LEARNING TO RANK WITH WEAK RETURN SIGNALS: SHORTCUT-ORTHOGONAL GRADIENT ALLOCATION FOR CROSS-SECTIONAL STOCK RANKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Cross-sectional stock ranking predicts relative future returns, so learning to rank fits the task. Weighted ranking objectives emphasize comparisons based on return differences, rank positions, or ranking importance. These methods decide which comparisons matter; whether the placement of the weights can also encourage a competing ordering when the directional signal is weak has received little attention. We show that weight allocation itself can induce an objective-level shortcut preference and that, in stock ranking, return magnitude is such a competing ordering. If weighting creates this preference, it should be visible before the model learns anything. At equal scores, the loss response toward a candidate shortcut depends only on training labels and comparison weights. We propose Shortcut-Orthogonal Gradient Allocation (SOGA), a diagnose-then-correct framework that measures the excess response over uniform weighting and minimally corrects risky allocations while retaining the original comparison emphasis as far as the constraints allow. Empirically, every tested base allocation flagged by the diagnostic degrades ranking relative to uniform weighting. Across the four primary equity panels, SOGA consistently improves flagged ranking objectives, with a corrected objective surpassing every tested uncorrected published or reference objective. The correction also transfers across backbones and outperforms uniform shrinkage in matched comparisons.

1 INTRODUCTION

Cross-sectional stock ranking orders stocks by future returns within each trading day (Feng et al., 2019; Yang et al., 2020; Li et al., 2024). Learning to rank suits the task because relative performance matters more than accurate prediction of every return. Pairwise objectives penalize score differences that disagree with the realized-return order. Weighted objectives further emphasize comparisons based on return differences, rank positions, or their effect on a ranking metric (Burges et al., 2005; 2006; Wang et al., 2018). Developed for tasks including document retrieval, these objectives have also been applied to stock ranking (Kwiatkowski & Chudziak, 2025).

These methods ask which comparisons should receive more weight. We ask a second question: what ordering does a particular placement of weights encourage? Assigning importance to comparisons does not by itself say which other orderings the objective favors, and when the directional signal is weak and a competing ordering is easier to learn, weighting can encourage that ordering even though the comparison labels never change. In our stock-ranking experiments, most published objectives we evaluate reduce ranking quality, and several produce negative rank correlation on more than one market. We hold data, labels, and architecture fixed and change only where the comparison weights are placed. We call this placement a pair-weight allocation. Allocations built from the same weight values produce markedly different rankings (Figure 1). Weight placement changes which errors receive emphasis and which competing ordering the objective rewards.

One competing ordering is return magnitude, the absolute distance from the daily mean return. Absolute returns show persistent patterns even though returns themselves are barely correlated over time (Cont, 2001), so movement size is likely easier to learn than direction. Ordering by magnitude is a shortcut in the sense of Geirhos et al. (2020): it groups large upward and downward deviations

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

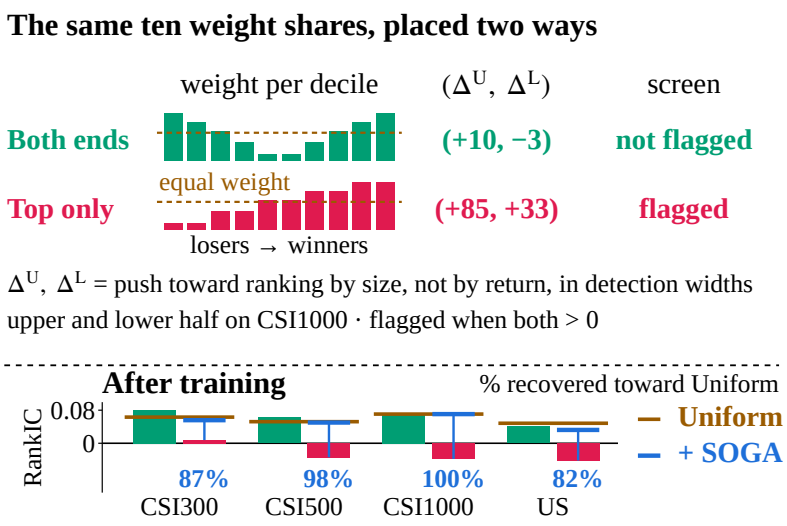

Figure 1: **Same weight values, different placement.** Top: the same ten weight shares per return decile placed on both return extremes (both-ends) or on the high-return ranks (top-only); each allocation's displacement from uniform weighting in the upper and lower halves of the ranking, in detection widths (Section 5), computed from labels and weights before training (CSI1000); and the screen's verdict. Bottom: after training, top-only falls below uniform weighting on every market and SOGA recovers 82 to 100% of the gap.

although they sit at opposite ends of the desired ranking. Magnitude can still separate returns on the same side of the daily mean; the failure occurs when it replaces learning the return order. Oracle diagnostics identify direction as the bottleneck, a model trained to rank magnitude reproduces the harmful allocation's ranking pattern, and restricting comparisons to stocks of similar magnitude turns an inverted ranking positive (Section 3 and Appendix D). These interventions point to a magnitude-related mechanism, and we address it by correcting the weights that encourage it rather than by discarding comparisons.

Based on the above analysis, if weight placement creates the harmful preference, it should be detectable before the prediction model is fit. At an equal-score state, all stocks are tied, so the scores carry no learned ordering, and we can ask whether an infinitesimal move toward magnitude order would lower the weighted loss. This directional response depends only on comparison weights and training labels, and comparing it with uniform weighting isolates the preference added by placement. Once identified, this excess can be reduced without discarding the allocation by changing the weights as little as possible while keeping their useful emphasis. To implement this, we propose Shortcut-Orthogonal Gradient Allocation (SOGA), a two-stage diagnose-then-correct framework. Before model fitting, SOGA measures the excess shortcut-aligned response relative to uniform and reads the two ranking halves separately to avoid cancellation. It flags allocations that add shortcut pressure in both halves. For each flagged allocation, it finds the closest corrected weight table satisfying the shortcut-response and coverage constraints under scale-invariant normalization. The original prediction model then trains normally using the fixed corrected weights. Across CSI300, CSI500, CSI1000, and US equities, every base weighting flagged by the diagnostic degrades ranking relative to uniform weighting. SOGA improves every corrected objective over its flagged base, with one corrected configuration exceeding all tested uncorrected objectives. It also outperforms uniform-shrinkage and correlation-penalty controls in matched comparisons, and the correction transfers across backbones.

Our contributions are threefold:

1. We characterize objective-induced competing-order preference as a failure mode of pairwise weighting and link its stock-ranking manifestation to a return-magnitude shortcut.

2. We show that objective-induced shortcut preference can be measured before model fitting through the equal-score directional response, and develop a half-wise diagnostic that reveals preference hidden by aggregate cancellation.
3. We introduce SOGA, a diagnose-then-correct algorithm that minimally adjusts flagged weight allocations subject to shortcut-response and coverage constraints. It improves every flagged objective tested, and one corrected configuration exceeds all tested uncorrected objectives across the four primary equity panels.

2 RELATED WORK

Ranking objectives and financial prediction. Stock models improve representations for relative-return prediction (Feng et al., 2019; Yang et al., 2020; Li et al., 2024), whereas our intervention changes training weights. Ranking objectives emphasize comparisons through pairwise and metric-sensitive losses (Burges et al., 2005; 2006; Burges, 2010; Wang et al., 2018), head prioritization (Rudin, 2009; Agarwal, 2011; Boyd et al., 2012; Lapin et al., 2016), or hard-negative mining (Schroff et al., 2015; Robinson et al., 2021). Weighted objectives have also been applied to return ranking (Kwiatkowski & Chudziak, 2025), and learning-to-rank losses to cross-sectional momentum (Poh et al., 2021). These methods choose which comparisons deserve emphasis, and ranking surrogates can be inconsistent for some ranking targets (Duchi et al., 2010; Calauzènes et al., 2012). We ask whether placing the same weights differently also changes the ordering the objective rewards. The question is specific to weighting rather than to pairwise training: the tested unweighted pointwise objectives are within one detection width (twice the standard error of the paired per-day difference, Section 5) of uniform pairwise training on the four equity panels, while carrying the same per-item weights into a squared-error objective reproduces most of the cost measured in the pairwise setting.

Importance weighting, shortcut learning, and feature competition. Byrd & Lipton (2019) find diminishing influence of importance weights in separable, overparameterized classification trained to convergence without regularization. The weights we study persist over the full training budget and reverse the ranking. Our setting differs in that rank-based weights can favor a second ordering available to the network, which offers a possible explanation for the persistent effect. Shortcut learning describes rules that bypass the intended solution (Geirhos et al., 2020), while gradient starvation explains how learning some features can suppress others (Pezeshki et al., 2021). Rather than treating reweighting only as a way to mitigate a shortcut already available in the data (Sagawa et al., 2020; Liu et al., 2021), we ask whether weighting itself adds a preference for that shortcut. Holding data, labels, and architecture fixed isolates this objective-level effect from changes in the shortcut's availability.

Learning and constraining training weights. Example reweighting can select weights using clean validation data (Ren et al., 2018; Shu et al., 2019), and distributionally robust methods reweight declared groups (Sagawa et al., 2020). SOGA instead constrains the added shortcut preference of an existing weighting using training labels. Its distinctive ingredient is the pre-training diagnostic and the resulting constraint, not projection itself.

3 PROBLEM SETUP AND KEY EMPIRICAL OBSERVATIONS

Cross-sectional stock ranking. On trading day t, stock i in universe $\mathcal{I}_t$ has features x_{ti} available at prediction time and a future-return label y_{ti}. The model's score is $s_{ti} = f_\theta(x_{ti})$. We evaluate its ranking using RankIC, the daily Spearman correlation between scores and labels, averaged over test days (Yang et al., 2020; Li et al., 2024). All training comparisons are within a day.

Pairwise weighting. Let $\mathcal{T}$ be the training days and $\mathcal{P}_t$ the admitted stock pairs on day t, with unequal labels and each unordered pair counted once. Write $d_{tij} = \mathrm{sign}(y_{ti} - y_{tj})$ and let $U_{tij} \geq 0$ be the comparison weight. Matching the daily normalization used in training, define $Z_t(\mathbf{U}) =$

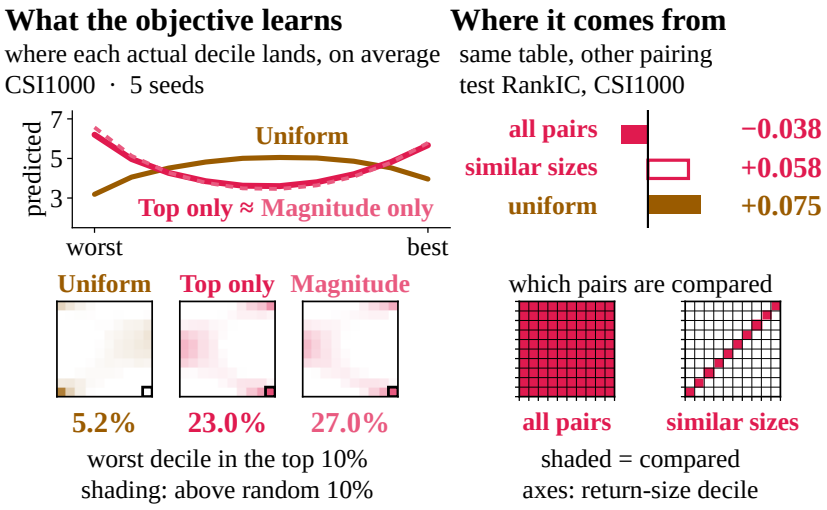

Figure 2: **What the flagged allocation learns, and where it comes from.** Left: share of the 10% largest losers on CSI1000 that each model places in its own top 10% of scores (chance 10%): 5.2% under uniform weighting, 23% under the flagged top-only allocation, and 27% for a model trained on magnitude alone. Right: training top-only only on pairs of similar-sized moves turns its rank correlation from -0.038 to $+0.058$; uniform weighting under the same restriction reaches $+0.075$.

