# Feature Speed, Transaction Costs, and the Limits of Machine Learning in Emerging Markets

Anonymous Author(s)

## Abstract

Across 15 emerging equity markets, machine-learning models produce stronger gross stock-selection signals than momentum sorts, yet the advantage disappears under our high-cost EM calibration. We examine this prediction–portfolio disconnect along three margins: characteristic persistence, the sensitivity of model-implied rankings, and the portfolio update rule. A pre-OOS Feature Speed Index (FSI), based on characteristic rank persistence, predicts implementation burden across 20 random, equal-sized Ridge feature sets chosen before evaluation. At the feature-set level, FSI has Spearman correlations of 0.83 with turnover and 0.80 with gross-to-net Sharpe loss; leave-set-out cross-validation gives $R^2 = 0.51$ for both. Restricting Ridge regression to slow characteristics reduces monthly turnover from 23% to 10% and raises average net Sharpe from 0.24 to 0.36, compared with 0.27 for momentum. The improvement appears in 10 of 15 markets but does not reach conventional significance in the pooled block bootstrap (one-sided $p = 0.12$). Holding the feature set fixed, tree-based forecasts are more sensitive to small input perturbations than Ridge forecasts, while prediction smoothing and less frequent rebalancing further reduce turnover. Their net benefits, however, depend on the model and update schedule. The results identify feature persistence and model-induced ranking stability as constraints on ML implementability in high-cost emerging markets.

## CCS Concepts

• **Social and professional topics** → **Financial markets**; • **Computing methodologies** → **Machine learning**.

## Keywords

machine learning, emerging markets, transaction costs, portfolio construction, factor investing

**ACM Reference Format:**

Anonymous Author(s). 2026. Feature Speed, Transaction Costs, and the Limits of Machine Learning in Emerging Markets. In *Proceedings of the 7th ACM International Conference on AI in Finance (ICAIF '26), November 14–17, 2026, Milan, Italy*. ACM, New York, NY, USA, 8 pages.

## 1 Introduction

Machine-learning models have improved out-of-sample return prediction in both US and emerging-market equities. Gu, Kelly, and Xiu [8] document gains for US equities, and Hanauer and Kalsbach [9] extend this evidence to emerging markets (EM). Yet prediction quality is not the same as portfolio implementability. In our 15-market sample, Ridge generates a stronger gross signal than a momentum sort, with a Sharpe ratio (SR) of 0.57 versus 0.49, but underperforms it under our baseline high-cost calibration (0.24 vs. 0.27). The question is therefore not only whether ML predicts returns, but whether its rankings remain stable enough to trade.

Jensen, Kelly, Malamud, and Pedersen [10] (JKMP) show that cost-agnostic ML overweights "fleeting small-scale characteristics" and develop an implementable efficient frontier that integrates costs into the optimization objective. We build on their diagnosis by distinguishing three sources of instability. Fast-moving inputs create feature-induced churn, the fitted model can add further ranking instability, and the rebalancing rule determines how much of either source reaches the portfolio. We refer to their combined effect as *effective prediction churn*.

Our central contribution is an upstream diagnostic of the first channel. The Feature Speed Index (FSI), computed from a fixed pre-OOS window, orders implementation burden across 20 random, equal-sized Ridge feature sets: its feature-set-mean correlation is 0.83 with turnover and 0.80 with cost-induced Sharpe loss. Among five smooth monthly strategies, the cost–speed relation strengthens as trading costs rise. This relation matters in EM because higher trading costs amplify small differences in prediction stability. Because FSI measures input persistence only, prediction smoothing and less frequent rebalancing can reduce turnover without changing FSI.

Our analysis uses 154 firm-level characteristics from the JKP Global Factor Data [11], walk-forward validation, and Corwin-Schultz [3] transaction costs. Three results stand out:

(1) **Transaction costs eliminate ML's gross advantage.** Ridge's gross SR advantage over MomentumSort (0.57 vs. 0.49) disappears after costs (0.24 vs. 0.27) because Ridge turns over 23% per month versus 17% for the sort. Tree models combine strong gross signals with still greater turnover and lower net performance.

(2) **Feature speed accounts for much of the implementation gap.** In a three-market mechanism diagnostic, ML assigns 43–47% of feature importance to fast characteristics, although they constitute 25% of the set. Across 20 random 57-characteristic Ridge sets, FSI correlates 0.83 with turnover and 0.80 with gross-to-net SR loss. Restricting Ridge to 57 slow characteristics cuts turnover from 23% to 10% and raises mean net SR from 0.24 to 0.36, compared with 0.27 for MomentumSort. The improvement appears in 10 of 15 markets but does not reach conventional significance in the pooled block bootstrap (one-sided $p = 0.12$).



*ICAIF '26, Milan, Italy*

(3) **Model sensitivity and portfolio updating create additional churn.** Holding the feature set fixed, tree-based forecasts are more sensitive to small input perturbations than Ridge forecasts. Fixed EMA raises XGBoost's net SR to 0.34, although in-sample selection reaches only 0.24; quarterly rebalancing lifts standard Ridge to 0.31–0.36, depending on the anchor month, without changing its feature set. These results show that prediction smoothing and less frequent rebalancing can improve net performance by reducing how often forecast changes are traded.

The empirical patterns are not confined to the baseline specification. Market-holdout tests show that the FSI relation is not driven by any single sampled EM market, while a US large-cap exercise preserves the relative ordering of Ridge portfolios trained on slow and fast features. These results complement JKMP's cost-aware portfolio optimization by showing that feature persistence can serve as an upstream screen before model training. They also provide evidence from ML portfolios consistent with Gârleanu and Pedersen's [7] prediction that transaction costs favor slow-decaying signals.

