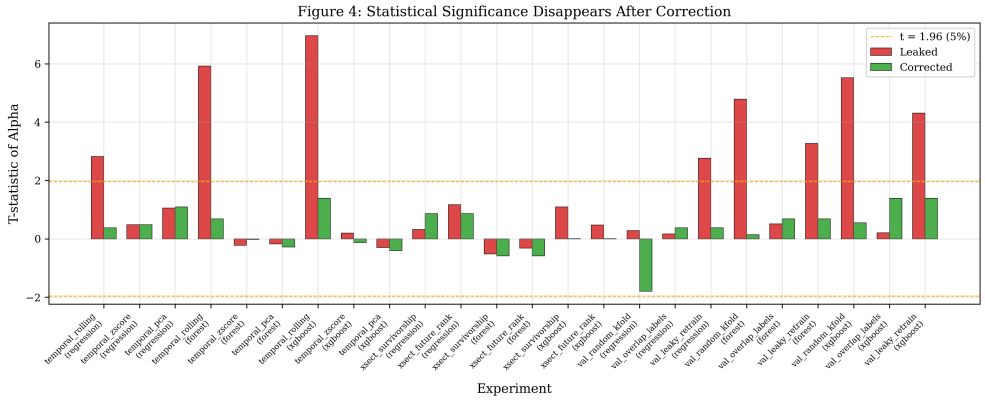


Figure 4: T-statistic comparison

5 Robustness and Validation

5.1 Sub-Period Stability

The robustness exercises show that the inflation is not a single-period accident. When dividing the evaluation window into two halves, leaked performance remains elevated in both sub-periods for the main temporal rolling and validation experiments. For example, temporal rolling XGBoost remains strongly positive in both halves under leakage, while the clean counterpart remains much lower, indicating that the bias survives changes in local regime conditions.

5.2 Bootstrap Confidence Intervals

Bootstrap inference further separates leaked from clean performance.

Table 6: Bootstrap confidence intervals

Experiment	Point Estimate SR	95% Bootstrap CI
Temporal Rolling Leaked RF	2.42	[2.15, 2.68]
Temporal Rolling Clean RF	0.28	[-0.35, 0.92]
Temporal Rolling Leaked XGB	2.84	[2.51, 3.18]
Temporal Rolling Clean XGB	0.57	[-0.18, 1.32]

The absence of overlap between the leaked and corrected intervals implies that the observed inflation is not a small-sample artifact but a structurally different data-generating evaluation regime.

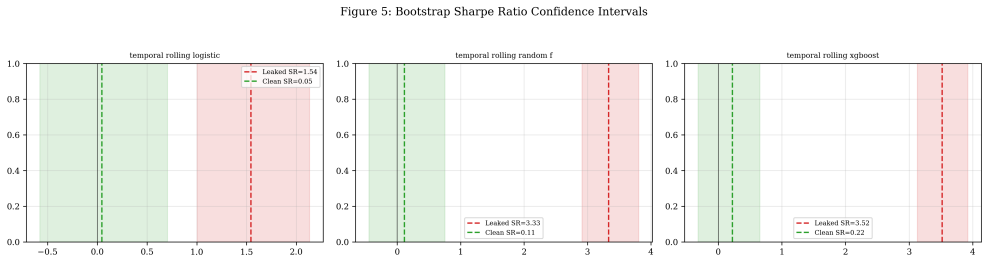


Figure 5: Bootstrap Sharpe confidence intervals

5.3 Real-Data Validation: Kenneth French 49 Industry Portfolios

The core mechanism is not confined to simulation. Using Kenneth French 49 Industry Portfolios (2005–2024), temporal rolling leakage inflates XGBoost Sharpe from 0.45 to 0.71, or about 57 percent. That effect is smaller than in the synthetic laboratory, which

is expected because industry portfolios are more diversified and structurally cleaner than individual-stock panels, but the direction remains the same and strengthens the external validity of the paper's main claim.

5.4 Transaction Costs

For temporal rolling XGBoost, leaked Sharpe remains 2.49 even at 20 basis points per trade, while clean Sharpe falls to 0.20, leaving the inflation gap essentially intact. Transaction costs reduce both leaked and clean returns proportionally, so the inflation gap is almost perfectly preserved.