$\sum_{(i,j)\in\mathcal{P}_t} U_{tij}$ and

$$\mathcal{L}_{\mathbf{U}}(\theta) = \frac{1}{|\mathcal{T}|} \sum_{t\in\mathcal{T}} \frac{\sum_{(i,j)\in\mathcal{P}_t} U_{tij}\, \ell(d_{tij}(s_{ti} - s_{tj}))}{Z_t(\mathbf{U})}, \tag{1}$$

where ℓ decreases as the correctly ordered margin grows. This normalization makes the objective invariant to a common positive rescaling of a day's weights. Weights leave the target order d_{tij} unchanged, but their placement can change the learned ranking. Figure 1 compares emphasis on both return extremes with emphasis on the high-return ranks using the same weight values; the bottom-only mirror image and a permutation control on a published label-distance rule are in Appendix E.

Magnitude-related ranking failure. Define the magnitude shortcut as $h_{ti} = |y_{ti} - \bar{y}_t|$, where $\bar{y}_t$ is the daily mean return. This section uses the constructed matched allocations because they isolate placement; the published objectives that the correction targets are evaluated in Section 5. Figure 2(a) shows the signature: of the 10% largest losers on CSI1000, the top-only model places 23% in its own top decile (chance 10%), close to a model trained to rank h alone (27%), whereas uniform weighting and the corrected table place about 5% there. Conversely, restricting training to pairs in the same daily magnitude decile changes the harmful allocation's RankIC on CSI1000 from -0.038 to $+0.058$ (Figure 2(b)). Both interventions point to a magnitude-related mechanism. SOGA addresses the preference by correcting the weights rather than by restricting pairs by magnitude, which would change the comparison set. Allocation construction, further diagnostics, and controls are in Appendices E and D.

4 SHORTCUT-ORTHOGONAL GRADIENT ALLOCATION

SOGA measures the ordering preference added by a pair-weight allocation, then corrects flagged tables under shortcut-response and extreme-region coverage constraints; return magnitude h is the shortcut studied here. "Orthogonal" refers to the excess equal-score score-gradient relative to uniform weighting, not to later parameter updates. Figure 3 summarizes the procedure.

4.1 SHORTCUT DIAGNOSIS

The diagnostic asks whether moving equal scores toward magnitude order lowers the normalized weighted loss. Let $\mathbf{U}_{\mathrm{ref}}$ assign weight one to every admitted pair and set diagnostic scores $s_{ti} =$

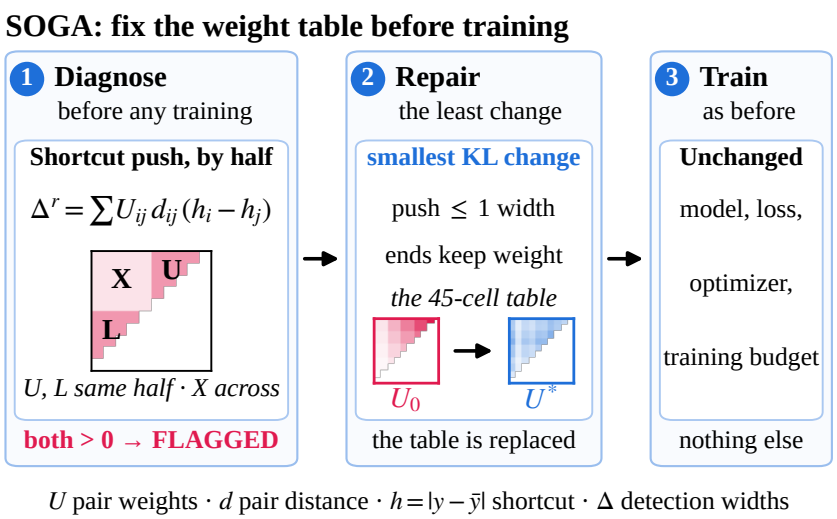

Figure 3: **SOGA in three steps.** *Diagnose*: before any model is fit, the training labels and the weight table $\mathbf{U}_0$ give the pushes toward magnitude order in the upper and lower halves, Δ^{U} and Δ^{L} (Equation 3); the table is flagged if both are positive. *Repair*: the closest table in the KL divergence D of Equation 4 with $|\Delta_h| \le \delta$ and both extreme regions covered (Equation 5); an unflagged table is kept. *Train*: only the weight table changes.

βh_{ti}. At $\beta = 0$, all margins are zero, independently of model predictions. For fixed weights and loss derivative $\ell'(0) < 0$ at zero margin, define the normalized shortcut response

$$A_h(\mathbf{U}) = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \frac{\sum_{(i,j) \in \mathcal{P}_t} U_{tij} d_{tij} (h_{ti} - h_{tj})}{Z_t(\mathbf{U})}, \qquad \left. \frac{\partial \mathcal{L}_{\mathbf{U}}(\beta h)}{\partial \beta} \right|_0 = \ell'(0)\, A_h(\mathbf{U}). \tag{2}$$

The identity holds for any declared coordinate v in place of h; which coordinate counts as a shortcut depends on the task, and our screening evidence concerns magnitude.

The response A_h measures normalized weighted agreement between return and magnitude order; since $\ell'(0) < 0$, a positive response means that moving toward magnitude order initially lowers the loss. With a zero daily mean, the return pair $(3, 1)$ contributes $+2$ and $(-1, -3)$ contributes -2: equal weights cancel them, whereas emphasizing the first pair rewards magnitude order without changing any label. Appendix A gives the derivation and numerical checks.

We measure the preference added by weight placement relative to uniform weighting rather than to zero, because the uniform response can itself be nonzero when returns are asymmetric. Define the displacement $\Delta_h(\mathbf{U}) = A_h(\mathbf{U}) - A_h(\mathbf{U}_{\mathrm{ref}})$. Because A_h uses the same weight normalization as the loss, Δ_h is invariant to common positive rescaling of the weights. Thus $\Delta_h > 0$ indicates added shortcut pressure from weight placement rather than scale. SOGA corrects this added preference.

Risk assessment across return ranks. The total displacement can hide opposing effects across the ranking. We separate pairs within the higher-return half ($\mathcal{P}_t^{\mathrm{U}}$), within the lower-return half ($\mathcal{P}_t^{\mathrm{L}}$), and across the halves ($\mathcal{P}_t^{\mathrm{X}}$). For group r, let A^r be its contribution to the normalized response and Δ^r its change from uniform weighting, using the same per-day denominator $Z_t(\mathbf{U})$ as Equation 1:

$$A^r(\mathbf{U}) = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \frac{\sum_{(i,j) \in \mathcal{P}_t^r} U_{tij} d_{tij} (h_{ti} - h_{tj})}{Z_t(\mathbf{U})}, \quad \Delta^r(\mathbf{U}) = A^r(\mathbf{U}) - A^r(\mathbf{U}_{\mathrm{ref}}), \quad r \in \{\mathrm{U}, \mathrm{L}, \mathrm{X}\}. \tag{3}$$

These contributions sum to $\Delta_h = \Delta^{\mathrm{U}} + \Delta^{\mathrm{L}} + \Delta^{\mathrm{X}}$. We flag an allocation when $\Delta^{\mathrm{U}} > 0$ and $\Delta^{\mathrm{L}} > 0$: both halves add shortcut pressure. Top-only has this pattern. Bottom-only has two negative displacements, and both-ends has opposite signs. The screen is used as a one-sided risk indicator for degradation relative to uniform weighting; it does not imply negative RankIC, and unflagged allocations can also be harmful (Section 5.4). SOGA corrects the flagged ones.

Algorithm 1 SOGA: diagnose the weight table, correct it if flagged, then train as before

Require: training labels, base weighting rule, tolerance δ, coverage floor ϵ

1: build the normalized base table $\mathbf{U}_0$ and the uniform reference $\mathbf{U}_{\mathrm{ref}}$ from the training labels
2: compute $\Delta^{\mathrm{U}}(\mathbf{U}_0)$ and $\Delta^{\mathrm{L}}(\mathbf{U}_0)$ (Equation 3)
3: **if** $\Delta^{\mathrm{U}}(\mathbf{U}_0) > 0$ and $\Delta^{\mathrm{L}}(\mathbf{U}_0) > 0$ **then** solve Equation 5 for $\mathbf{U}^*$ and verify its constraints; **else** $\mathbf{U}^* \leftarrow \mathbf{U}_0$
4: train the original ranking model with the fixed weights $\mathbf{U}^*$

4.2 WEIGHT CORRECTION

SOGA corrects a compact table of rank-region pairs instead of every pair weight individually. Divide each day's return ranks into $Q = 10$ regions; entry $U(a, b)$ weights comparisons between regions $1 \le a < b \le Q$, and same-region pairs are excluded, giving at most 45 cells $\mathcal{B}$. The base table $\mathbf{U}_0$ is normalized to mean pair weight one over the cells.

We keep the table close to its base by minimizing the Kullback–Leibler divergence between the normalized tables,

$$D(\mathbf{U}\|\mathbf{U}_0) = \sum_{(a,b)\in\mathcal{B}} \bar{U}(a,b) \log \frac{\bar{U}(a,b)}{\bar{U}_0(a,b)}, \qquad \bar{U}(a,b) = U(a,b) \Big/ \textstyle\sum_{\mathcal{B}} U, \tag{4}$$

so each cell counts equally and the divergence is unchanged if a table is rescaled. Let C_q denote region q's normalized share of incident comparison weight. With tolerance $\delta \ge 0$ (in the units of Δ_h, calibrated in Appendix C) and floor $\epsilon > 0$, solve

$$\mathbf{U}^* = \arg\min_{\mathbf{U}\ge 0} D(\mathbf{U}\|\mathbf{U}_0) \;\text{ s.t. }\; |\Delta_h(\mathbf{U})| \le \delta,\; \min\{C_1, C_Q\} \ge \epsilon, \tag{5}$$

and the solution is reported rescaled to mean pair weight one over the cells. The two constraints bound the added shortcut response relative to uniform and keep both extreme return regions represented. Scale is a normalization rather than a constraint: Δ_h, C_q, and D are invariant to a common positive rescaling of $\mathbf{U}$, so the band cannot be met by shrinking all weights. The floor prevents emptying either extreme, which is where a harmful allocation concentrates its weight. Minimizing the divergence retains as much of the original emphasis as the constraints allow, and a feasible base is returned unchanged. Appendix C gives the calibration of δ and ϵ, the table aggregation, coverage construction, numerical solver, and feasibility details.

Diagnosis reads the two halves separately because the total can hide cancellation, whereas correction constrains the total Δ_h, crossing pairs included, which leaves more freedom to keep the base weighting; the half-wise quantities are a risk screen, and the correction controls the aggregate excess without enforcing the screening condition itself, so a corrected table need not become unflagged. Algorithm 1 gives the steps.

Training and scope. Prediction-dependent weights are frozen at equal scores before diagnosis, and test-time prediction needs no shortcut labels. At $\delta = 0$ the guarantee is orthogonality of the excess equal-score score-gradient relative to uniform weighting, not of either gradient alone or of later parameter updates (Appendices A and B).

5 EXPERIMENTS

Setup. The primary panels are CSI300, CSI500, CSI1000, and US, with four-day returns on CSI and ten-day returns on US; the US panel holds the $1{,}499$ most traded names outside the S&P 500 and Nasdaq-100 (Appendix D). We use MASTER (Li et al., 2024) ($775{,}041$ parameters), trained for 40 epochs without early stopping, matching architecture, optimizer, and budget across five paired seeds; the objectives compared are listed in Table 1, where MSE is the backbone's published mean-squared-error objective. Single-model mean daily RankIC is scored per seed and averaged, and all single-model comparisons are of these means. The detection width θ is twice the standard error of paired per-day differences. Appendix D gives dataset details, uncertainty estimators, and backbone configurations.

Table 1: **Primary single-model comparison on four equity panels.** Mean daily RankIC, five paired seeds; bold marks the best mean per column, underlining the best non-SOGA objective. Each correction gains 3.5 to 7.7 paired detection widths over its own uncorrected row (widths in Appendix H); for LambdaRank the gain includes freezing (Section 5.2).