## 2 Related Work

Machine-learning models have produced substantial gains in cross-sectional return prediction. Gu et al. [8] establish this result for US equities, and Hanauer and Kalsbach [9] extend the evidence to emerging markets. Whether these predictive gains survive portfolio implementation is less clear. Azevedo et al. [1] document monthly turnover of 120–140% and cost-induced performance reductions of 13–40% in ML strategies, while Esakia and Goltz [5] find that simpler approaches remain competitive in long-only portfolios. These findings are consistent with the broader evidence that transaction costs eliminate many anomaly profits [12], favor diversified characteristic combinations [4], and reward low-turnover portfolio construction [6].

The closest study is JKMP [10], who incorporate transaction costs directly into ML portfolio optimization and show that cost-agnostic models place excessive weight on "fleeting" characteristics. We examine a complementary upstream mechanism. Our Feature Speed Index measures characteristic persistence in a fixed pre-OOS window, and our decomposition separates turnover arising from the inputs from instability introduced by the model. This approach allows us to test whether restricting the information set to persistent characteristics can improve net performance before transaction-cost estimates enter the optimization problem.

The mechanism is related to Gârleanu and Pedersen [7], who show that transaction costs increase the optimal weight on slow-decaying signals. We examine this prediction in ML portfolios built from 154 characteristics across 15 emerging markets. The results further distinguish the persistence of the underlying signals from instability introduced by the prediction model and the rebalancing rule.

## 3 Data and Methodology

### 3.1 Data

We use monthly equity data for 15 emerging markets from the JKP Global Factor Data [11]: BRA, CHN, GRC, IDN, IND, KOR, MEX, MYS, PHL, POL, SAU, THA, TUR, TWN, and ZAF. Each provides 154 firm-level characteristics and one-month stock returns denominated in US dollars. The prediction target is the raw one-month return (`ret_1_0`). For each market-month, we restrict the universe to the top $N$ stocks by lagged market capitalization ($N \in [50, 300]$; most at 100). All predictors are lagged one month. Market capitalization is used only for the universe screen, not as a predictor.

### 3.2 Strategies

The headline comparison contains six strategies: two factor sorts (MomentumSort: 12-month return with 1-month skip; CompositeSort: the average cross-sectional rank of momentum, four quality signals, and low volatility), three ML models (Ridge $\alpha$=100; XGBoost and LightGBM with depth 3 and 200 trees), and a MomentumML_Blend (80% momentum + 20% Ridge rank). For the two tree models, a depth-3, 100-tree XGBoost selector retains the 20 most important features inside each training window. All use top-quintile, long-only, equal-weighted portfolios with a 20% rebalancing buffer. All ML models train on cross-sectional rank targets: $\tilde{y}_{i,t} = \text{Rank}(r_{i,t})/N_t$.

### 3.3 Walk-Forward Validation

We use expanding-window walk-forward with 60-month minimum training. For each test month $t$, predictors and the universe screen use information dated $t-1$. The model is trained only on returns through $t-1$, then used to form the portfolio whose return is measured in month $t$. The OOS period begins between June 2017 and July 2020 (market-dependent), ends in February 2025, and contains 56–93 months.

### 3.4 Transaction Cost Model

Costs use the one-month-lagged Corwin-Schultz [3] half-spread. We clip each traded name's half-spread to 20–200 bps and average across entering and exiting names, obtaining $\bar{c}_t$. We set $c_t(0) = 0$; for $\kappa > 0$, the charged cost per trade is $c_t(\kappa) = \text{clip}(\kappa\bar{c}_t, 20, 200)$ bps, so the final cap is applied after scaling. Missing spreads are replaced by that month's cross-sectional median. We charge entries and exits; reported one-way turnover divides total trades by twice the new portfolio size. We use $\kappa = 3$ as the baseline high-cost EM stress calibration and report the full cost curve rather than treating it as known. The multiplier allows for taxes, commissions, FX conversion, market impact, and capacity costs absent from the spread estimate; it is not an estimate of those components. For robustness, we also test flat cost schedules from 10 to 100 bps per trade.

### 3.5 Performance Metrics

Because all returns are denominated in US dollars, Sharpe ratios use monthly excess returns over the contemporaneous US one-month Treasury bill reported with the Kenneth French emerging-market factor data and are annualized by $\sqrt{12}$. Gross and net SRs use the same risk-free series; net returns additionally subtract transaction costs. We compute statistics separately over each market's OOS window and report equal-weighted means across the 15 markets. Performance relative to the cap-filtered equal-weighted universe is

reported as an annualized information ratio of $r_p - r_{\text{EW}}$; the risk-free return cancels from this active return.

### 3.6 The Feature Speed Index and the Cost-Drag Relation

To quantify ex-ante exposure to input-induced prediction instability, we define the Feature Speed Index (FSI) as the equal-weight average instability of a strategy's feature set $S$. Let $S_m^{\text{obs}} \subseteq S$ denote the characteristics with estimable pre-OOS rank autocorrelation in market $m$:

$$\text{FSI}_m(S) = \frac{1}{|S_m^{\text{obs}}|} \sum_{j \in S_m^{\text{obs}}} (1 - \text{AC}_{j,m}), \tag{1}$$

where $\text{AC}_{j,m}$ is the cross-sectional Spearman rank autocorrelation of feature $j$ in market $m$. For each market, we estimate it once using a fixed 60-month window that ends before that market's first OOS return and hold the resulting feature-set FSI fixed throughout the OOS diagnostic. Unavailable or constant characteristics are omitted from this average, but not from the model. Coverage, $|S_m^{\text{obs}}|/|S|$, is 91.2–100% (mean 98.7%) in the random-set test. The measure therefore uses no OOS observations or realized model coefficients. For single-feature sorts, $S$ contains only the sorting characteristic. FSI captures the input side of prediction churn. Model architecture and portfolio updating can amplify or dampen that churn, so we analyze them separately.

