REGULARIZATION-ALIGNED LEARNING FOR SPARSE PORTFOLIO OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Sparse portfolio optimization seeks high risk-adjusted return from a small set of assets. Optimizing portfolios using estimated returns and risk can amplify estimation error and impair out-of-sample performance. We refer to systematic modifications intended to mitigate this instability broadly as *regularization*. Learned return forecasts provide one form of regularization, and decision-focused learning incorporates portfolio performance into model training. Yet the existing decision-focused sparse tangency method evaluates dense soft portfolios during training and selects coordinates that mix assets, so its learning objective need not reflect the intended sparse asset decisions. Tangency decisions alone also leave forecast scale unidentified, although it affects signal aggregation and uncertainty adjustment. In this work, we introduce Regularization-Aligned Sparse Tangent Portfolio Optimization (RA-STPO) to make the form, scale, and use of regularization explicit. It learns bounded corrections to a frozen prediction anchor by averaging losses on resampled sparse return paths, then uses calibration, measured uncertainty, and past relative performance to control their deployment. In the main comparison across five markets, RA-STPO achieves the highest average Sharpe ratio and terminal wealth. Our code is available at `https://anonymous.4open.science/r/RA-STPO-A210`.

1 INTRODUCTION

Sparse portfolio optimization seeks a fully invested long-only portfolio that holds at most k of N assets, where $1 \le k \le N$. For an expected excess-return signal $a \in \mathbb{R}^N$ and covariance estimate $\widehat{\Sigma} \succ 0$, its empirical tangency problem is

$$\mathcal{W}_k = \{w \ge \mathbf{0} : \mathbf{1}^\top w = 1,\ \|w\|_0 \le k\},$$
$$S(w; a, \widehat{\Sigma}) = \frac{a^\top w}{\sqrt{w^\top \widehat{\Sigma} w}}, \qquad S_k^*(a, \widehat{\Sigma}) = \max_{w \in \mathcal{W}_k} S(w; a, \widehat{\Sigma}). \tag{1}$$

This is the cardinality-constrained form of classical mean–variance and Sharpe-ratio optimization (Markowitz, 1952; Sharpe, 1966). Enumerating the $\binom{N}{k}$ asset sets recovers the empirical optimum when a positive-return direction exists. Mixed-integer methods solve related cardinality-constrained portfolio problems to global optimality (Bienstock, 1996; Gao & Li, 2013; Bertsimas & Cory-Wright, 2022). Relaxations, first-order methods, and asset-selection heuristics reduce the cost of searching over asset subsets (Kim et al., 2016; Lin et al., 2024a;b; Bae et al., 2025; Lin et al., 2025).

The empirical target is precise, but its inputs are not. Portfolio optimizers amplify estimation error, especially in the mean (Michaud, 1989; Best & Grauer, 1991; Chopra & Ziemba, 1993), and in-sample optimization need not outperform simple diversification out of sample (DeMiguel et al., 2009b). Mean and covariance shrinkage (Jorion, 1986; Ledoit & Wolf, 2004), weight penalties (Brodie et al., 2009), long-only and norm constraints (Jagannathan & Ma, 2003; DeMiguel et al., 2009a), and robust objectives (Goldfarb & Iyengar, 2003) all respond by changing how estimated moments enter the decision. We call any such systematic modification *regularization* in a broad operational sense.

Learned forecasts are also regularizers under this definition. Prediction-focused learning (PFL) trains forecasting models to minimize prediction error, whereas decision-focused learning (DFL) trains them using the loss of the resulting decisions (Elmachtoub & Grigas, 2022; Mandi et al., 2024; Shah et al., 2022). In portfolio optimization, the decision layer maps each forecast to the allocation

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.

evaluated by that loss. Differentiable convex optimization layers enable backpropagation through optimal solutions by implicitly differentiating optimality conditions (Amos & Kolter, 2017; Agrawal et al., 2019). For sparse portfolios, asset selection is discrete, and smoothing this selection typically introduces a temperature or perturbation scale (Berthet et al., 2020).

The method of Jeon et al. (2026), which we call *DF-STPO*, applies decision-focused learning to sparse tangency portfolio optimization. We focus on two implications for the regularization induced by this construction. First, training evaluates dense soft portfolios and selects Cholesky-transformed coordinates that can mix assets, following OSCAR (Bae et al., 2025). The allocations shaping forecast learning therefore need not represent the intended sparse asset choices. Second, DF-STPO regularizes forecasts by balancing prediction and decision losses. Prediction error supplies scale information, but the mixture weight does not directly specify how far each asset's forecast may depart from a predictive reference. Since tangency decisions do not identify absolute forecast scale, signal averaging and uncertainty adjustment require an explicit reference for forecast scale.