	CSI300	CSI500	CSI1000	US
Published/reference objectives				
MSE (Li et al., 2024)	$\underline{+0.0685}$	$+0.0484$	$\underline{+0.0711}$	$+0.0438$
Uniform pairwise ($\mathbf{U}_{\mathrm{ref}}$)	$+0.0632$	$\underline{+0.0522}$	$+0.0706$	$\underline{+0.0484}$
LambdaRank (Burges et al., 2006)	$+0.0008$	-0.0221	-0.0196	-0.0351
Weighted hinge v1 (Kwiatkowski & Chudziak, 2025)	$+0.0084$	-0.0308	-0.0346	-0.0417
Weighted hinge v2 (Kwiatkowski & Chudziak, 2025)	$+0.0149$	-0.0232	-0.0175	-0.0375
Softplus hinge (Kwiatkowski & Chudziak, 2025)	-0.0044	-0.0368	-0.0401	-0.0433
ListNet (Cao et al., 2007)	-0.0095	-0.0319	-0.0328	-0.0421
Top-k hinge (k=10%)	$+0.0113$	$+0.0170$	$+0.0247$	$+0.0107$
SOGA-*corrected objectives*				
LambdaRank (frozen) + SOGA	$\mathbf{+0.0773}$	$\mathbf{+0.0589}$	$+0.0741$	$+0.0293$
Weighted hinge v1 + SOGA	$+0.0731$	$+0.0545$	$\mathbf{+0.0765}$	$\mathbf{+0.0538}$
Weighted hinge v2 + SOGA	$+0.0758$	$+0.0568$	$\mathbf{+0.0765}$	$+0.0441$
Softplus hinge + SOGA	$+0.0571$	$+0.0506$	$+0.0707$	$+0.0333$

5.1 MAIN SINGLE-MODEL RESULTS ON EQUITY PANELS

One fixed configuration, weighted hinge v1 + SOGA, exceeds every tested uncorrected published/reference objective on all four primary equity panels (Table 1). A SOGA objective also has the highest mean RankIC in each panel, showing that correction does more than repair inversions.

Failures cut across objective families: nine of fifteen objectives degrade ranking quality with architecture and inputs fixed, including weighted pairwise losses and ListNet, and LambdaRank adds more shortcut preference in both halves than the constructed harmful allocation and ends with negative rank correlation on three equity panels. Weight placement, not the pairwise or listwise form, is the issue (Figure 1); Appendix E lists all objectives and controls.

5.2 SOGA REPAIRS HARMFUL COMPARISON WEIGHTING

Because LambdaRank updates its weights from the current ranking, we freeze the rule at equal scores before correcting its table, separating the two interventions. Freezing alone recovers 1% to 18% of the damage; correction adds 3.6 to 6.1 detection widths on all four equity panels with every seed agreeing and yields positive mean rank correlation on every market. The one shortfall is US, where corrected frozen LambdaRank remains 0.019 below $\mathbf{U}_{\mathrm{ref}}$. The three published weighted hinge variants behave similarly: all fifteen market–variant cells are flagged, and correction recovers 3.5 to 7.7 widths, most on US where the base is worst. Weighted hinge v1 and softplus hinge share a corrected table but retain different penalties and trained results.

Why not just use uniform weighting? Uniform weighting is a sound default, but Table 1 shows that a corrected objective has a higher mean on every market. Equation 5 retains as much original comparison emphasis as the constraints allow instead of pulling every allocation toward uniform. Across five bases and four markets, the projection moves the mildly displaced label-distance weighting by less than 0.3% of the KL distance to the uniform reference and the constructed harmful allocation by up to 77%; uniform shrinkage, on the twelve base–market cells where it was run, moves 88 to 97% of the way (Appendix H). The trained RankIC of label-distance is indistinguishable from its uncorrected value on the four equity panels. The correction therefore acts strongly on harmful allocations while leaving mildly displaced ones almost intact (Figure 4). These projection ablations include unflagged bases, which Algorithm 1 leaves untouched (full analysis in Appendix H).

5.3 TRANSFER ACROSS BACKBONES

On CSI1000, correction raises mean RankIC on every reported backbone, from a 444-parameter linear model to a $775{,}041$-parameter transformer (Table 2). Every backbone uses the same flagged

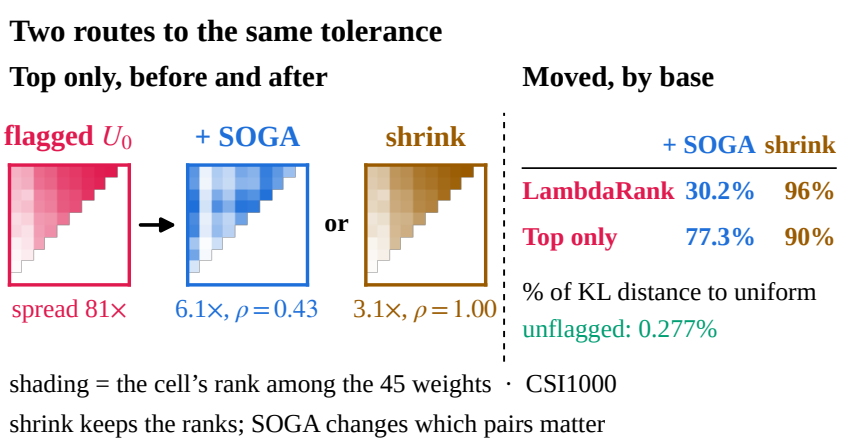

Figure 4: **Two routes to the same tolerance, CSI1000.** Left: the flagged top-only table before correction, after SOGA, and after uniform shrinkage stopped at the same tolerance; shading is each cell's rank among the 45 weights, and the annotations give the table's dynamic range and its rank correlation ρ with the base. Shrinkage keeps the ranks and loses the spread; SOGA changes which pairs matter. Right: share of the KL distance to the uniform table travelled, $1 - D(\mathbf{U}^*\|\mathbf{U}_{\text{ref}})/D(\mathbf{U}_0\|\mathbf{U}_{\text{ref}})$, by SOGA and by shrinkage for the two flagged bases; the unflagged label-distance base, which Algorithm 1 would leave untouched, moves 0.28%.

Table 2: **Architecture-wise repair, CSI1000.** Mean daily RankIC, five seeds; flagged base: weighted hinge v1, same loss in both arms. Recovery = (SOGA − Flagged)/(Uniform − Flagged).

Backbone	Params	MSE	Uniform	Flagged	+ SOGA	Recovery
Linear	444	+0.0469	+0.0557	−0.0445	+0.0608	105%
FITS	1,801	+0.0036	+0.0557	−0.0407	+0.0624	107%
GRU	42,881	+0.0437	+0.0660	−0.0168	+0.0532	85%
ALSTM	66,369	+0.0354	+0.0525	−0.0103	+0.0369	75%
TimeMixer	95,686	+0.0356	+0.0671	−0.0110	+0.0548	84%
LSTM	106,817	+0.0329	+0.0406	−0.0021	+0.0311	78%
TimesNet	349,555	+0.0540	+0.0620	−0.0174	+0.0566	93%
MASTER	775,041	+0.0711	+0.0706	−0.0346	+0.0765	106%

base, the published weighted hinge v1, so the repair is compared under one objective: the flagged objective ends with negative rank correlation on all eight backbones, and the correction closes 75% to 107% of the gap to each backbone's uniform reference, ending above uniform on three backbones and below on five. The Transformer is omitted because its reference is indistinguishable from zero. Since the correction table depends only on the base, the labels, and the bounds, one table serves every backbone; the same study with frozen LambdaRank as the base gives the same picture (Appendix I), and backbone configurations are in Appendix D.

5.4 The Pre-Training Diagnostic as a Risk Screen

All 22 flagged base allocations in the 96-cell evaluation, including five on US, degrade relative to uniform weighting, with every seed agreeing. The broader collection also includes corrected tables, ablations, and controls, on which the flag is less reliable: precision and recall are 90.0% and 73.8% overall and 82.8% and 66.7% on the 56 cells not used to formulate the rule. The prospective check is the cryptocurrency study, built after the criterion and its thresholds were fixed (Appendix D).

Half-wise diagnosis and granularity. The half-wise criterion agrees with trained outcomes on 78.1% of cells, against 67.7% for the sign of the total displacement, and it is right on 22 of the 34 cells where the two disagree; separating the halves exposes shortcut pressure that cancels in the total. On the 33 tables that can be rebuilt at $Q = 5$, 10 and 20, the flagged set is identical, so the diagnostic does not depend on $Q = 10$ (Appendix F).

Table 3: **SOGA versus simpler shortcut corrections.** Paired Δ/θ against each alternative on the same base (frozen LambdaRank for shrinkage, top-only otherwise); positive favors SOGA.

Alternative	CSI300	CSI500	CSI1000	US
Uniform shrinkage	$+2.01$	$+3.47$	$+3.18$	$+4.10$
Correlation penalty	$+3.13$	$+4.79$	$+5.90$	$+5.43$
Gradient projection (zero)	$+2.25$	$+0.37$	$+2.57$	-1.77
Gradient projection (reference)	$+3.43$	$+1.89$	$+3.40$	-1.34

Connection to early optimization. Over the first twenty updates, all eight arm–market cells displaced by more than 120 widths accumulate score changes in the predicted shortcut direction on every seed (rank correlation $+0.83$ between displacement and accumulated projection across twelve cells), while the four cases at 7 to 27 widths move the other way: the diagnostic tracks early optimization where the displacement is large and remains a risk indicator, not a guarantee (Appendix A).

5.5 Alternative Corrections and Selectivity

Why correct the weights? Three controls remove the same preference at different levels: uniform shrinkage moves the table along the geometric path $\mathbf{U}_0^t$ toward uniform weighting, the correlation penalty suppresses shortcut dependence in the model's outputs, and gradient projection removes the score-gradient component along the shortcut at every step, either entirely or down to its value under uniform weighting.

SOGA outperforms uniform shrinkage and the correlation penalty on all four equity panels in the matched comparisons (Table 3), so constrained weight correction does better than either uniform interpolation or an output penalty. Gradient projection comes closest: SOGA is ahead of both variants on CSI300 and CSI1000, ahead of the reference-target variant and within one detection width of the zero-target variant on CSI500, and behind both on US.

The two differ in where the constraint lives: SOGA changes the table once and needs no shortcut values afterwards, whereas the gradient and output controls need them at every step, and for gradient projection the Jacobian transforms the constrained gradient at each update (Appendix A). Full results are in Table 7; sensitivity and boundary cases in Appendix H.

Additional evidence. A cryptocurrency market built after the diagnostic was fixed reproduces the failure and the repair (Appendix D); a rank-average ensemble of the weightings the diagnostic admits is above uniform weighting on all five markets, whereas the same-size ensemble with two flagged members falls below (Appendix I); and most of the head-metric (NDCG) gain of a harmful weighting is reproduced by a model that ranks magnitude alone (Appendix G).

6 Limitations and Conclusion

The equal-score diagnostic characterizes the incentive at the start of training, not every later effect of weights that evolve during it. The three CSI panels share a market and a calendar, the backbone study uses CSI1000 alone, and the US constituent list has the limitations stated in Appendix D; on US, corrected frozen LambdaRank stays below uniform weighting and neither arm settles within the budget. The screened ensembles of Appendix I extend the diagnostic beyond repairing a single objective.

Comparison weighting can induce a preference for a competing ordering, and in stock ranking that ordering is return magnitude. The equal-score response indicates this preference before training, and constraining flagged allocations removes most of the downstream failure: one fixed corrected objective exceeds every tested uncorrected objective across the four primary equity panels, although individual corrected objectives can still fall below uniform weighting on some panels.

REPRODUCIBILITY STATEMENT

Appendix D gives the panels, dates, label formula, backbone configurations, and the uncertainty estimators; Appendix C gives the construction of the rank-region table, the calibration of δ and ϵ, and the solver; Appendix E gives the construction of every allocation and published objective evaluated; and Appendices A and F give the derivation of the diagnostic and the bookkeeping behind its validation. Every table is generated from stored per-seed outputs by a script, and the sign predictions of Section 5.4 were written to a file frozen before the corresponding arms were trained. Code, the frozen prediction file, and the constituent lists will be released upon publication.