For smooth models rebalanced monthly, we estimate the following cross-sectional relation separately at each cost multiplier:

$$\Delta\text{SR}_{s,m,\kappa} = a_\kappa + b_\kappa \text{FSI}_{s,m} + \varepsilon_{s,m,\kappa}, \qquad \Delta\text{SR} = \text{SR}_{\text{gross}} - \text{SR}_{\text{net}}, \tag{2}$$

where $s$ indexes the strategy and $m$ the market. To isolate differences in feature speed from mechanical changes in transaction costs, we estimate the relationship separately at each $\kappa$. We also report a pooled regression of SR degradation on $\kappa \times \text{FSI}$ as a summary across cost levels. The diagnostic contains five feature sets: MomentumSort and Ridge using all, slow-plus-medium, slow, or fast characteristics. We evaluate tree models, prediction smoothing, and quarterly rebalancing separately because they alter other components of prediction churn.

## 4 Results

### 4.1 ML's Gross Advantage Disappears After Costs

With rank-target training, Ridge performs similarly to MomentumSort across 15 EM markets at 3× costs: average net SRs are Blend 0.29, MomSort 0.27, Ridge 0.24, CompositeSort 0.21, XGBoost 0.13, and LightGBM 0.08. Ridge wins 7/15 markets (two-sided Wilcoxon $p = 0.49$). XGBoost's lower mean is suggestive but not significant at 5% ($p = 0.073$), while LightGBM is significantly worse ($p = 0.005$).

Table 1 isolates the cost interaction. Ridge generates a *stronger* gross signal (0.57) than MomSort (0.49), but 23% turnover vs. 17% reverses the ordering after costs. Cost drag is annualized realized trading cost, defined as 12 times the mean monthly portfolio cost.

The per-market results in Table 2 show no uniform ordering. Ridge outperforms MomentumSort in 7 of 15 markets, including TWN (0.59 vs. 0.42) and CHN (0.12 vs. −0.02), but trails in markets such as IND (0.25 vs. 0.53) and IDN (0.29 vs. 0.50).

**Table 1: Gross vs. net Sharpe ratios at 3× costs, averaged across 15 EM markets. Ridge has a stronger gross signal but higher turnover erases the advantage.**

| Strategy | Gross SR | Net SR | Turnover | Cost Drag |
|---|---|---|---|---|
| MomentumSort | 0.49 | 0.27 | 17.0% | ~590 bps |
| Ridge | 0.57 | 0.24 | 23.3% | ~700 bps |
| XGBoost | 0.62 | 0.13 | 35.6% | ~1,070 bps |
| LightGBM | 0.60 | 0.08 | 38.2% | ~1,160 bps |
| Blend (80/20) | 0.54 | 0.29 | 19.0% | ~640 bps |

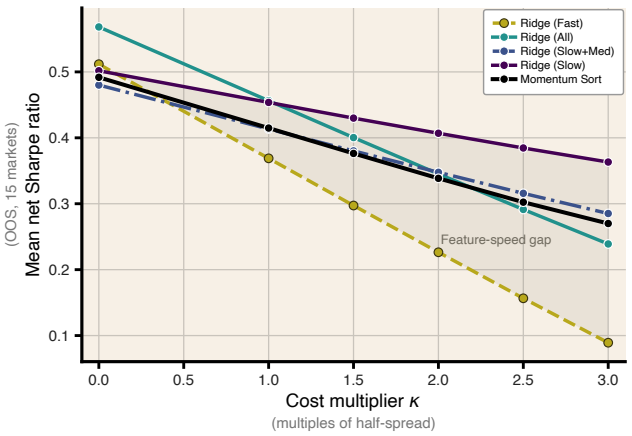


**Figure 1: Net SR vs. cost multiplier $\kappa$. Fast features degrade ~3× faster than slow features. The feature-speed gap widens monotonically with $\kappa$ and exceeds 0.25 SR at 3× EM costs.**

### 4.2 The Cost–Speed Interaction

We classify the 151 modeled predictors into three speed categories: **SLOW** (57 accounting fundamentals, rank AC 0.97), **MEDIUM** (56 intermediate features, AC 0.83), and **FAST** (38 market microstructure variables, AC 0.67). Across the fixed pre-OOS snapshots, the three classes have significantly different rank persistence (Kruskal-Wallis $p = 7.36 \times 10^{-12}$).

Figure 1 plots net Sharpe ratio against the cost multiplier $\kappa$. In the aggregate EM results, fast features produce competitive gross SRs before costs (0.51 vs. 0.50 for slow). As costs rise, however, fast-feature performance degrades ~3× faster (degradation 0.42 vs. 0.14 SR at 3×), because turnover differences make the cost gap widen as $\kappa$ rises. The relation is only approximately linear because costs are capped after scaling. The shortfall is therefore primarily a cost interaction in this sample, not an absence of gross predictive content.

The higher EM cost regime makes the economic value of input stability especially visible. This motivates testing feature selection upstream, while later sections examine complementary output and execution interventions.

**Table 2: Per-market OOS net Sharpe ratios at 3× costs. $N$ is the universe size and $T$ is the number of OOS months. Δ is Ridge minus MomSort, so positive values favor Ridge.**

| Market | $N$ | $T$ | MomSort | Ridge | Δ | Market | $N$ | $T$ | MomSort | Ridge | Δ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| BRA | 100 | 83 | 0.01 | −0.13 | −0.14 | PHL | 100 | 87 | −0.12 | −0.05 | 0.07 |
| CHN | 300 | 78 | −0.02 | 0.12 | 0.14 | POL | 100 | 69 | 0.48 | 0.49 | 0.01 |
| GRC | 50 | 78 | 0.34 | 0.61 | 0.27 | SAU | 100 | 56 | 1.00 | 0.69 | −0.31 |
| IDN | 100 | 80 | 0.50 | 0.29 | −0.21 | THA | 100 | 84 | −0.09 | −0.17 | −0.08 |
| IND | 200 | 81 | 0.53 | 0.25 | −0.28 | TUR | 100 | 79 | 0.28 | 0.39 | 0.11 |
| KOR | 200 | 86 | −0.04 | −0.03 | 0.01 | TWN | 200 | 78 | 0.42 | 0.59 | 0.17 |
| MEX | 100 | 93 | 0.15 | 0.07 | −0.08 | ZAF | 100 | 93 | 0.30 | 0.15 | −0.15 |
| MYS | 100 | 90 | 0.30 | 0.29 | −0.01 | | | | | | |