In this paper, we make the following contributions.

- We analyze the asset representation and scale invariance of sparse tangency decisions and their implications for decision-focused regularization.
- We formulate an uncertainty-scaled forecast correction scheme that learns from resampled sparse return paths using exact allocation gradients and residual-based asset selection.
- We introduce **Regularization-Aligned Sparse Tangent Portfolio Optimization (RA-STPO)**, which connects this learning rule with uncertainty-adjusted mean signals and performance-adaptive sparse allocation. In the main comparison across five markets, RA-STPO achieves the highest average Sharpe ratio and terminal wealth with matched network counts.

2 Decision-Focused Learning for Sparse Portfolio Optimization

2.1 The DF-STPO surrogate

At date t, let X_t contain information from the T-day return window and let y_{t+1} be the next return. DF-STPO predicts $\widehat{\mu}_t = f_\theta(X_t)$ and factors its positive-definite covariance estimate as $A_t^\top A_t = \widetilde{\Sigma}_t$. It first obtains a continuous tangency direction u_t from a risk-constrained program. For $k < N$, it scores transformed coordinates by $s_{ti} = |(A_t u_t)_i|$ and replaces hard selection with

$$p_{ti} = \mathrm{sigmoid}(\beta[s_{ti} + \tau_\beta(s_t)]), \qquad \sum_{i=1}^{N} p_{ti} = k. \tag{2}$$

The shared shift τ_β enforces the sum constraint, and β controls sharpness. A second solve uses the masked forecast $p_t \odot \widehat{\mu}_t$. Training combines its realized portfolio loss with prediction error:

$$\mathcal{L}_\alpha = \alpha \mathcal{L}_{\text{decision}} + (1-\alpha)\mathcal{L}_{\text{prediction}}, \qquad 0 \le \alpha \le 1. \tag{3}$$

Evaluation replaces the soft mask with hard top-k selection and re-optimizes the weights. Appendix A.3 gives the convex program, its analytic implementation, and the algorithm.

2.2 Implications of the Decision Surrogate

Decisions in asset coordinates. At finite β, every mask entry in (2) is positive. The constraint $\sum_i p_i = k$ controls mask mass without enforcing portfolio cardinality. Training can therefore reward asset combinations unavailable to a k-asset portfolio, and hard selection at evaluation does not change the allocations evaluated during training. Selection geometry introduces a separate distinction. An asset-supported portfolio maps into the span of the corresponding columns of A_t, generally not a coordinate subspace. Selecting transformed coordinates instead restricts the transformed vector to a coordinate subspace. The following proposition characterizes transformations that preserve asset supports up to relabeling.

Proposition 2.1. *For $F \subseteq [N] = \{1, \dots, N\}$, let $E_F = \{x : \mathrm{supp}(x) \subseteq F\}$. A nonsingular $A \in \mathbb{R}^{N\times N}$ satisfies $AE_F = E_{\pi(F)}$ for every F under one fixed relabeling π if and only if $A = P_\pi D$ where P_π is a permutation matrix and D is nonsingular and diagonal.*

Preserving the same labels requires A to be diagonal. A triangular Cholesky factor of a correlated covariance is not of this form and also depends on asset order. Under hard top-k selection, DF-STPO still selects transformed coordinates and treats their indices as asset indices. Appendix E.1 verifies the ordering effect numerically.

The scale of forecast corrections. The loss mixture in (3) can regularize forecasts through prediction error. The following proposition shows that the tangency decision itself cannot distinguish positive rescalings of the same signal.

Proposition 2.2. *For fixed $\widehat{\Sigma} \succ 0$, any $a \in \mathbb{R}^N$, and every $c > 0$,*

$$\arg\max_{w \in \mathcal{W}_k} S(w; ca, \widehat{\Sigma}) = \arg\max_{w \in \mathcal{W}_k} S(w; a, \widehat{\Sigma}). \tag{4}$$

This holds because $S(w; ca, \widehat{\Sigma}) = cS(w; a, \widehat{\Sigma})$: multiplying all Sharpe ratios by the same $c > 0$ leaves the maximizing set unchanged. Prediction error can supply scale information, but a loss-mixture weight does not directly bound each asset's departure from a predictive reference in return units. An uncertainty-scaled correction bound makes regularization strength interpretable across assets and dates and supplies a common reference for signal averaging and uncertainty adjustment.