ETHICS STATEMENT

This work uses only market price and volume data that are publicly available or obtainable from standard data vendors, involves no human subjects and no personal data, and does not produce trading recommendations; the reported rank correlations and spreads are ordering diagnostics on held-out days, not executable returns. The constituent-selection limitations of the US panel are stated in Appendix D.

AI USE STATEMENT

We used generative AI tools to assist with writing (restructuring, condensing, and polishing the authors' drafts), to locate related work for verification, and to write the scripts that aggregate stored experimental outputs into the reported tables and figures. The research question, method, and experimental design are the authors' own, and all AI-assisted text and scripts were checked by the authors, who take full responsibility for the content.

REFERENCES

Shivani Agarwal. The infinite push: A new support vector ranking algorithm that directly optimizes accuracy at the absolute top of the list. In *Proceedings of the 2011 SIAM International Conference on Data Mining (SDM)*, pp. 839–850, 2011.

Andrew Ang, Robert J. Hodrick, Yuhang Xing, and Xiaoyan Zhang. The cross-section of volatility and expected returns. *The Journal of Finance*, 61(1):259–299, 2006.

Turan G. Bali, Nusret Cakici, and Robert F. Whitelaw. Maxing out: Stocks as lotteries and the cross-section of expected returns. *Journal of Financial Economics*, 99(2):427–446, 2011.

Xavier Bouthillier, Pierre Delaunay, Mirko Bronzi, Assya Trofimov, Brennan Nichyporuk, Justin Szeto, Nazanin Mohammadi Sepahvand, Edward Raff, Kanika Madan, Vikram Voleti, Samira Ebrahimi Kahou, Vincent Michalski, Tal Arbel, Chris Pal, Gaël Varoquaux, and Pascal Vincent. Accounting for variance in machine learning benchmarks. In *Proceedings of Machine Learning and Systems (MLSys)*, volume 3, 2021.

Stephen Boyd, Corinna Cortes, Mehryar Mohri, and Ana Radovanovic. Accuracy at the top. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 25, pp. 953–961, 2012.

Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pp. 89–96, 2005.

Christopher J. C. Burges. From RankNet to LambdaRank to LambdaMART: An overview. Technical Report MSR-TR-2010-82, Microsoft Research, 2010.

Christopher J. C. Burges, Robert Ragno, and Quoc Viet Le. Learning to rank with nonsmooth cost functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 19, pp. 193–200, 2006.

Jonathon Byrd and Zachary Lipton. What is the effect of importance weighting in deep learning? In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 872–881, 2019.

Clément Calauzènes, Nicolas Usunier, and Patrick Gallinari. On the (non-)existence of convex, calibrated surrogate losses for ranking. In *Advances in Neural Information Processing Systems*, volume 25, 2012.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: From pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning (ICML)*, pp. 129–136, 2007.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 1724–1734, 2014.

Rama Cont. Empirical properties of asset returns: Stylized facts and statistical issues. *Quantitative Finance*, 1(2):223–236, 2001. doi: 10.1080/713665670.

John C. Duchi, Lester W. Mackey, and Michael I. Jordan. On the consistency of ranking algorithms. In *Proceedings of the 27th International Conference on Machine Learning*, pp. 327–334, 2010.

Fuli Feng, Xiangnan He, Xiang Wang, Cheng Luo, Yiqun Liu, and Tat-Seng Chua. Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems (TOIS)*, 37(2): 1–30, 2019.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8): 1735–1780, 1997.

Jan Kwiatkowski and Jarosław A. Chudziak. On evaluating loss functions for stock ranking: An empirical analysis with transformer model. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 4910–4914, 2025. doi: 10.1145/3746252.3760812.

Maksim Lapin, Matthias Hein, and Bernt Schiele. Loss functions for top-k error: Analysis and insights. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1468–1477, 2016.

Tong Li, Zhaoyang Liu, Yanyan Shen, Xue Wang, Haokun Chen, and Sen Huang. MASTER: Market-guided stock transformer for stock price forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 162–170, 2024.

Evan Z. Liu, Behzad Haghgoo, Annie S. Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just train twice: Improving group robustness without training group information. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 6781–6792, 2021.

Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

Daniel Poh, Bryan Lim, Stefan Zohren, and Stephen Roberts. Building cross-sectional systematic strategies by learning to rank. *The Journal of Financial Data Science*, 3(2):70–86, 2021.

Dimitris N. Politis and Joseph P. Romano. A circular block-resampling procedure for stationary data. In Raoul LePage and Lynne Billard (eds.), *Exploring the Limits of Bootstrap*, pp. 263–270. Wiley, New York, 1992.

Yao Qin, Dongjin Song, Haifeng Chen, Wei Cheng, Guofei Jiang, and Garrison W. Cottrell. A dual-stage attention-based recurrent neural network for time series prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 2627–2633, 2017.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 4334–4343, 2018.

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. In *International Conference on Learning Representations (ICLR)*, 2021.

Cynthia Rudin. The P-norm push: A simple convex ranking algorithm that concentrates at the top of the list. *Journal of Machine Learning Research*, 10:2233–2271, 2009.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, 2020.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, 2015.

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y. Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations*, 2024.

Xuanhui Wang, Cheng Li, Nadav Golbandi, Michael Bendersky, and Marc Najork. The LambdaLoss framework for ranking metric optimization. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 1313–1322, 2018.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: Theory and algorithm. In *Proceedings of the 25th International Conference on Machine Learning (ICML)*, pp. 1192–1199, 2008.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling time series with $10k$ parameters. In *International Conference on Learning Representations*, 2024.

Xiao Yang, Weiqing Liu, Dong Zhou, Jiang Bian, and Tie-Yan Liu. Qlib: An AI-oriented quantitative investment platform. *arXiv preprint arXiv:2009.11189*, 2020.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11121–11128, 2023.

APPENDIX CONTENTS

A Derivation, rank invariance, and the guarantee boundary
B First-order account of the early ordering, and its limits
C Table construction and implementation
D Markets, backbones, uncertainty, oracles, and trajectories
E Allocations, published objectives, and controls
F Criterion validation on trained cells
G Spread and head metrics
H Alternative corrections, sensitivity, and boundary cases
I Ensemble definitions, leave-one-out, and selection across halves

A DERIVATION, RANK INVARIANCE, AND THE GUARANTEE BOUNDARY

Equal-score identity. For fixed weights and a loss differentiable at zero, differentiating the day-normalized objective in Equation 1 gives $\partial \mathcal{L}_{\mathbf{U}}(\beta h)/\partial\beta = |\mathcal{T}|^{-1}\sum_t Z_t(\mathbf{U})^{-1}\sum_{(i,j)\in\mathcal{P}_t} U_{tij}\,\ell'(\beta d_{tij}(h_{ti}-h_{tj}))\,d_{tij}(h_{ti}-h_{tj})$. At $\beta = 0$ every ℓ' equals $\ell'(0)$, which gives Equation 2. For the softplus margin loss used in these numerical checks, $\ell'(0) = -1/2$. Evaluating the left-hand side by central differences of the actual normalized pairwise loss at $\beta = \pm 10^{-4}$ on eight allocation–market cells reproduces $\ell'(0)A_h(\mathbf{U})$ to a relative error of 3×10^{-5}, and the decomposition $\Delta_h = \Delta^{\mathrm{U}} + \Delta^{\mathrm{L}} + \Delta^{\mathrm{X}}$ holds to 10^{-14} on every table reported. A tolerance $|\Delta_h| \le \delta$ therefore bounds the departure of the equal-score diagnostic derivative from its reference value by $|\ell'(0)|\delta$.

Rank invariance and the choice of reference. A pairwise rank loss with rank-based weights depends on the labels only through their order. Replacing y by its within-day rank-Gaussian transform $y' = \Phi^{-1}((r-\frac{1}{2})/n)$ leaves the training of every rank-weighted arm bit-identical (loss difference 0.0 on six allocations), whereas the shortcut $h' = |y' - \bar{y}'|$ changes: y' is antisymmetric within the day, the region pairs mirror exactly (mirror mismatch below 2.1×10^{-15}), and $A_{h'}(\mathbf{U}_{\mathrm{ref}})$ becomes 1.3×10^{-16} with a daily series of zero variance. The offset of $A_h(\mathbf{U}_{\mathrm{ref}})$ under the actual labels, 12 to 22 widths on the CSI markets, 4 on US and 26 on cryptocurrency, is therefore the asymmetry of the label distribution, a quantity the rank-based loss never uses, and uniform weighting is the natural zero of displacement.

The direction coordinate cannot be reversed. With $b_{ti} = \mathrm{sign}(y_{ti} - \bar{y}_t)$, the label order implies the sign order, so $d_{tij}(b_{ti} - b_{tj}) \ge 0$ for every pair and $A_b(\mathbf{U}) \ge 0$ for any nonnegative allocation. In our measurements, the top-only allocation multiplies the shortcut component by 10.58, 9.16, and 8.27 on the CSI markets and the direction component by 0.88, 0.91, and 0.92; these are objective-level incentive ratios, and the nonnegativity of A_b does not by itself guarantee that the learned scores preserve direction.

The sense of orthogonality, and how far it holds. For the normalized loss at equal scores, the shortcut band bounds the departure of the directional derivative from its reference value by $|\ell'(0)|\delta$, and at $\delta = 0$ the difference between the weighted and the reference score gradients is orthogonal to h. This is the sense of "orthogonal" in SOGA: it is a statement about score gradients at equal scores, not about either gradient alone. With $g = \nabla_s \mathcal{L}$ and $J = \partial s/\partial\theta$, one gradient step gives $\Delta s = -\eta J J^\top g + O(\eta^2)$, and $h^\top g = 0$ does not imply $h^\top J J^\top g = 0$, so orthogonality of the update does not follow. We measured the first twenty updates of every arm, projecting each score change onto the batch-centered shortcut direction and comparing the accumulated projection with a resampled noise floor (twice the resampled standard deviation; the floor admits a pure-noise sample 4.3% of the time). On the eight arm–market cells displaced by more than 120 widths, the accumulated projection carries the sign of the displacement in every case and on every seed, six to forty-eight times the floor; the rank correlation between displacement and projection over twelve scored cells is $+0.83$. On the four cells displaced by 7 to 27 widths it carries the opposite sign. The diagnostic therefore tracks early score changes where the displacement is large and can fail where it is small; it is not a general correspondence between flagging and update direction.

B First-Order Account of the Early Ordering, and Its Limits

For fixed pair weights and a scorer smooth near the equal-score reference, concatenate stock-days and define the normalized pair pull $g_{\mathbf{U}} = |\mathcal{T}|^{-1} \sum_t Z_t(\mathbf{U})^{-1} \sum_{(i,j)\in\mathcal{P}_t} U_{tij} d_{tij}(e_{ti} - e_{tj})$, so that $A_v(\mathbf{U}) = v^\top g_{\mathbf{U}}$ for any concatenated per-stock quantity v.

Lemma 1 (First update of any scorer)**.** *At equal scores, one gradient step with rate η on Equation 1 changes the scores by $\Delta s = -\eta\, \ell'(0)\, J J^\top g_{\mathbf{U}} + O(\eta^2)$.*

Lemma 2 (Early trajectory of a linear scorer)**.** *For $s = X\theta$ under gradient flow from $\theta = 0$, $s(t) = -t\, \ell'(0)\, X X^\top g_{\mathbf{U}} + O(t^2)$; the early ordering is $s \propto \sum_k A_{x_k}(\mathbf{U})\, x_k$, the table's normalized responses against each feature.*

Proof. Both are first-order expansions: at equal scores every ℓ' equals $\ell'(0)$, so $\nabla_s \mathcal{L}_{\mathbf{U}} = \ell'(0) g_{\mathbf{U}}$, and one gradient step, or the first-order term of gradient flow, is $J^\top$ of that gradient mapped back through J. □

Both lemmas are elementary and recorded for reference, not as new results. The second lemma is the placement statement in one line: tables with the same values placed differently can have different $g_{\mathbf{U}}$ and different first-order score changes, which the weight histogram alone does not determine; distinct tables need not always produce distinct score changes.