**Table 3: Use of characteristics by speed class. Ridge and XGBoost importance and the Ridge prediction-change attribution are estimated for THA, KOR, and TWN over 2021–2023; pre-OOS rank AC is averaged across the 15 fixed market snapshots.**

| Speed | # Feat. | Pre-OOS AC | % Import. | % Pred. Δ |
|---|---|---|---|---|
| SLOW | 57 | 0.97 | 31–34% | 12.6% |
| MEDIUM | 56 | 0.83 | 22–23% | 24.1% |
| FAST | 38 | 0.67 | 43–47% | 63.4% |

## 4.3 How Fast Features Enter Portfolio Predictions

We classify characteristics by their construction and expected update frequency, without using return, cost, or OOS performance outcomes. We then examine THA, KOR, and TWN over 2021–2023 to see how the fitted models use features with different persistence. Fast characteristics make up 25% of the candidate set, but account for 43% of Ridge importance and 47% of XGBoost importance. In a separate Ridge attribution, they account for 63.4% of month-to-month changes in predicted stock rankings. Slow characteristics, by comparison, make up 38% of the feature set but account for only 12.6% of those changes. In these three markets, the fitted predictions respond disproportionately to the least persistent inputs.

At the individual-characteristic level, one-month rank ICs are similar for the slow and fast groups (0.027 vs. 0.025). After 12 months, however, slow characteristics retain 69% of their initial IC, compared with 57% for fast characteristics (Wilcoxon $p = 0.01$ across 15 markets; Figure 2). Fast characteristics therefore remain predictive, but their lower persistence makes them more costly to trade. This suggests a direct test: hold the model and portfolio rule fixed, and remove the least persistent inputs.

## 4.4 Slow-Feature Ridge

We train Ridge using only the 57 slow characteristics, leaving the model, hyperparameters, and portfolio construction unchanged. Table 4 shows the effect. At 3× costs, turnover falls from 23.3% for Ridge_All to 10.2% for Ridge_Slow, while net SR rises from 0.24 to 0.36. MomentumSort reaches 0.27 at 17.0% turnover.

Ridge_Slow has a higher net SR than MomentumSort in 10 of 15 markets. For inference, we average each strategy's returns across

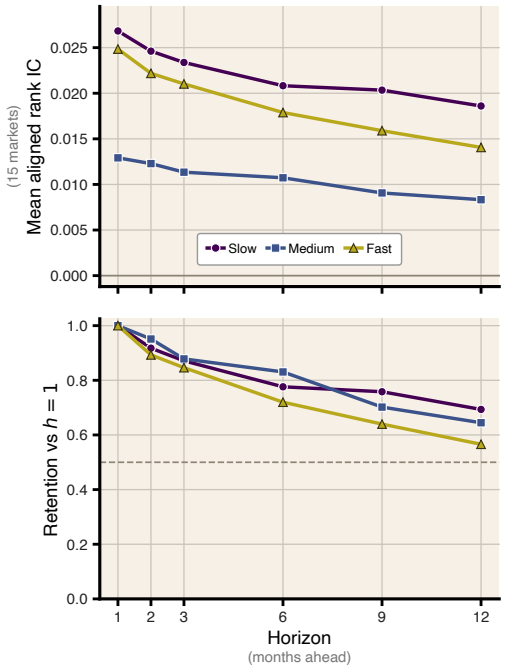


**Figure 2: Model-free rank-IC decay by feature speed class. Slow features retain 69% of their one-month IC at 12 months versus 57% for fast features; no model is fitted anywhere in this analysis.**

available markets by calendar month and synchronously resample the aligned 93-month series in six-month blocks. Across 10,000 draws, the difference is positive in 88%; the one-sided $p$-value is 0.12 and the percentile 95% confidence interval is $[-0.10, 0.52]$. Across markets, Ridge_Slow raises the SR by 0.12 relative to Ridge_All. Jointly resampling markets and synchronized six-month calendar blocks gives a 95% CI of $[0.02, 0.26]$ and a one-sided $p$-value of 0.008. Pooling all market return observations instead gives a larger difference of 0.21 (95% CI $[0.07, 0.44]$) because markets with longer return histories receive more weight.

The forecast statistics show the same contrast. Ridge_Slow combines an IC of 0.031 with score stability of 0.939. Ridge_Fast has a lower IC (0.013) and much less stable scores (0.539). Restricting

**Table 4: Feature-set comparisons and benchmarks. Mean OOS results across 15 markets at 3× costs. Stability is the rank correlation of stock scores in adjacent months.**

| Strategy | Net SR | TO | # Feat. | IC | Stab. |
|---|---|---|---|---|---|
| **Ridge_Slow** | **0.36** | **10.2%** | 57 | 0.031 | 0.939 |
| AC > 0.75 | 0.36 | 13.0% | ~120 | — | — |
| AC tercile (full sample) | 0.36 | 13.0% | ~127 | — | — |
| Ridge_SlowMed | 0.29 | 14.5% | 113 | — | — |
| MomentumSort | 0.27 | 17.0% | 1 | 0.031 | 0.894 |
| Ridge_All | 0.24 | 23.3% | 151† | 0.022 | 0.665 |
| Ridge_Fast | 0.09 | 30.1% | 38 | 0.013 | 0.539 |

† 151 modeled predictors after excluding `size_grp`, `me`, and `market_equity` from 154 source feature columns. Dashes indicate that IC and stability were not computed.

Ridge to slow characteristics therefore improves both the predictive content and the persistence of its combined forecast.

The turnover reduction could reflect the smaller feature set rather than feature persistence. We therefore compare the fixed slow set with 20 random 57-feature subsets drawn before evaluating returns. With seed 20260803, we draw each set without replacement from 151 characteristics and fix it across markets; pairs overlap by 21 characteristics on average. Their gross SRs are similar (0.50 for slow versus 0.49 on average for random), but the random subsets turn over 20.2% and achieve a net SR of 0.21, compared with about 10% and 0.36 for the slow set. The slow set has lower turnover in all 15 markets and a higher net SR in 14 (one-sided Wilcoxon $p < 0.001$). Feature count alone therefore does not explain the result.