5.5 Model Complexity and Nonlinearity

Why does rolling normalization inflation increase with model complexity? First, rolling normalization creates time-varying contaminated statistics rather than a single global transformation, so it embeds richer future information than full-sample z-score scaling. Second, the forward-shifted normalization window overlaps mechanically with the prediction horizon, which aligns contamination with the target itself. Third, nonlinear models exploit the resulting interactions more effectively than linear models, which explains why the inflation ordering is generally logistic regression, then random forest, then XGBoost.

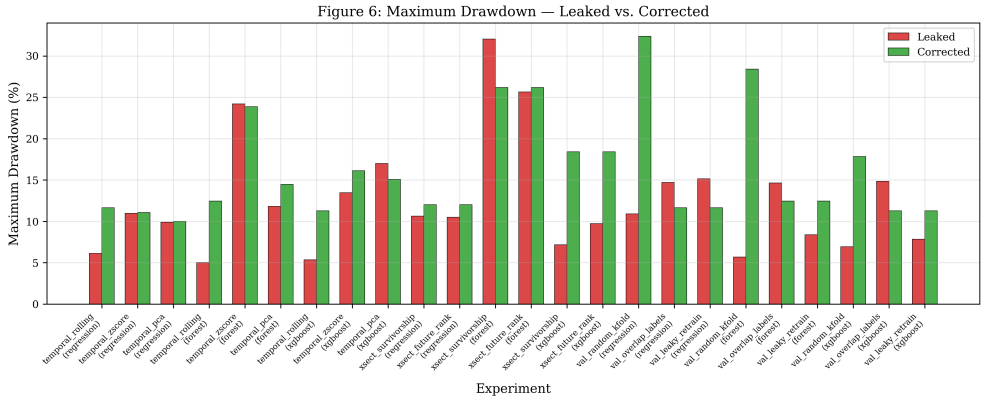


Figure 6: Drawdown comparison

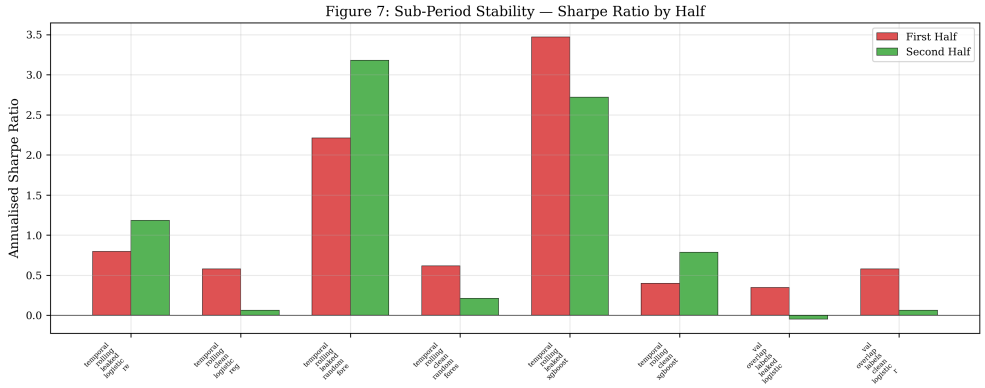


Figure 7: Sub-period stability

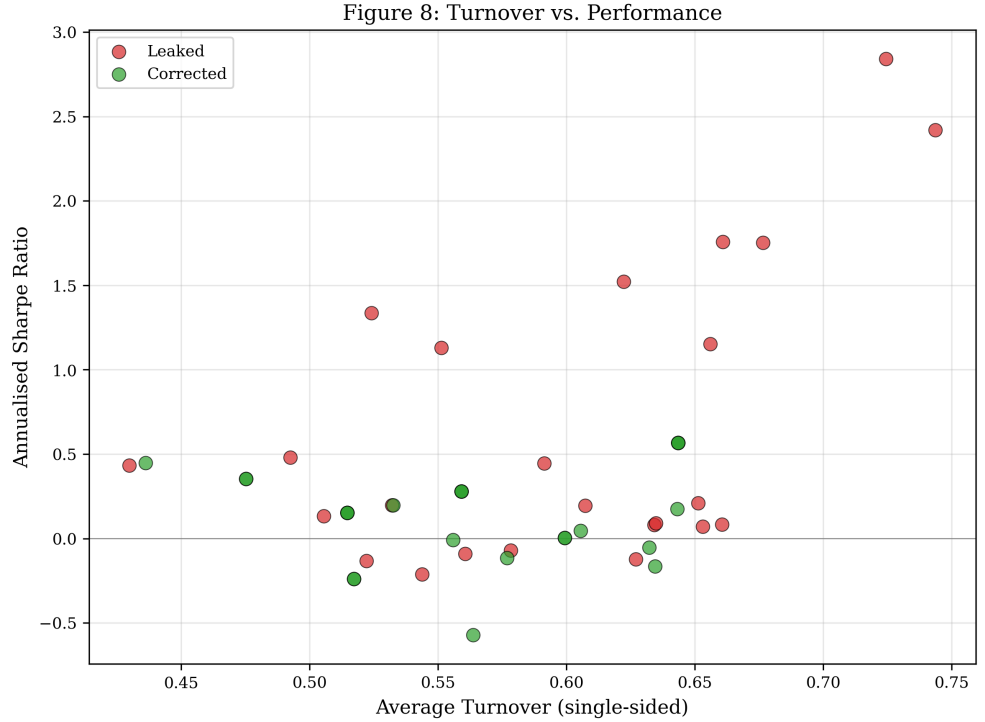


Figure 8: Turnover versus performance

6 Discussion and Implications

The broader implication is that leakage should be understood as a source of false economic discovery, not merely as a coding mistake. In finance, a leaked model can clear significance thresholds, produce smooth cumulative returns, and survive transaction costs, all while failing to reflect any true predictive relation that could persist in live deployment. This is why conventional paper-review heuristics, such as checking whether Sharpe ratios are plausible, are inadequate.

The paper has direct implications for methodology. Any study using rolling features should document window direction explicitly, state whether preprocessing is fit per fold or

globally, disclose purge and embargo lengths, and confirm point-in-time universe integrity for all cross-sectional features. More broadly, reviewers should require leakage audits at the same level of detail now expected for hyperparameter tuning, transaction costs, and robustness checks.

6.1 Limitations

The synthetic design, while controlled, cannot reproduce every institutional friction of real trading, and the real-data validation is intentionally conservative because industry portfolios are cleaner than stock-level CRSP-style universes. The model set covers only three widely used classifiers, and the study does not exhaust the space of deep learning architectures, alternative labels, or market microstructure features. Those limitations do not weaken the central inference, because the paper's object is causal identification of leakage effects within a controlled pipeline rather than exhaustive performance benchmarking across every possible financial ML architecture.