Where the first-order account stops. Writing $g_{\mathbf{U}} = c_b \tilde{b} + c_h \tilde{h} + r$ with $\tilde{b}, \tilde{h}$ the centered direction and residualized magnitude coordinates, one might expect $\mathrm{corr}(s(t), y) \approx \kappa_b c_b \|\tilde{b}\| \rho_b + \kappa_h c_h \|\tilde{h}\| \rho_h$ with κ the kernel's Rayleigh quotients. The first-order ordering $s_0 = X X^\top g_{\mathbf{U}}$ is positively correlated with the label on all twenty allocation–market cells ($+0.161$ to $+0.398$), an empirical observation for these feature representations, not a guarantee for arbitrary linear smoothing; the kernel ratios κ_h/κ_b (0.74, 1.03, 1.27, 2.62 on CSI300, CSI500, CSI1000, US, computed on the flattened eight-day window of 158 stock and 63 market features) are far from what the second term would need. The observed inversion therefore emerges during training rather than in this first-order ordering; feature competition (Pezeshki et al., 2021) is a plausible mechanism that this calculation does not establish. The diagnostic measures an initial incentive, whose relation to early updates is measured in Appendix A. A controlled generator with heavy-tailed magnitude, a direction proxy of adjustable AUC and allocations interpolating uniform to top-only reproduces the monotone decay of rank correlation with displacement on 396 cells but not the crossing of zero, so the markets contain an ingredient that construction lacks.

Why placement matters: a location–scale account. The first-order calculation does not say which ordering a weighting rewards once the scores are free to move; a population version of the pair response does. Fix a day and two stocks with independent labels $y_i = \mu_i + \sigma_i \varepsilon_i$ and $y_j = \mu_j + \sigma_j \varepsilon_j$, where $\sigma_i, \sigma_j > 0$ and the noise ε is continuous, symmetric about zero, with positive density, survival function S and hazard rate λ such that $x\lambda(x)$ is strictly increasing on $x \geq 0$ (Gaussian, logistic and Laplace noise qualify). For a symmetric pair weight $w(y_i, y_j) \geq 0$, write $\pi_w(i \succ j) = \mathbb{E}[w(y_i, y_j)\, \mathbf{1}\{y_i > y_j\}]$ and $P_w(i,j) = \pi_w(i \succ j) - \pi_w(j \succ i) = \mathbb{E}[w\, d_{ij}]$. At equal scores the expected gradient of the pair term $w\, \ell(d_{ij}(s_i - s_j))$ with respect to $s_i - s_j$ is $\ell'(0) P_w(i,j)$, and since $\ell'(0) < 0$ gradient descent moves $s_i - s_j$ in the direction of $P_w(i,j)$.

Proposition 1. *(a) Under uniform weights, $w \equiv 1$, $\operatorname{sign} P_w(i,j) = \operatorname{sign}(\mu_i - \mu_j)$ for all σ_i, σ_j. (b) Under upper-region weights, $w = \mathbf{1}\{\max(y_i, y_j) > c\}$ with $\mu_i = \mu_j = \mu \leq c$, $\sigma_i > \sigma_j$ implies $P_w(i,j) > 0$; by continuity $P_w(i,j) > 0$ persists when μ_i is slightly below μ_j. (c) The mirror weights $\mathbf{1}\{\min(y_i, y_j) < 2\mu - c\}$ give $-P_w(i,j)$, so their sum with the weights of (b) is scale-blind at equal means.*

Proof. (a) Here $y_i - y_j = (\mu_i - \mu_j) + \eta$ with $\eta = \sigma_i \varepsilon_i - \sigma_j \varepsilon_j$ continuous, symmetric about zero and with positive density, so $P_w(i,j) = 2\Pr(\eta > \mu_j - \mu_i) - 1$ has the sign of $\mu_i - \mu_j$. (b) Take $\mu = 0$ without loss of generality and let $B = \{y_i > c,\ y_j > c\}$. Since $w = 1$ exactly when the larger label exceeds c,

$$P_w(i,j) = \big[\Pr(y_i > c) - \Pr(y_j > c)\big] + \Pr(B)\big[\Pr(y_i > y_j \mid B) - \Pr(y_j > y_i \mid B)\big].$$

The first bracket is nonnegative because $c \geq 0$ and $\sigma_i > \sigma_j$. On B the labels are independent with survival functions $S(t/\sigma)/S(c/\sigma)$ for $t \geq c$, and $\partial_\sigma \log[S(t/\sigma)/S(c/\sigma)] = [(t/\sigma)\lambda(t/\sigma) - (c/\sigma)\lambda(c/\sigma)]/\sigma > 0$ for $t > c$. The label of i therefore strictly dominates that of j on B, so $\Pr(y_i > y_j \mid B) > \frac{1}{2}$ and the second term is positive. (c) The reflection $y \mapsto 2\mu - y$ preserves the joint law of (y_i, y_j) at equal means, reverses d_{ij}, and maps $\{\max(y_i, y_j) > c\}$ to $\{\min(y_i, y_j) < 2\mu - c\}$. □

The half-wise screen of Section 4.1 is the sample counterpart of (b): moving normalized weight from lower-half to upper-half pairs raises Δ^{U} and makes the negative lower-half contribution smaller, raising Δ^{L}, while both-ends weighting moves the two halves in opposite directions as in (c). Uniform weighting is blind to scale, whereas upper-region weighting rewards it even at equal expected returns. When expected return falls with scale across stocks, as in the low-volatility effect (Ang et al., 2006; Bali et al., 2011), this scale preference orders stocks against their expected return, which is one ingredient that can turn the monotone decay of the controlled generator into the crossing of zero observed in the markets; we do not test this account here.

C Table Construction and Implementation

Regions and cells. Region boundaries are the within-day label deciles, with ties at a boundary broken by stock identifier so that membership is deterministic. Let $\mathcal{P}_{t,ab}$ contain the unequal-label pairs on day t whose region indices are $a < b$; every admitted pair belongs to exactly one cell, and same-region pairs are not admitted, so at $Q = 10$ there are 45 cells. Weight lookup is symmetric and only the $a < b$ entry is stored. The primary diagnostic evaluates $A_h(\mathbf{U})$ day by day using Equation 2, matching the normalization of the training loss. For the published hinge-table solvers and cryptocurrency, the optimizer uses the pooled-ratio analogue $A_h^{\mathrm{pool}}(\mathbf{U}) = \{\sum_{t,(i,j)} U_{tij} d_{tij}(h_{ti} - h_{tj})\}/\{\sum_{t,(i,j)} U_{tij}\}$; the numerical difference from the day-averaged convention on reported solutions is quantified below. Training draws one day per batch and normalizes each day's loss by its total pair weight.

Constructing a region-shared base. If the source rule already assigns a common weight to each region pair, that allocation is used. For a finer-grained rule w^0_{tij}, the cell weight is the count-weighted average $\widetilde{U}_0(a,b) = N_{ab}^{-1} \sum_t \sum_{(i,j)\in\mathcal{P}_{t,ab}} w^0_{tij}$, normalized to mean one over the cells. Averaging preserves pooled cell mass but loses within-cell variation, so the base-versus-SOGA comparison always uses the same region-shared base, and the original fine-grained rule is compared separately as a discretization control (Appendix E).

Coverage construction and correction details. Writing $\tilde{C}_q(\mathbf{U}) = \sum_{(a,b):\, q\in\{a,b\}} N_a N_b\, U(a,b)$ with N_a the mean daily count of region a, let $C_q = \tilde{C}_q / \sum_{q'} \tilde{C}_{q'}$, so $\sum_q C_q = 1$ and uniform weighting places about $1/Q$ in each region.

Scale is a normalization convention rather than an active constraint: every quantity in Equation 5 is invariant under $\mathbf{U} \mapsto c\mathbf{U}$ for $c > 0$, so rescaling can neither satisfy nor violate the band. The solver enforces the band and the coverage floor, each accepted with an additive slack of 10^{-9}, and the accepted table is rescaled to mean pair weight one over the 45 cells; the solvers for the published hinge tables and for cryptocurrency impose the mean-one equality directly at tolerance 10^{-6}, which selects the same table. The implemented divergence weights cells equally rather than by pair count; the two coincide when regions are equal-sized, exactly so on CSI1000, and agree to 2×10^{-5} at the reported solutions on the other markets. The floor prevents the correction from satisfying the band by emptying an extreme region, which is where a harmful allocation concentrates its weight; we set $\epsilon = 0.1$ on the CSI markets and $\epsilon = \min\{C_1(\mathbf{U}_{\mathrm{ref}}), C_Q(\mathbf{U}_{\mathrm{ref}})\}$ on US and cryptocurrency, that is 0.100, 0.0997 and 0.0972. Constraining only the two extremes is weaker than a floor on every region, and the results are reported as implemented. Among tables with finite divergence, minimizing D preserves the original emphasis as far as the constraints allow; a feasible base has zero divergence and is therefore returned unchanged. We solve the constrained problem numerically rather than relying on a uniqueness claim for the day-averaged response constraint. Infeasible cases are reported rather than treated as corrected arms.

Diagnosis versus correction. Diagnosis examines the halves separately because total displacement can hide cancellation. Correction instead constrains total Δ_h, including crossing pairs, to leave more freedom to preserve the base weighting; it does not force either half-wise displacement individually to zero or guarantee that the corrected table becomes unflagged. The separate-half variant moves nineteen of twenty tables further from their bases (Appendix H); Algorithm 1 in the main text lists the steps.

Divergence, support and feasibility. In Equation 4 the convention is $0 \log(0/u) = 0$ for $u > 0$ and zero contribution at $U = U_0 = 0$; if $U_0(a, b) = 0$, finite divergence requires $U(a, b) = 0$, so positive mass is never introduced on zero base support. Because every quantity in Equation 5 is scale-invariant, we report tables at mean weight one without treating scale as an active degree of freedom. Finite divergence preserves the base support, and a base that already satisfies the constraints has zero divergence and is returned unchanged. With the day-averaged ratio in Equation 2, we do not rely on convexity or uniqueness of the full constrained problem; the multi-start numerical procedure below is used to find the smallest-divergence feasible solution among the accepted starts. An instance can be infeasible when coverage demands mass on absent support or the displacement band cannot be reached on the retained support; infeasible cells are reported, never relaxed. The coverage shares use the products $N_a N_b$ of mean daily region counts, whereas the diagnostic response uses the per-day normalization of Equation 2; these are distinct definitions.

Correction tolerance and reproducibility details. The default δ is one shortcut-displacement detection width, which allows a small departure from the reference rather than exact neutrality; the sensitivity study varies it from 1 to 50 widths. Since Δ_h is a normalized directional response in the units of h, δ has the units of the declared shortcut, not of test RankIC, and it is fixed before Equation 5 is solved, because recomputing a width from each candidate table would change the optimization problem. The displacement is computed as the day-by-day normalized response averaged over days in the criterion and in most solvers, and as the pooled-ratio analogue defined above in the solvers for the published hinge tables and for cryptocurrency; on the reported tables the two agree to within 0.002 widths, except 0.012 for the corrected LambdaRank table on US, and every reported corrected table sits at $+1.00$ under either convention. Labels are standardized within each day before h is formed, and δ is twice the standard deviation of the daily normalized response of uniform weighting divided by the square root of the number of days: 0.00452 on CSI300, 0.00418 on CSI500, 0.00343 on CSI1000, 0.00376 on US and 0.00280 on cryptocurrency. Equation 5 is solved over the positive entries on the retained support, parameterized by log-weights bounded in $[-8, 8]$ by sequential least squares from 40 starts (the base table, the uniform table and 38 random draws), with function tolerance 10^{-12} and at most 1000 iterations; the accepted solution of smallest divergence is kept. The log-weight box is a numerical implementation bound, and our claims do not rely on a certified global optimum of the unconstrained parameterization. The frozen LambdaRank table is $U_0(a, b) = |G_a - G_b|$ with $G_q = 2^q - 1$ over label regions, so no score tie-breaking is involved. Same-region pairs are excluded; with $Q = 10$ the halves are regions 1–5 and 6–10. Training draws one trading day per batch with days shuffled each epoch, and each day's loss is normalized by its total pair weight. The coverage shares use the $N_a N_b$ weighting defined above.