We also replace the fixed list with a persistence cutoff (rank AC > 0.75) re-estimated from the trailing 60 training months. It selects about 120 features and produces a net SR of 0.36 at 13% turnover, without using a transaction-cost estimate. For comparison, a tercile based on full-sample rank AC gives similar results, but this descriptive check uses future feature observations and is not a feasible selection rule.

Relative to an equal-weight portfolio of all eligible stocks, Ridge_Slow also performs best. At 3× costs, its information ratio is 0.18, compared with 0.07 for MomentumSort, −0.15 for Ridge_All, and −0.46 for Ridge_Fast. Ridge_Slow outperforms the equal-weight benchmark in 10 of 15 markets.

## 4.5 Slow Features Do Not Eliminate Tree-Model Turnover

We next ask whether the slow-feature screen has the same effect for tree models. Table 5 compares unsmoothed model–feature combinations on a common sample. This exercise uses depth-3, 100-tree models without the top-20 selector and a longer OOS window than the earlier tables (about 102 months per market), so the levels are not directly comparable.

XGBoost has the highest gross SR (0.66), but its net SR falls to 0.19, below both Ridge_All (0.26) and Ridge_Slow (0.38). Restricting XGBoost to slow features lowers turnover from 34.9% to 28.9%, but does not improve net performance: its net SR falls from 0.19 to 0.17. In contrast, Ridge_Slow turns over 9.9% and retains most of its gross performance. Across all nine ML model–feature combinations in the underlying comparison, the rank correlation between gross and net SR is −0.45.

**Table 5: Selected prediction-portfolio results on the common tree-comparison sample, sorted by gross SR.**

| Model | Strategy | Gross SR | Net SR | SR Lost | TO |
|---|---|---|---|---|---|
| XGB | All | 0.659 | 0.190 | 0.469 | 34.9% |
| LGBM | All | 0.654 | 0.172 | 0.482 | 35.5% |
| Ridge | All | 0.583 | 0.257 | 0.326 | 23.0% |
| XGB | Slow | 0.562 | 0.174 | 0.388 | 28.9% |
| MomSort | — | 0.525 | 0.293 | 0.232 | 17.2% |
| Ridge | Slow | 0.520 | 0.383 | 0.137 | 9.9% |

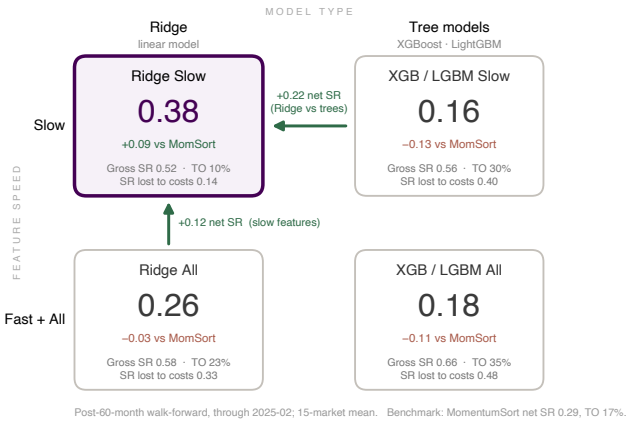


**Figure 3: Feature speed and model type. Ridge with slow inputs has the lowest turnover and the highest net SR.**

Output smoothing helps, but does not close the gap. On the main OOS sample, a fixed $\alpha = 0.2$ chosen post hoc raises XGBoost's net SR from 0.13 to 0.34 and lowers turnover from 36% to 12%. In-sample selection reaches a net SR of 0.24 at 25% turnover, below MomentumSort's 0.27.

Figure 3 summarizes the two margins. On the common tree-comparison sample, the gross-to-net SR loss is 0.14 for Ridge_Slow, 0.33 for Ridge_All, 0.39 for XGB_Slow, and 0.47 for XGB_All. Slow inputs therefore account for only part of the turnover in tree portfolios. We test the remaining difference by perturbing each standardized slow characteristic with independent Gaussian noise in every third OOS cross-section (20 common draws across models; seed 20260803). Tied predictions receive their average rank. At 0.01 standard deviations, one minus the Spearman correlation between original and perturbed predictions is 0.001 for Ridge, 0.250 for XGBoost, and 0.237 for LightGBM. Both trees are more sensitive in all 15 markets (one-sided Wilcoxon $p < 0.001$), with the same ordering for perturbations from 0.05 to 0.20. This sensitivity is consistent with the higher turnover of the tree portfolios on the same slow inputs.

## 4.6 Joint Test of Feature Speed and Model Type

The tree comparisons suggest that feature speed and model type both contribute to cost drag. We test them jointly by pooling the

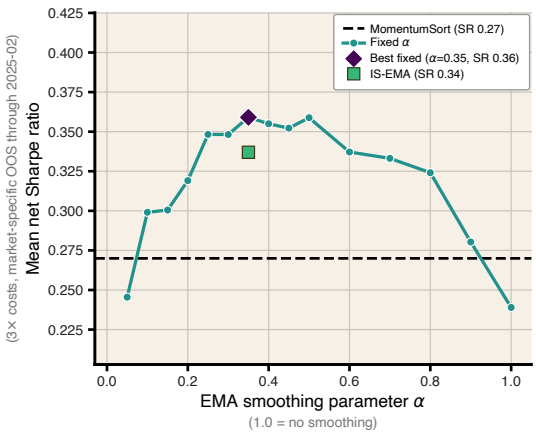


**Figure 4: Ridge net SR at 3× costs across fixed EMA weights. The green square shows in-sample selection; the dashed line shows MomentumSort.**

five smooth monthly strategies used in Equation 2 with XGBoost and LightGBM, across 15 markets and four cost levels:

$$\Delta \mathrm{SR}_{s,m,\kappa} = a + b(\kappa \times \mathrm{FSI}_{s,m}) + \gamma(\kappa \times \mathrm{Tree}_s) + \varepsilon_{s,m,\kappa}, \qquad (3)$$

where $\mathrm{Tree}_s$ equals one for XGBoost and LightGBM. For these models, FSI is calculated from the full candidate feature set, not the 20 features selected within each training window.