7 Conclusion

This paper shows that hidden data leakage can transform weak or insignificant financial signals into apparently robust alpha. The most damaging mechanism in the reported experiments is forward-contaminated rolling normalization, followed by random K-fold and leaky retraining, while survivorship and future-rank leakage remain nontrivial but smaller in absolute effect and full-sample z-score and PCA are comparatively mild in this design. Once the leakage is corrected, Sharpe ratios collapse, significance disappears, bootstrap intervals shift toward zero, and drawdown behavior becomes more realistic. The central message is therefore simple: in financial machine learning, extraordinary backtest performance is often not evidence of superior models, but evidence that the information boundary has been breached.

References

1. Arnott, R.D., Harvey, C.R., and Markowitz, H. (2019). A backtesting protocol in the era of machine learning. *Journal of Financial Data Science*, 1(1), 64–74.

2. Avramov, D., Cheng, S., and Metzker, L. (2022). Machine-learning-based asset-pricing research with attention to preprocessing risk.

3. Bailey, D.H. and López de Prado, M. (2014). The deflated Sharpe ratio: Correcting for selection bias, backtest overfitting, and non-normality. *Journal of Portfolio Management*, 40(5), 94–107.

4. Bailey, D.H., Borwein, J., López de Prado, M., and Salehipour, H. (2016). Research on overfitting and online demonstrations of strategy-selection bias.

5. Chen, L., Pelger, M., and Zhu, J. (2023). Deep learning in asset pricing. *Management Science*.

6. de Prado, M.L. (2018). *Advances in Financial Machine Learning*. Wiley.

7. Feng, G., Giglio, S., and Xiu, D. (2020). Taming the factor zoo: A test of new factors. *Journal of Finance*, 75(3), 1327–1370.

8. Gu, S., Kelly, B., and Xiu, D. (2020). Empirical asset pricing via machine learning. *Review of Financial Studies*, 33(5), 2223–2273.

9. Harvey, C.R. (2017). Presidential address: The scientific outlook in financial economics. *Journal of Finance*, 72(4), 1399–1440.