Scaling and pooling. Adding a day-specific constant to h does not change within-day differences; multiplying it by a positive constant rescales A_h and the tolerance together, so each shortcut carries a declared scale. The primary criterion averages day-normalized responses, so each trading day contributes equally. Only the pooled-ratio solver variants noted above give days with more admitted pairs more influence; their reported solutions remain numerically close to the day-averaged convention.

Two reporting conventions. Single arms are scored per seed and averaged; ensembles average predictions over seeds first and score the average. The two conventions are reported side by side and never subtracted from one another.

D Markets, Backbones, Uncertainty, Oracles, and Trajectories

Primary comparison protocol. The primary benchmark comprises CSI300, CSI500, CSI1000 and US equities; cryptocurrency is an additional cross-asset evaluation described below. CSI300,

CSI500 and CSI1000 cover 619 trading days from June 2020 to December 2022 and rank four-day forward returns from 158 stock and 63 market features; the US market has 1,499 names over 710 days and ten-day returns. The primary backbone is the market-guided transformer of Li et al. (2024) with 775,041 parameters, trained for 40 epochs without early stopping; further backbones appear in Section 5.3. Primary comparisons match architecture, optimizer and budget and pair five random seeds; the oracle and trajectory diagnostics use the configurations given below. Uniform pairwise weighting is the reference $\mathbf{U}_{\mathrm{ref}}$, and the MSE objective of Li et al. (2024) is the backbone's published configuration. Single arms are scored per seed and averaged, and reported differences are read against a detection width θ, a circular block-bootstrap interval (Politis & Romano, 1992) and a Newey–West statistic, all defined under Uncertainty below.

Equity markets. CSI300, CSI500 and CSI1000 have roughly 300, 500 and 1000 names per day and test sets of 185,700, 309,421 and 619,000 stock-days over the 619-day test period from June 17, 2020 to December 30, 2022; constituent overlaps are 0, 0 and 120 rows, so the three share a market and a calendar but not constituents. The US panel holds 1,499 names over 710 test days. Its constituents are fixed: from US-listed symbols that were never members of the S&P 500 or the Nasdaq-100, those with at least 80% of trading days available over 2008 to 2017 were kept and the 1,500 with the highest mean daily dollar volume over 2015 to 2017 were selected, one of which had no dollar-volume data in the ranking window and was dropped, leaving 1,499 in the universe, of which 1,497 appear in the test window (median cross-section 1,466); the list includes about thirty index and sector funds alongside common stocks. Because membership was screened on the full history and the volume window overlaps the validation period, the panel carries survivorship and look-ahead in its constituent selection, though not in its features or labels; the CSI panels use the exchange's point-in-time index membership and do not. Labels are $\mathrm{Ref}(\mathtt{close}, -(k{+}1))/\mathrm{Ref}(\mathtt{close}, -1) - 1$ with $k = 4$ on CSI and cryptocurrency and $k = 10$ on US, recomputed from raw prices and matched to the stored panels at Spearman 1.000000. On the CSI markets training covers January 2, 2008 (October 31, 2014 for CSI1000) to March 31, 2020 and testing June 17, 2020 to December 30, 2022; on US training covers January 2, 2003 to December 31, 2015, validation January 4, 2016 to December 29, 2017, and testing January 2, 2018 to October 26, 2020; on cryptocurrency training covers January 1, 2021 to June 30, 2023 and testing July 1, 2024 to December 31, 2025. The forward labels of the last training days end before the test period begins, and the validation split is not used for model selection. Each stock-day carries 158 stock features and 63 market-index features over an eight-day window.

Cryptocurrency market. The cryptocurrency market was built after the criterion and its thresholds were fixed. Candidates are USDT pairs on one exchange; stablecoins, coins listed for under a year at the ranking date, and coins with under 80% coverage of the training period are removed; the 150 with the highest mean dollar volume over July 2020 to June 2023 are kept, ranked before the test period and including 45 that were delisted during it. Training starts January 1, 2021, revised from January 1, 2019 after the earlier start left 67 constituents, below the pre-set minimum of 100; the revision preceded any label statistic or training. The test period runs from July 2024 to December 2025, 549 days, with a median cross-section of 122; three volume-tier indices serve as the market features. Daily RankIC autocorrelation is 0.647 and the block length is 5. Table 4 gives the single-model results.

Backbones and training. Table 2 repeats the failure and the repair on CSI1000 across backbones spanning three orders of magnitude in parameter count, including recent time-series architectures. All rows use the published weighted hinge v1 as the flagged base and the same corrected table, so the repair comparison holds the objective fixed across architectures; it does not establish that every backbone responds identically to the same allocation, and recovery relative to uniform varies. The same eight backbones were also run with LambdaRank frozen at equal scores as the base (Appendix I): the flagged objective is below uniform on every backbone and the correction recovers 67% to 119% of the gap, above uniform on the same three backbones as in Table 2 and on TimesNet. The Transformer's (Vaswani et al., 2017) reference is indistinguishable from zero on this market, so its row is not evaluable and is omitted. Per-backbone ensembles are in Appendix I.

MASTER (Li et al., 2024) has 775,041 parameters and is trained with Adam at learning rate 10^{-5}, gradient clipping 3.0, 40 epochs, one trading day per batch, and no early stopping; the checkpoint used is the last. The other backbones are trained with AdamW at learning rate 10^{-3} for 40 epochs

Table 4: **Additional cross-asset single-model results.** Mean daily RankIC on cryptocurrency, using the same per-seed scoring convention as Table 1. The LambdaRank correction uses the table frozen at equal scores.

Objective	Crypto
MSE (Li et al., 2024)	$+0.0396$
Uniform pairwise ($\mathbf{U}_{\mathrm{ref}}$)	$+0.0580$
LambdaRank	-0.0072
Weighted hinge v1	-0.0181
Weighted hinge v2	-0.0031
Softplus hinge	-0.0210
ListNet	-0.0435
Top-k hinge (k=10%)	$+0.0413$
LambdaRank (frozen) + SOGA	$+0.0544$
Weighted hinge v1 + SOGA	$+0.0525$
Weighted hinge v2 + SOGA	$+0.0543$
Softplus hinge + SOGA	$+0.0539$

without early stopping and without tuning. LSTM (Hochreiter & Schmidhuber, 1997) and ALSTM (Qin et al., 2017) are the network classes of Qlib 0.9.7 (Yang et al., 2020) at their default widths. Linear is two linear layers applied to the last day's features, without the trend decomposition of DLinear (Zeng et al., 2023), and GRU (Cho et al., 2014) is a compact implementation that reads the last day's features as eight segments; FITS (Xu et al., 2024), TimeMixer (Wang et al., 2024) and TimesNet (Wu et al., 2023) are minimal implementations of the published architectures written for this study rather than the reference code. Parameter counts are those of the trained models. The MSE row of every backbone is its published objective with the same label processing, and a backbone–market block is evaluable only if its reference arm is distinguishable from zero.

Uncertainty. For each arm, the daily RankIC is first averaged over seeds; the paired difference between two arms is taken on their common days, and the detection width θ is twice the standard deviation of that daily series divided by the square root of the number of days, with a floor of 0.0025. Serial dependence, including that from overlapping multi-day labels, enters through two further readings: a circular block bootstrap (Politis & Romano, 1992) with a fixed block length of $\lceil (1+\rho_1)/(1-\rho_1) \rceil$ days and wrap-around, $\rho_1 = 0.569$ on equities, 2000 resamples and a 95% interval, and a Newey–West statistic with a Bartlett kernel of lag $\lfloor 4(T/100)^{2/9} \rfloor$. A difference is read as distinguishable only when all three agree. These estimators quantify temporal uncertainty after seed averaging rather than a separate seed-level confidence interval; for the principal repair comparisons we additionally report whether the paired seed differences agree in direction. These summaries concern paired test metrics, not the shortcut tolerance of Appendix C.

Oracles and mechanism interventions. Both oracles use CSI300 test predictions of MASTER averaged over five seeds, on the same four-day labels used in the primary CSI300 experiments. With the score $\hat{s}$ and the label y demeaned within each day, the direction oracle scores $|\hat{s}_i| \operatorname{sign}(y_i)$ and moves RankIC from 0.085 to 0.753; the magnitude oracle scores $|y_i| \operatorname{sign}(\hat{s}_i)$ and adds 0.006. Direction, not magnitude, is the bottleneck. The two training interventions of Section 3 point the same way: a model trained to rank magnitude reproduces the harmful allocation's ranking pattern, and comparing only stocks of similar magnitude during training turns the harmful allocation's ranking on CSI1000 from negative to positive. The magnitude component of an arm is its within-day rank correlation with $|y - \bar{y}|$ divided by that of an oracle magnitude ranker. A bucket-transition table records, for one trained model, how stock-days in each realized-return decile distribute across predicted-score deciles; two models are compared by the Spearman correlation of their flattened row-normalized tables.

Trajectories. On the US market neither the reference nor the corrected LambdaRank arm converges within the budget: both oscillate between 0.00 and 0.065, the corrected arm's five final readings span 0.039, its lowest seed peaks at epoch 5, and post-epoch-25 slopes are positive for both arms, so the traces show no sustained late decline in RankIC; they do not measure a return to the

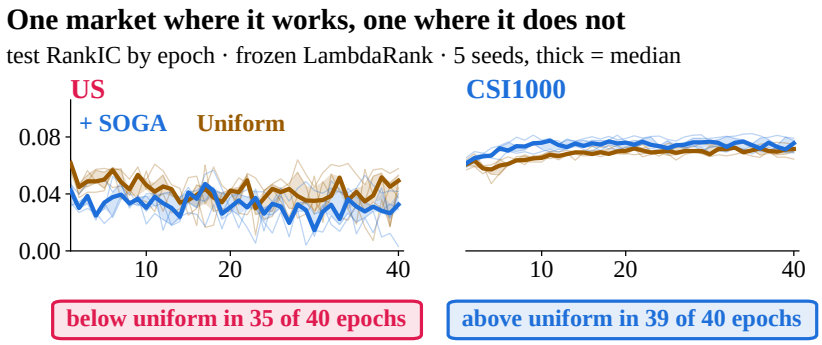

Figure 5: Test RankIC by epoch for the uniform reference and frozen LambdaRank + SOGA on US (top) and CSI1000 (bottom), on a shared vertical scale; thin lines are the five seeds and the thick lines their median with the interquartile band. The corrected arm is below uniform weighting at 35 of the 40 epochs on US and above it at 39 of the 40 on CSI1000. The final readings span 0.039 (corrected) against 0.020 (uniform) on US, and 0.010 against 0.011 on CSI1000.

shortcut directly. On CSI1000 both arms settle, the reference peaking at epochs 22 to 39 across the five seeds (Figure 5).

E ALLOCATIONS, PUBLISHED OBJECTIVES, AND CONTROLS

Matched modes. The three allocations of Figure 1 draw on per-bucket shares $\{17.0, 13.9, 10.4, 6.5, 2.3\}\%$, each value used twice. Both-ends places the largest shares at both extremes; top-only and bottom-only place the same shares in opposite rank directions and are exact reversals of one another. Their trained RankIC values on CSI300/CSI500/CSI1000/US are: uniform $+0.0632/+0.0522/+0.0706/+0.0484$; both-ends $+0.0806/+0.0622/+0.0712/+0.0403$; bottom-only $+0.0521/+0.0638/+0.0795/+0.0689$; top-only $+0.0078/-0.0340/-0.0378/-0.0417$; on cryptocurrency $+0.0580/+0.0520/+0.0750/-0.0129$. A permutation control on the published label-distance rule draws one uniform permutation of its 45,000 pair weights on CSI300 under a fixed seed, leaving the multiset of values and the pair set unchanged; it moves the rule from $+0.0776$ to $+0.0617$, a loss of about one fifth, and below the uniform reference at $+0.0632$.