Across 420 strategy–market–cost cells, the regression with only $\kappa \times \mathrm{FSI}$ has $R^2 = 0.490$. Adding the tree interaction raises $R^2$ to 0.779. The FSI coefficient is 0.396, while the tree coefficient is 0.080 per unit of $\kappa$ (market-clustered 95% CI $[0.065, 0.094]$). At $\kappa = 3$, this corresponds to approximately 0.24 additional SR degradation for tree models after accounting for candidate-set speed. The result also holds in market holdouts: when the regression is estimated without each market and then evaluated on the omitted market, the pooled leave-one-market-out cross-validated $R^2$ is 0.732. The coefficient remains descriptive: cost levels reuse return paths, there are only 15 market clusters, and tree FSI uses candidate rather than selected features. In this set of strategies and markets, tree models therefore lose more SR to costs than feature speed alone would predict. We do not interpret the estimate as a general ranking of model architectures.

## 4.7 Smoothing Prediction Ranks

Instead of changing the feature set, we can smooth Ridge's prediction ranks before forming the portfolio. We use an EMA: $\hat{r}^{\mathrm{smooth}}_{i,t} = \alpha \cdot \hat{r}^{\mathrm{raw}}_{i,t} + (1-\alpha) \cdot \hat{r}^{\mathrm{smooth}}_{i,t-1}$, where a lower $\alpha$ places more weight on past ranks. Among the fixed values in Figure 4, net SR peaks at 0.36 when $\alpha = 0.35$. Because this value is chosen after observing the full curve, we also select $\alpha$ separately at each retraining date using the highest net SR over the most recent 24 training months. Those months are scored by the model fitted on the same training window, rather than by nested OOS predictions, so we label the result in-sample selected. It reaches a net SR of 0.34 and outperforms MomentumSort's 0.27 in 10 of 15 markets.

**Table 6: Ridge prediction smoothing across 15 markets. IS-EMA selects $\alpha$ using fitted performance over the latest 24 training months.**

| Strategy | Net SR | Gross SR | TO | Stability† | Wins |
|---|---|---|---|---|---|
| Ridge (no smooth) | 0.24 | 0.57 | 23.3% | 0.750 | 7/15 |
| MomentumSort | 0.27 | 0.49 | 17.0% | — | — |
| **Ridge IS-EMA** | **0.34** | **0.48** | **10.3%** | **0.919** | **10/15** |

†Prediction stability uses the full post-60-month walk-forward path; other columns are OOS.

Table 6 shows where the improvement comes from. Smoothing lowers Ridge's gross SR from 0.57 to 0.48, but cuts turnover from 23.3% to 10.3% and raises prediction stability from 0.750 to 0.919. The reduction in trading more than offsets the weaker gross performance, raising net SR from 0.24 to 0.34.

Smoothing is less reliable for XGBoost. As discussed in Section 4.5, a fixed $\alpha = 0.2$ chosen post hoc raises its net SR from 0.13 to 0.34, whereas in-sample selection reaches 0.24. The selector therefore recovers 0.11 of the 0.21 post-hoc improvement for XGBoost, compared with 0.10 of 0.12 for Ridge. Recent in-sample performance appears to provide a less reliable guide to the smoothing parameter for XGBoost.

## 4.8 The Cost–Speed Relation and Its Limits

The preceding interventions show that feature speed is not the only source of turnover. We use FSI for a narrower purpose: to compare the trading burden of feature sets before a model is estimated. We first estimate Equation 2 across five smooth monthly strategies, 15 markets, and four cost levels. At each $\kappa \in \{1, 1.5, 2, 3\}$, FSI explains 62.2–63.6% of the cross-sectional variation in cost-induced SR degradation. The slopes rise from 0.321 at 1× costs to 0.961 at 3× costs, so faster feature sets lose progressively more performance as trading costs rise. The pooled regression has a slope of 0.379 and $R^2 = 0.708$. Across the corresponding 75 strategy–market cells, FSI also explains 72% of realized turnover. Because each cost level is applied to the same underlying return path, this regression mainly shows how trading costs amplify differences associated with feature speed.

We therefore test FSI on the 20 random 57-characteristic subsets introduced in Section 4.4. The deliberately slow set is excluded when estimating the relation. Figure 5 plots the resulting feature-set means. Across the random sets, mean pre-OOS FSI ranges from 0.096 to 0.221. At the feature-set level, its Spearman correlation is 0.83 with turnover and 0.80 with the gross-to-net SR loss at 3× costs. Leaving one feature set out across all 15 markets gives cross-validated $R^2 = 0.51$ for both outcomes. The corresponding within-market values are 0.12 and 0.05, so FSI orders feature sets more reliably than it predicts market-specific outcomes. Slopes remain positive in two-way bootstraps and with controls for markets, speed composition, and coverage. Set overlap may make the nominal $p < 0.001$ values too small, so we emphasize the rank correlations and cross-validation.

Ridge_Slow provides a separate check. Its mean FSI is 0.033, below the range of the random sets, and both its turnover and SR loss are lower than the random-set regressions predict. It is therefore

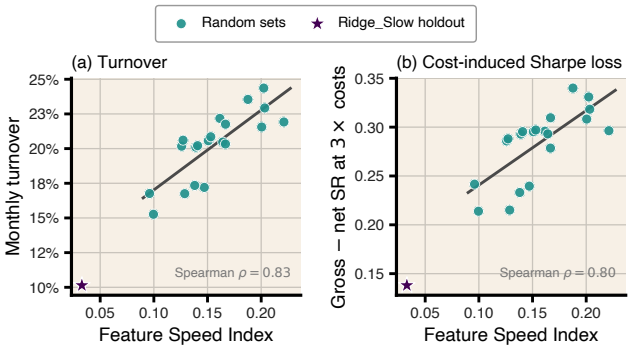


**Figure 5: FSI and implementation burden across 20 random, equal-sized Ridge feature sets. Points are feature-set averages across 15 markets, and fitted lines use the random sets only. Ridge_Slow (star) is excluded from estimation. Spearman correlations are 0.83 with turnover and 0.80 with gross-to-net SR loss.**

consistent with the relation without determining its slope. These tests give FSI support across distinct, equal-sized Ridge feature sets, although they do not make the relationship causal.