3 Proposed Method

Motivated by the analysis in Section 2, we propose Regularization-Aligned Sparse Tangent Portfolio Optimization (RA-STPO). RA-STPO follows two principles: derive regularization scales from quantities available at the decision date, and use each quantity in a role consistent with what it represents. A prediction anchor defines the reference, uncertainty sets correction bounds and conservative mean adjustments, and observed relative performance controls the correction's use. Figure 1 follows this progression. Detailed proofs are collected in Appendix C.

3.1 Sparse asset selection and exact allocation

We first construct a sparse allocation map and derive its exact local gradients. The scale invariance of the Sharpe ratio gives a quadratic reformulation. With $\mathcal{C}_k = \{v \geq \mathbf{0} : \|v\|_0 \leq k\}$, define

$$q_a^* = \min_{v \in \mathcal{C}_k} q_a(v), \qquad q_a(v) = \tfrac{1}{2} v^\top \widehat{\Sigma} v - a^\top v. \tag{5}$$

For any nonzero $v \geq \mathbf{0}$ with $a^\top v > 0$, optimization along its ray gives

$$\min_{t \geq 0} q_a(tv) = -\frac{1}{2} \frac{(a^\top v)^2}{v^\top \widehat{\Sigma} v} = -\tfrac{1}{2} S(v; a, \widehat{\Sigma})^2. \tag{6}$$

Thus $q_a^* = -\{\max(S_k^*(a, \widehat{\Sigma}), 0)\}^2/2$. For nonzero clipped signals, the maximum is positive and the quadratic recovers the same optimal direction.

Exact allocation on a face. Set subscripts restrict vectors to the indicated coordinates. For an asset set F, let $\Delta_F = \{w \geq \mathbf{0} : \mathbf{1}^\top w = 1,\ w_{F^c} = 0\}$, so $\mathcal{W}_k = \bigcup_{1 \leq |F| \leq k} \Delta_F$. The allocation map is

$$v^F(a, \widehat{\Sigma}) = \arg\min_{v \geq \mathbf{0},\, v_{F^c} = 0} q_a(v), \qquad w(a; F, \widehat{\Sigma}) = \frac{v^F(a, \widehat{\Sigma})}{\mathbf{1}^\top v^F(a, \widehat{\Sigma})} \quad \text{if } v^F(a, \widehat{\Sigma}) \neq \mathbf{0}. \tag{7}$$

Strict convexity makes v^F unique. Its normalization maximizes the Sharpe ratio on Δ_F whenever that face has positive value. At zero valuation, we use uniform weights on F. An empty face uses uniform weights on the first k asset labels under a fixed ordering.

Let $H \subseteq F$ be the positive support of v^F. On a neighborhood where H is unchanged,

$$v_H = \widehat{\Sigma}_{HH}^{-1} a_H, \qquad w_H = \frac{v_H}{\mathbf{1}^\top v_H}, \qquad S_H^2 = a_H^\top v_H, \qquad w_{H^c} = 0. \tag{8}$$

Writing $b_H = \mathbf{1}^\top v_H$ and differentiating the exact allocation gives

$$\frac{\partial w_H}{\partial a_H} = \frac{(I - w_H \mathbf{1}^\top) \widehat{\Sigma}_{HH}^{-1}}{b_H}, \qquad \nabla_{a_H} \ell_H = -\frac{\widehat{\Sigma}_{HH}^{-1} (I - \mathbf{1} w_H^\top) y_H}{b_H}. \tag{9}$$