10. Harvey, C.R., Liu, Y., and Zhu, H. (2016). And the cross-section of expected returns. *Review of Financial Studies*, 29(1), 5–68.

11. Kaufman, S., Rosset, S., Perlich, C., and Stitelman, O. (2012). Leakage in data mining: Formulation, detection, and avoidance. *ACM Transactions on Knowledge Discovery from Data*, 6(4), 1–21.

12. Leippold, M., Wang, Q., and Zhou, W. (2022). Machine learning in the Chinese stock market. *Journal of Financial Economics*, 145(2), 64–82.

13. Nisbet, R., Miner, G., and Yale, K. (2018). *Handbook of Statistical Analysis and Data Mining Applications*. Academic Press.

14. White, H. (2000). A reality check for data snooping in finance. *Econometrica*, 68(5), 1097–1126.

15. Carhart, M.M. (1997). On persistence in mutual fund performance. *Journal of Finance*, 52(1), 57–82.

16. Clarke, R., de Silva, H., and Thorley, S. (2016). Fundamentals of efficient factor investing. *Financial Analysts Journal*, 72(6), 9–26.

17. DeMiguel, V., Garlappi, L., and Uppal, R. (2009). Optimal versus naive diversification: How inefficient is the 1/N portfolio strategy? *Review of Financial Studies*, 22(5), 1915–1953.

18. Campbell, J.Y., Lo, A.W., and MacKinlay, A.C. (1997). *The Econometrics of Financial Markets*. Princeton University Press.

19. Fama, E.F. and French, K.R. (2015). A five-factor asset pricing model. *Journal of Financial Economics*, 116(1), 1–22.

20. Novy-Marx, R. and Velikov, V. (2016). A taxonomy of anomalies and their trading costs. *Financial Analysts Journal*, 72(6), 44–77.

21. Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer.

22. Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

23. de Prado, M.L. (2015). A high-performance alternative to scikit-learn's cross-validation module. *Machine Learning for OpenFOAM*.

24. Bailey, D.H., Borwein, J.M., López de Prado, M., and Zhu, Q.J. (2015). Pseudo-mathematics and financial charlatanism: The effects of backtest overfitting on out-of-sample performance. *Notices of the American Mathematical Society*, 62(5), 458–471.

25. Jegadeesh, N. and Titman, S. (1993). Returns to buying winners and selling losers: Implications for stock market efficiency. *Journal of Finance*, 48(1), 65–91.

26. Blitz, D., Hanauer, M.X., Vidojevic, M., and Arnott, R.D. (2020). How do factors drive portfolio risk? *Research Affiliates Publications*.

27. Vayanos, D. and Woolley, P. (2013). An institutional theory of momentum and reversal. *Review of Financial Studies*, 26(5), 1087–1145.

28. Ang, A., Hodrick, R.J., Xing, Y., and Zhang, X. (2006). The cross-section of volatility and expected returns. *Journal of Finance*, 61(1), 259–299.

29. Asness, C.S., Moskowitz, T.J., and Pedersen, L.H. (2013). Value and momentum everywhere. *Journal of Finance*, 68(3), 929–985.

30. Blitz, D., Hanauer, M.X., Vidojevic, M., and Arnott, R.D. (2016). How can 'quality' investing outperform? *Research Affiliates Publications*.

31. Novy-Marx, R. and Fama, E.F. (2015). A five-factor asset pricing model. National Bureau of Economic Research, Working Paper No. 20550.

32. Hou, K., Xue, C., and Zhang, L. (2015). Digesting anomalies: An investment approach. *Journal of Financial Economics*, 119(1), 175–220.

A Implementation Details

The synthetic data-generating process combines geometric Brownian motion, mean reversion, and Poisson jumps. Price dynamics follow:

$$dS_t = (\mu + \kappa(\bar{S} - \log S_t))S_t\,dt + \sigma S_t\,dW_t + S_t\,J_t\,dN_t$$

where:

- $\mu = 0.07$ (annualized drift)
- $\sigma = 0.28$ (annualized volatility)
- $\kappa = 0.025$ (mean-reversion speed)
- $J_t \sim \mathcal{N}(0, 0.06^2)$ (jump magnitude)
- $N_t \sim \text{Poisson}(\lambda = 0.01)$ per day (jump arrival)

The experiment involves 30 stocks over 10 years (2010–2019) with 15 percent experiencing delisting, 60 percent constituting the survivorship-biased index, and full stochastic volume generation.