The random-set results concern differences across feature sets; they do not imply that FSI alone determines realized turnover. Holding the feature set fixed, quarterly rebalancing cuts Ridge_All's turnover from 23% to about 11%; across the three anchors, net SR is 0.31–0.36 versus 0.24 monthly. Tree models also turn over more than Ridge on the same slow inputs. FSI therefore measures the turnover pressure embedded in the predictors; the model and rebalancing rule determine how much of that pressure reaches the portfolio.

The original five-strategy relation is also stable across the sampled markets. Mean OOS $R^2$ is 0.59 when one market is held out and 0.67 across all leave-5-out splits. The feature classification itself changes little: 52 of the 57 slow characteristics retain that label in every market, and the mean cross-market Jaccard similarity is 0.97. These results show that no small group of sampled markets drives the pattern; they do not establish performance in new market populations.

## 5 Robustness and Alternatives

*US equities.* We apply the same pipeline to the 500 largest US stocks over 2021–2025 (50 OOS months). Ridge_Slow beats Ridge_Fast at every cost level in Table 7, with the gap widening from 0.23 SR before costs to 0.70 at 3×. MomentumSort remains best throughout. The US sample therefore preserves the slow-feature advantage within Ridge.

*Cost-aware training.* We next ask whether training directly on trading costs can improve on Ridge_Slow. Each model uses the same 57 slow features and is initialized from the Ridge rank-target fit. The cost-aware variants are then fine-tuned using differentiable soft portfolio weights and the loss

$$\mathcal{L}(\theta) = -\,\mathrm{SR}\big(r_t^{\mathrm{gross}} - C_t(\theta)\big) + \lambda\,\overline{C_t(\theta)} + 10^{-3}\|\theta\|_2^2, \qquad (4)$$

**Table 7: US equities: net SR by strategy and cost multiplier. Top 500 stocks, OOS 2021–2025.**

| $\kappa$ | MomSort | Ridge_All | Ridge_Slow | Ridge_Fast |
|---|---|---|---|---|
| 0× | 0.626 | 0.420 | 0.475 | 0.245 |
| 1× | 0.545 | 0.284 | 0.432 | 0.046 |
| 2× | 0.463 | 0.148 | 0.389 | −0.153 |
| 3× | 0.382 | 0.012 | 0.346 | −0.352 |

**Table 8: Cost-aware models using the 57 slow features. Mean OOS results across 15 markets at 3× costs.**

| Strategy | Net SR | Turnover | Gross SR |
|---|---|---|---|
| **Ridge_Slow** | **0.36** | 10.2% | 0.50 |
| CostAware_Linear | 0.22 | 7.4% | 0.32 |
| MomentumSort | 0.27 | 17.0% | 0.49 |
| CostAware_Lipschitz | 0.20 | 12.9% | 0.37 |
| Lipschitz_NN | 0.05 | 32.7% | 0.44 |

where the differentiable training cost is $C_t(\theta) = \kappa \sum_i \bar{c}_{i,t} |w_{i,t}(\theta) - w_{i,t-1}(\theta)|$, using the 1× clipped spread defined in Section 3.4. We set $\kappa = 3$ during training, matching the high-cost evaluation. The penalty $\lambda$ is chosen from fitted performance within each training window. We estimate a linear model and a spectral-normalized MLP with this objective; a second spectral-normalized MLP is trained on rank-target MSE without the cost term. Final performance uses the same hard portfolio and capped cost rule as the main analysis; the soft objective is used only for estimation.

None of the alternatives exceeds Ridge_Slow in Table 8. CostAware_Linear lowers turnover to 7.4%, but its gross SR falls to 0.32 and its net SR to 0.22. The two MLPs reach net SRs of 0.20 and 0.05. In these specifications, lower turnover does not offset the loss in gross performance.

The main EM ordering is also unchanged after excluding Greece, the only 50-stock universe (Ridge_Slow 0.35 vs. MomentumSort 0.27), and under value weighting (0.29 vs. 0.22; Ridge_All 0.19).

## 6 Discussion

*Sources of portfolio turnover.* Portfolio turnover can originate at three points in the forecasting and implementation process. The first is the persistence of the underlying characteristics. The second is the sensitivity of model-implied rankings to changes in those characteristics. The third is the portfolio update rule, which determines how frequently changes in rankings are translated into trades. Our empirical exercises examine each channel separately. Ridge_Slow addresses the first by restricting the information set to persistent characteristics. The perturbation test shows that, on the same slow inputs, XGBoost and LightGBM rankings respond more sharply than Ridge rankings to small changes in characteristics. Persistent inputs alone therefore do not ensure stable rankings. EMA smooths the predicted ranks directly, while quarterly rebalancing reduces the frequency with which ranking changes are implemented. FSI measures only the first channel, using a fixed 60-month pre-OOS window. It is therefore unaffected by model choice, prediction smoothing, or rebalancing frequency. FSI should

accordingly be interpreted as a measure of feature-induced churn rather than total strategy turnover.

*Relation to theory and JKMP.* The results accord with Gârleanu and Pedersen's [7] prediction that costs increase the value of persistent signals: among smooth monthly strategies, those using more persistent characteristics lose less SR to costs. JKMP [10] instead incorporate costs into portfolio optimization. Our screen acts before model estimation and requires no cost estimate, so the approaches are complementary. The cost-aware specifications considered here do not outperform Ridge_Slow, but this narrow comparison does not establish that screening is generally preferable. Combining the two is a natural extension.