Published objectives. Changes in RankIC against uniform weighting on the five markets (CSI300/CSI500/CSI1000/US/Crypto) are: LambdaRank $-0.062/-0.074/-0.090/-0.083/-0.065$; hinge v1 $-0.055/-0.083/-0.105/-0.090/-0.076$; hinge v2 $-0.048/-0.075/-0.088/-0.086/-0.061$; softplus hinge $-0.068/-0.089/-0.111/-0.092/-0.079$, all with paired ratios between -2.9 and -7.1 and every seed co-directional; ListNet $-0.073/-0.084/-0.103/-0.091/-0.101$. On the four equity panels ListMLE (Xia et al., 2008), cross-sectional squared error, and the IC loss are indistinguishable from uniform, as is the finite-p push family (Rudin, 2009) on CSI1000; top-k hinge is below uniform on every market and infinite push (Agarwal, 2011) by -0.024 on CSI1000. All fifteen hinge cells are flagged by the criterion, and their repairs are feasible, with displacements of $+119$ to $+172$ widths returned to $+1$. Hinge v1 and softplus hinge share a base weight table and therefore a corrected weight table, but retain different losses; their trained performance rows in Table 1 do not coincide. The push family is displaced negatively because a push weight at the reference is a power of how many comparisons an item enters from below, which weights comparisons where the lower-ranked item has the larger magnitude; top-k weights depend on the current prediction and at initialization reduce in expectation to uniform, so the equal-score screen does not characterize their later prediction-dependent weights.

Controls. Retraining an identical model under five seeds gives a reference standard deviation of 0.0058, illustrating why seed variability must be reported (Bouthillier et al., 2021). A continuous-rank implementation reproduces the failure, with discretization explaining 10% to 16% of the gap. Transferring the per-item weights of the top-only allocation into a squared-error objective reproduces 99.6% of the 0.0575 pairwise cost of that allocation against uniform weighting on CSI300 in the

Table 5: Outcome by configuration on 96 decidable cells (98 with two zero-configuration cells set aside), and the 50 flagged cells by kind of table. Harmful means below the uniform reference.

Configuration	harmful	benign
Both halves positive (flagged)	45	5
Both halves negative	5	6
Halves of opposite sign	11	24

Flagged cells by kind	harmful	benign
Base allocations	22	0
Support-restricted variants	3	0
Corrected tables	7	2
Ablations of the correction	0	2
Null controls	13	1

three-seed matched-mode run (0.0634 against 0.0059); this weighted control is distinct from the unweighted MSE of the backbone's published configuration. Deleting the two deciles that carry the top-only weight recovers 0.0498 of its shortfall on CSI300, deleting the two at the opposite end costs a further 0.0067, and random bucket deletion costs 0.001 to 0.007. Splitting the test period into halves preserves every matched-mode verdict, and on CSI1000 the top-only allocation is below uniform weighting by 0.097, 0.093 and 0.108 on three backbones spanning three orders of magnitude in parameter count (Linear, GRU and MASTER, five seeds).

F Criterion Validation on Trained Cells

Validation scope. In the 96-cell evaluation, all 22 flagged base allocations degrade relative to uniform weighting; the broader collection, which adds corrected tables, ablations and controls, contains both false positives and false negatives (Table 5).

Two cell sets. The criterion was formulated on 41 allocation–market pairs (21 both-positive, none benign; 7 both-negative, 5 benign; 12 opposite-sign, 10 benign; one zero). The larger set of 96 decidable cells adds corrected tables, ablations of the correction, null controls, support-restricted variants and the symmetrized controls below; because several tables carry more than one trained arm, the two sets are matched by arm identity, which pairs the 40 decidable formulation pairs with 40 cells and leaves 56 held-out cells. These held-out cells remain within related allocation and market families, so this is a cell-level validation rather than a leave-one-market or leave-one-objective-family test. Precision and recall of the flag are 90.0% and 73.8% overall and 82.8% and 66.7% on the held-out cells. Read as a two-sided test the half-wise rule agrees with the outcome on 78.1% of cells against 67.7% for the sign of the total displacement, and on the 34 cells where the two disagree it is right on 22. Of the five benign flagged cells, two are within one width of the reference and three above it by up to 2.24 widths; none is a base allocation. A further twenty predictions made after the pre-registered list was closed agree on eighteen; both misses are displaced below 28 widths, and among all forty predictions, registered and not, the twenty-two allocations displaced by at least 28 widths are all predicted correctly.

Granularity and the silent configurations. The flagged set is identical at $Q = 5$, 10 and 20 on the 33 cells whose table can be rebuilt at another Q, a check of the screen without retraining; only the split between the two unflagged configurations moves. The opposite-sign configuration was tested directly with a symmetrized top-only table, $\frac{1}{2}(\mathbf{U}_0 + \mathbf{U}_0^{\mathrm{rev}})$ with $\mathbf{U}_0^{\mathrm{rev}}$ the rank reversal, whose total displacement is between $+0.8$ and $+2.1$ widths while its halves stay at $+12.6$ to $+17.6$ and -8.8 to -12.5; its coverage of 0.095 to 0.096 is below the floor, so it is a control rather than a correction. Trained, it ends at -0.78, $+1.29$, $+0.15$ and -2.29 widths against the reference on the four equity panels, and the both-ends allocation, of opposite sign on cryptocurrency, ends at -0.98 there. Two ablations that replace the KL objective by L_2 and maximum-entropy objectives are flagged and not harmful; two cells are too few to read.

Table 6: **Head metric versus spread.** Change against $\mathbf{U}_{\mathrm{ref}}$; share is the magnitude-trained control's NDCG gain as a fraction of the top-only allocation's.

Market	Arm	RankIC	Spread	NDCG@10%	Share
CSI300	top-only	-0.0555	-0.0020	$+0.0538$	
	trained on h	-0.0955	-0.0075	$+0.0482$	0.90
	top-only + SOGA	-0.0074	-0.0003	$+0.0115$	
CSI500	top-only	-0.0862	-0.0026	$+0.0763$	
	trained on h	-0.1062	-0.0046	$+0.0765$	1.00
	top-only + SOGA	-0.0021	$+0.0001$	$+0.0028$	
CSI1000	top-only	-0.1084	-0.0075	$+0.0734$	
	trained on h	-0.1371	-0.0117	$+0.0692$	0.94
	top-only + SOGA	-0.0001	$+0.0001$	$+0.0036$	
US	top-only	-0.0901	-0.0075	$+0.1009$	
	trained on h	-0.1052	-0.0117	$+0.0935$	0.93
	top-only + SOGA	-0.0162	-0.0016	$+0.0225$	

G SPREAD AND HEAD METRICS

What the emphasis buys. The spread is the difference between the mean realized k-day returns of the top and bottom score deciles, averaged over test days; NDCG uses the within-day label decile as gain. Both are ordering diagnostics on held-out days, not executable returns: nothing is annualized, no turnover or cost is applied, and no portfolio is constructed. On the four equity panels, top-only improves NDCG by 0.054 to 0.101 at ratios of 6 to 17 but reduces the spread. A model trained only on h has negative rank correlation with returns and yet reproduces 90 to 100% of that NDCG gain (Table 6), so an NDCG gain alone does not distinguish a better return ranking from a magnitude ordering. The repaired top-only spread is within one reported width of uniform on all four panels, and the screened ensemble is above uniform on the spread but below it on NDCG, which separates the two metrics further.

Spread levels by arm (CSI300/CSI500/CSI1000/US): uniform $+0.0076/+0.0040/+0.0079/+0.0090$; top-only $+0.0056/+0.0014/+0.0004/+0.0015$; top-only + SOGA $+0.0073/+0.0041/+0.0080/+0.0074$; LambdaRank (frozen) + SOGA $+0.0094/+0.0049/+0.0086/+0.0046$; both-ends $+0.0096/+0.0054/+0.0080/+0.0070$; bottom-only $+0.0025/+0.0025/+0.0072/+0.0069$; label-distance $+0.0095/+0.0048/+0.0085/+0.0078$. The repaired top-only table is within one width of uniform on all four markets, and the repaired LambdaRank table is above uniform on the three CSI markets by 0.0018, 0.0008 and 0.0006. Precision at $k = 50$ behaves like NDCG: the magnitude-trained model reaches $1.53\times$ chance on CSI300, as does the top-only allocation, against uniform's 1.11, with a spread of $+0.0001$. On CSI300 the magnitude model scores 0.2743 and 0.3282 at NDCG@10% and @20% against 0.2261 and 0.2707 for uniform, gains of $+0.0482$ and $+0.0575$, or 90% and 93% of the top-only allocation's; on the other three markets the shares are $1.00/1.01$, $0.94/0.99$ and $0.93/0.97$. A head metric therefore rewards a magnitude ordering as much as a return ordering on this task, and on its own it would rank an inverted model first.

H ALTERNATIVE CORRECTIONS, SENSITIVITY, AND BOUNDARY CASES

Repair decomposition and penalty functions. Against $\mathbf{U}_{\mathrm{ref}}$, the corrected rows of Table 1 have paired detection widths of 0.0025 to 0.0085; against their own uncorrected rows the widths are 0.0099 to 0.0197, and each of the sixteen corrections gains 3.5 to 7.7 of them. Can weight correction repair a harmful allocation without replacing it by uniform weights? Because LambdaRank updates its weights from the current ranking, we first freeze the rule at the equal-score reference and then apply SOGA to that fixed table, which separates two interventions that a direct comparison with the published objective would confound; the base-versus-correction comparison holds the rank-region representation fixed, and separate discretization controls are in Appendix E. Freezing alone recovers 1% to 18% of the damage. Correcting the frozen table recovers a further 3.6 to 6.1 detection widths on all four equity panels with every seed agreeing, and the corrected objective has positive mean rank correlation on every market, including those where the published and frozen versions invert the ranking; the correction, not the freezing, accounts for the recovery. The same holds

for the three published weighted hinge variants: all fifteen market–variant cells are flagged, and correcting them recovers 3.5 to 7.7 detection widths, most on the US market where the base is worst. Weighted hinge v1 and softplus hinge share a base weight table and therefore a corrected table, but keep different penalty functions and have different trained results (Table 1); their common failure and recovery support a role for weight placement across these penalties, not invariance to the penalty. The corrected frozen LambdaRank table on the US market, 0.019 below $\mathbf{U}_{\mathrm{ref}}$, is the clearest remaining shortfall.

Selectivity relative to uniform weighting. The projection study includes unflagged bases as ablations of Equation 5, although Algorithm 1 leaves such inputs untouched. Across five bases and four markets, the projection moves the mildly displaced label-distance weighting by under 0.3% of the KL distance to the reference table and the constructed harmful allocation by up to 77%, while uniform shrinkage of the same bases along the geometric path $\mathbf{U}_0^t$, stopped at the coverage centroid of the SOGA solution, moves 88 to 97% of the way to the reference table on the twelve base–market cells where it was run (top-only, bottom-only, frozen LambdaRank and weighted hinge v1) and has no counterpart for the label-distance base; the trained RankIC of the label-distance weighting is indistinguishable from its uncorrected value on the four equity panels. This selectivity follows from minimizing change within the correction constraints.

Matched alternative corrections. Uniform shrinkage tests whether moving toward uniform is enough; the correlation penalty tests suppression at the output level; gradient projection constrains score gradients during training. SOGA outperforms uniform shrinkage and the correlation penalty on all four equity panels in matched comparisons that apply each alternative to the same base (Table 7). Uniform shrinkage, stopped at the coverage centroid of the SOGA solution, moves nearly all the way toward uniform weighting while its shortcut response remains 28 to 43 detection widths outside the band; SOGA exceeds it by 2.0 to 4.1 detection widths. The correlation penalty acts on predicted scores and is swept over two orders of magnitude; SOGA exceeds it by 3.1 to 5.9. Gradient projection, which removes the score-gradient component along the shortcut at every training step, is the closest competitor: SOGA is ahead on CSI300 and CSI1000 by 2.3 and 2.6 widths, within one width on CSI500, and behind on the US market. The two projection targets differ by less than one reported detection width on each panel; this is not an equivalence test. Their operational distinction from SOGA is that a per-step constraint on the score gradient is altered by the Jacobian at every update (Appendix A), whereas a table constraint is enforced once and the original objective trains against it. Like uniform shrinkage, SOGA can be precomputed and needs no shortcut values during model optimization, unlike these gradient-projection and correlation-penalty controls.

These comparisons support the form of the correction without establishing that every design choice is necessary: the two L_2 and maximum-entropy objective ablations are too few to establish the superiority of the KL objective, and the δ and ϵ sweeps report final model RankIC only for the local check below.