*Relation to prior EM evidence.* Hanauer and Kalsbach [9] report stronger long-short performance for ML in emerging markets. Our gross ordering agrees, but we study long-only portfolios after estimated trading costs. Ridge's additional turnover then absorbs its gross advantage. The results therefore qualify the implementation of stronger predictions, not the predictive evidence itself.

*Practical implications.* FSI can identify feature sets that are likely to be costly to trade before a model is estimated. In our sample, excluding fast characteristics cuts Ridge turnover from 23% to 10% without changing monthly rebalancing, while net SR rises from 0.24 to 0.36. The difference from MomentumSort remains imprecisely estimated. Screening, EMA, and less frequent rebalancing can all reduce trading, but at different costs: screening discards potentially useful fast-moving information, EMA delays the response to new predictions, and less frequent rebalancing postpones portfolio updates. Similar net gains therefore need not represent the same economic trade-off.

*Limitations.* The OOS period spans 56–93 months per market and includes common shocks. Costs are estimated, and the 3× multiplier is a calibration. Random subsets match feature count, not missingness, characteristic families, or correlation structure. The perturbation test measures local sensitivity to artificial noise; split crossings and tied predictions may contribute to the tree response, so it is not a direct estimate of turnover. The large-cap sample and fixed tree settings also limit generalization [2, 8]. Without factor-adjusted alphas, style exposures may contribute to return differences; turnover is therefore more directly identified than the source of the net Sharpe improvement. Market holdouts rule out dominance by one sampled market, not performance in other markets or strategy classes.

*Future work.* Persistent-feature selection could be combined with cost-aware optimization, while a time-varying screen could adapt to changes in feature persistence without using future information. Objectives that penalize cross-sectional rank instability may address turnover more directly than standard prediction losses. Tests in smaller-cap universes and against realized execution costs would show how the results change with liquidity.

## 7 Conclusion

Across 15 emerging markets, Ridge has a stronger gross signal than MomentumSort but turns over more, leaving the two with similar net performance under the high-cost calibration. We organize this gap around three parts of the forecasting process: characteristic persistence, the sensitivity of model-implied rankings, and the portfolio update rule.

FSI measures the first component before model estimation. Across 20 random, equal-sized Ridge feature sets, its feature-set-mean correlations are 0.83 with turnover and 0.80 with gross-to-net SR loss. The five smooth monthly strategies show the same cost–speed relation as trading costs rise. Restricting Ridge to slow characteristics lowers turnover from 23% to 10% and raises net SR from 0.24 to 0.36. Its gross performance is similar to that of the random sets, so the turnover reduction is not simply a consequence of using fewer predictors. Ridge_Slow's advantage over Ridge_All is supported by paired inference; its advantage over MomentumSort remains imprecisely estimated.

The other two components also matter. On the same slow inputs, direct perturbations produce larger ranking changes for the tree models than for Ridge, and the trees retain higher turnover. EMA stabilizes predicted rankings, while quarterly rebalancing changes how often those rankings are traded. The cost-aware alternatives do not improve on Ridge_Slow in the specifications considered here. Market holdouts support the FSI relation, and the US exercise preserves the slow–fast ordering within Ridge.

Implementability is therefore a joint property of the inputs, prediction model, and trading rule. FSI offers an upstream measure of feature-induced turnover, but it does not summarize the full strategy or replace cost-aware portfolio design. Whether these relations extend beyond the sampled large-cap markets and calibrated costs remains open.

## References


[1] Vitor Azevedo, Christopher Hoegner, and Mihail Velikov. 2024. The Expected Returns on Machine-Learning Strategies. *Journal of Finance* (2024). Forthcoming.

[2] Andrew Y Chen, Matthias X Hanauer, and Tobias Kalsbach. 2025. Non-Standard Errors in Machine Learning: Evidence from Asset Pricing. *Working Paper* (2025).

[3] Shane A Corwin and Paul Schultz. 2012. A Simple Way to Estimate Bid-Ask Spreads from Daily High and Low Prices. *The Journal of Finance* 67, 2 (2012), 719–760.

[4] Victor DeMiguel, Alberto Martin-Utrera, Francisco J Nogales, and Raman Uppal. 2020. A Transaction-Cost Perspective on the Multitude of Firm Characteristics. *The Review of Financial Studies* 33, 5 (2020), 2180–2222.

[5] Mikheil Esakia and Felix Goltz. 2025. Machine Learning for Long-Only Portfolio Construction. *Working Paper* (2025).

[6] Andrea Frazzini, Ronen Israel, and Tobias J Moskowitz. 2018. Trading Costs. *Working Paper* (2018).

[7] Nicolae Gârleanu and Lasse Heje Pedersen. 2013. Dynamic Trading with Predictable Returns and Transaction Costs. *The Journal of Finance* 68, 6 (2013), 2309–2340.

[8] Shihao Gu, Bryan Kelly, and Dacheng Xiu. 2020. Empirical Asset Pricing via Machine Learning. *The Review of Financial Studies* 33, 5 (2020), 2223–2273.

[9] Matthias X Hanauer and Tobias Kalsbach. 2023. Machine Learning and the Cross-Section of Emerging Market Stock Returns. *Emerging Markets Review* 55 (2023), 101022.

[10] Theis Ingerslev Jensen, Bryan Kelly, Semyon Malamud, and Lasse Heje Pedersen. 2024. Machine Learning and the Implementable Efficient Frontier. *The Review of Financial Studies* (2024). Forthcoming.

[11] Theis Ingerslev Jensen, Bryan Kelly, and Lasse Heje Pedersen. 2023. Is There a Replication Crisis in Finance? *The Journal of Finance* 78, 5 (2023), 2465–2518.

[12] Robert Novy-Marx and Mihail Velikov. 2016. A Taxonomy of Anomalies and Their Trading Costs. *The Review of Financial Studies* 29, 1 (2016), 104–147.