Gradient projection. The zero version removes at every step the full component of the score gradient along the batch-centered shortcut; the reference version removes only the excess over the same component under uniform weighting at the same scores, $g' = g - [(h^\top g - h^\top g_{\mathrm{ref}})/h^\top h]\, h$. The reference version projects out 58% of the gradient against 65% for the zero version and ends within one width of it on every market. Gradient projection drives the learned magnitude dependence from $+0.428$ to -0.046, consistent with its stronger suppression of the shortcut, and it requires the shortcut at every training step.

Sensitivity to the correction bounds. The sweep measures the corrected weight tables, not final model performance; its correlations are between allocations, not between predicted scores and returns. Widening δ from 1 to 50 widths moves the solution's displacement to the band edge at every setting but changes its rank correlation with the harmful base only from 0.50 to 0.76; lowering ϵ from 0.17 to 0.023 spans 62% of the achievable KL range against 38% for the tolerance. Neither sweep produces a solution close to the both-ends allocation, whose rank correlation with the corrected tables never exceeds 0.55. Figure 4 illustrates correction selectivity on CSI1000. A local downstream check trains MASTER on CSI1000 with the frozen LambdaRank base over the same five seeds. Widening δ to five times the default gives final RankIC $+0.0738$ against $+0.0741$ at the

Table 7: **Comparison with other corrections**, mean daily RankIC. The upper block starts from the frozen LambdaRank table, the lower block from the top-only allocation; each correction is applied to the same base as SOGA in its block. Δ rows give the paired ratio of SOGA against the named row.

	CSI300	CSI500	CSI1000	US
Reference allocation $\mathbf{U}_{\mathrm{ref}}$	$+0.0632$	$+0.0522$	$+0.0706$	$+0.0484$
Base: LambdaRank, frozen at the reference				
Base, uncorrected	$+0.0120$	-0.0208	-0.0146	-0.0344
+ uniform shrinkage	$+0.0610$	$+0.0383$	$+0.0604$	-0.0010
+ SOGA	$+0.0773$	$+0.0589$	$+0.0741$	$+0.0293$
Δ SOGA vs. shrinkage	$+2.01$	$+3.47$	$+3.18$	$+4.10$
Base: top-only allocation				
Base, uncorrected	$+0.0078$	-0.0340	-0.0378	-0.0417
+ gradient projection, to zero	$+0.0344$	$+0.0471$	$+0.0498$	$+0.0508$
+ gradient projection, to reference	$+0.0295$	$+0.0373$	$+0.0477$	$+0.0384$
+ correlation penalty	$+0.0120$	-0.0244	-0.0185	-0.0234
+ SOGA	$+0.0558$	$+0.0501$	$+0.0705$	$+0.0322$
Δ SOGA vs. projection to zero	$+2.25$	$+0.37$	$+2.57$	-1.77
Δ SOGA vs. projection to reference	$+3.43$	$+1.89$	$+3.40$	-1.34
Δ SOGA vs. correlation penalty	$+3.13$	$+4.79$	$+5.90$	$+5.43$

default ($\Delta/\theta = -0.10$). Halving ϵ returns the default table, because the end-region floor does not bind at the default solution, whose smaller end-region share is 0.123.

Alternative constraints. The comparisons below describe feasible solutions and how much they change the base, without downstream RankIC for those solutions; they explain the constraint trade-offs rather than establishing a trained advantage for the default choice. A one-sided constraint coincides with the two-sided one on every base displaced toward the shortcut. Constraining the two halves separately is feasible on all twenty base–market cells but moves nineteen of them further from the base, and on the label-distance base on CSI300 raises the fraction travelled from 0.2% to 13.6%. Replacing the displacement constraint by the criterion itself leaves the both-ends allocation unchanged, but on the other sixteen cells the solver drives one half to zero and leaves the other untouched, producing half-repaired tables with displacements of $+60$ to $+181$ widths; maximizing end-region weight instead yields corner solutions with dynamic range in the thousands. Neither achieves the intended trade-off between controlling displacement and retaining the base, and their trained performance has not been evaluated.

Boundary cases. The bottom-only allocation, displaced away from the shortcut by 120 to 157 widths, is moved 58 to 60% of the way to the reference table because the coverage floor at the high end binds, yet its trained ranking changes by at most 0.93 widths on the three CSI markets. The corrected LambdaRank table on the US market ends 0.019 below uniform weighting on rank correlation and 0.0043 below on the spread; the US market is also where neither arm converges within the budget (Appendix D).

I ENSEMBLE DEFINITIONS, LEAVE-ONE-OUT, AND SELECTION ACROSS HALVES

Definitions. For each admitted allocation m, predictions are averaged over seeds, $\bar{p}_{ti}^{(m)} = S^{-1}\sum_s p_{ti}^{(m,s)}$, converted to within-day ranks $r_{ti}^{(m)}$, and the ensemble scores stock i on day t by the mean rank over members. ENS-A uses uniform, both-ends, bottom-only and label-distance; ENS-B adds the repaired LambdaRank and top-only tables; ENS-F replaces both-ends and label-distance in ENS-A by the uncorrected top-only and LambdaRank tables; ENS-G uses uniform, bottom-only, both-ends and top-only; ENS-C uses all eight; ENS-D averages five seeds of uniform weighting alone. Rank averaging was fixed before computation; z-score averaging is within one width of it on every market.

Table 8: **Screened ensemble results.** Mean daily RankIC under the ensemble convention: predictions are averaged over seeds before scoring, and the uniform reference uses the same convention. These means are separate from the single-model comparison in Table 1.

	CSI300	CSI500	CSI1000	US	Crypto
Uniform pairwise, ensemble convention	$+0.0700$	$+0.0578$	$+0.0783$	$+0.0607$	$+0.0674$
ENS-A (ours)	$\mathbf{+0.0888}$	$\mathbf{+0.0692}$	$\mathbf{+0.0867}$	$\mathbf{+0.0646}$	$\mathbf{+0.0751}$
ENS-B (ours, with repaired objectives)	$+0.0874$	$+0.0673$	$+0.0857$	$+0.0575$	$+0.0718$

Table 9: **Ensemble ablations**, paired ratio Δ/θ. ENS-A: uniform, both-ends, bottom-only, label-distance, all admitted by the screen (three members on cryptocurrency, where label-distance had not been trained when membership was fixed). ENS-F: two admitted members replaced by top-only and LambdaRank. ENS-G: one flagged member. ENS-C: all eight allocations. ENS-D: five seeds of uniform weighting.

	CSI300	CSI500	CSI1000	US	Crypto
ENS-A vs. ENS-F (screen only)	$+2.30$	$+3.40$	$+4.31$	$+4.21$	$+3.94$
ENS-F vs. uniform	-1.59	-3.44	-4.32	-4.66	-3.47
ENS-G vs. ENS-A	-1.18	-2.09	-2.47	-1.65	-2.25
ENS-C vs. uniform (no screen)	$+1.46$	-0.59	-1.10	-2.83	-2.05
ENS-A vs. ENS-D (seeds only)	$+2.16$	$+2.18$	$+1.99$	$+0.63$	$+1.26$

Screened combinations. The admitted weightings can be used together. ENS-A averages the within-day ranks of the admitted allocations after averaging each allocation's predictions over seeds, with membership and aggregation fixed before any ensemble was computed. It is above $\mathbf{U}_{\mathrm{ref}}$ on all five markets and above its detection width on four (Table 8); no single allocation in the pool is above $\mathbf{U}_{\mathrm{ref}}$ on every market, the closest, bottom-only, being below on CSI300.

Table 9 shows that the gain comes from the screen and not from averaging alone. ENS-F keeps the size of ENS-A but replaces two admitted members by the two flagged ones, and falls below uniform weighting on every market; ENS-A is above it by 2.3 to 4.3 detection widths on all five. ENS-G keeps the strongest single member and admits one flagged arm, and is below ENS-A on all five markets. ENS-C admits every allocation and is below uniform on four. ENS-D averages five seeds of uniform weighting, serving as a seed-averaging control rather than a matched total-training-budget control, and ENS-A is above it on every market, by about two detection widths on the CSI markets. Leave-one-out below shows that no single member is necessary for a positive mean margin on the CSI panels, while on the US and cryptocurrency markets the bottom-only member is essential. The criterion admits members by the geometry of their weight tables rather than by measured performance, so an admitted member can still be weak on a particular market; the substitutions tested support screening without establishing the effect of every possible member.

Ensembles across backbones, and the LambdaRank-based repair. The screened ensemble, built per backbone from the same admitted allocations, has a higher mean than the backbone's single-model uniform objective on every backbone where it has been computed (Table 10); the two follow different reporting conventions (Appendix C), so the gap is not read as an effect size.

Every member averages five seeds. On cryptocurrency the label-distance allocation had not been trained when membership was fixed, so ENS-A there has three members and the corresponding leave-one-out entry is empty. The LambdaRank members of ENS-F and ENS-C use the table frozen at equal scores; ENS-C consists of uniform, both-ends, bottom-only, label-distance, the corrected LambdaRank and top-only tables, and the uncorrected LambdaRank and top-only tables.

Leave-one-out. In Table 11, removing bottom-only is the only removal that turns the ensemble's margin negative, on US and cryptocurrency; on the three CSI markets every three-member ensemble remains above uniform. On CSI300, where bottom-only is itself below uniform, removing it produces a nearly identical mean margin, while removing any other member puts the margin within the detection width. This is consistent with complementary members, although it does not isolate diversity as the cause. On cryptocurrency removing both-ends raises the ensemble, consistent with

Table 10: **Cross-backbone study on CSI1000 with LambdaRank frozen at equal scores as the base, and screened ensembles.** Mean daily RankIC, five seeds. Uniform, MSE and the hinge-based repair are in Table 2; recovery is defined as there. ENS-A is the screened ensemble built per backbone from the admitted allocations.

Backbone	Uniform	LR-frozen	+ SOGA	Recovery	ENS-A
Linear	$+0.0557$	-0.0136	$+0.0656$	114%	$+0.0700$
FITS	$+0.0557$	-0.0128	$+0.0684$	119%	$+0.0715$
GRU	$+0.0660$	-0.0014	$+0.0525$	80%	$+0.0806$
ALSTM	$+0.0525$	$+0.0022$	$+0.0364$	68%	$+0.0796$
TimeMixer	$+0.0671$	$+0.0089$	$+0.0598$	87%	$+0.0863$
LSTM	$+0.0406$	$+0.0116$	$+0.0311$	67%	$+0.0709$
TimesNet	$+0.0620$	-0.0102	$+0.0626$	101%	$+0.0844$
MASTER	$+0.0706$	-0.0146	$+0.0741$	104%	$+0.0867$

Table 11: Leave-one-out from ENS-A; entries are the change against uniform under the ensemble convention, starred where above the detection width.

Market	all four	−uniform	−both-ends	−bottom-only	−label-distance
CSI300	$+0.0188^{*}$	$+0.0187$	$+0.0143$	$+0.0187^{*}$	$+0.0153$
CSI500	$+0.0114^{*}$	$+0.0131^{*}$	$+0.0102^{*}$	$+0.0074^{*}$	$+0.0124^{*}$
CSI1000	$+0.0084^{*}$	$+0.0083$	$+0.0083^{*}$	$+0.0051$	$+0.0087^{*}$
US	$+0.0039$	$+0.0029$	$+0.0085$	-0.0050	$+0.0073$
Crypto	$+0.0077^{*}$	$+0.0085$	$+0.0122^{*}$	-0.0021	

its opposite-sign configuration on that market. ENS-A's NDCG at the top decile is below uniform on every market, consistent with its having no head-weighted member (Appendix G).

Selection across halves of the test period. Selecting among the five base allocations by mean RankIC on one half of the test period and evaluating on the other, with a tie-break fixed in advance, gives ratios of $+0.34$, $+0.76$, -0.28 and $+0.87$ against uniform when selecting on the first half and -1.08, $+0.24$, -0.15 and $+1.51$ when selecting on the second, with different arms selected on three of the four equity panels; the two folds share data and the second is a symmetry check rather than an independent reading. Selection among weightings does not transfer across the two halves, which is why the screened ensemble averages the admitted arms instead of selecting one.