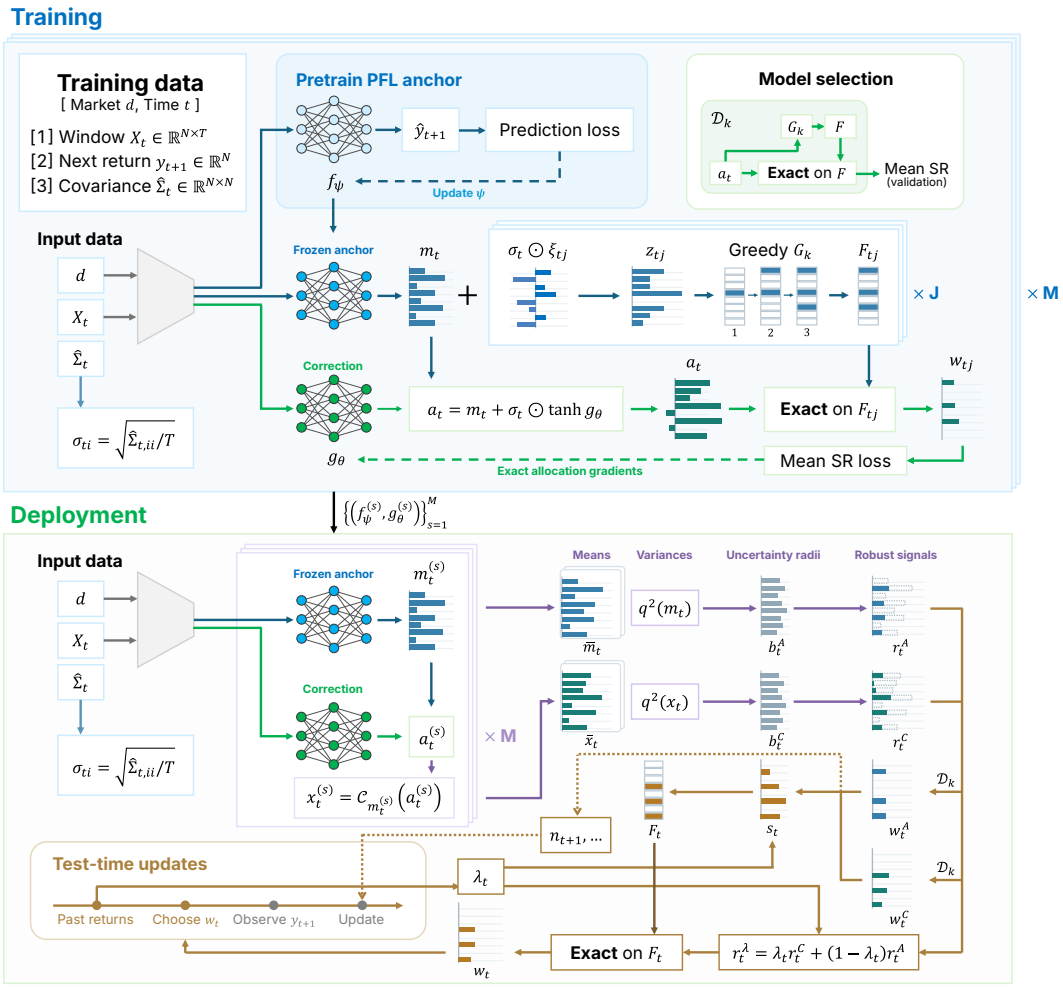


Figure 1: Overview of RA-STPO, illustrated with $N = 8$ and $k = 3$. **Training (top).** Greedy asset selection and exact face allocation provide sparse portfolios and local gradients (Section 3.1). Each of M anchor–corrector pairs learns bounded corrections to a frozen PFL anchor by averaging the Sharpe ratio (SR) losses of J sparse paths (Section 3.2). **Deployment (bottom).** Calibration, member aggregation, and uncertainty adjustment produce robust mean signals (Section 3.3). Past component performance determines λ_t for asset selection and signal blending before exact allocation. Return statistics are updated after observing y_{t+1} for the next decision (Section 3.4).

Here, $\ell_H = -y_H^\top w_H$ is the realized loss. Derivatives outside H vanish, and the clipping Jacobian is applied when a comes from a clipped network output. This Jacobian enables backpropagation through sampled sparse allocations to train the correction network in Section 3.2.

Greedy asset selection using residual-alpha scores. Exact allocation on a fixed face is tractable, but searching over all k-asset faces is combinatorial. We therefore select faces by nonnegative greedy pursuit (Yaghoobi et al., 2015) and analyze residual gains using restricted curvature (Elenberg et al., 2018). For a nonnegative signal z used for asset selection, we first standardize each asset by its estimated volatility:

$$D = \mathrm{diag}(\sqrt{\widehat{\Sigma}_{11}}, \ldots, \sqrt{\widehat{\Sigma}_{NN}}), \quad C = D^{-1}\widehat{\Sigma}D^{-1}, \quad \zeta = D^{-1}z. \tag{10}$$

Here, C is the correlation matrix and ζ_i is the signal per unit of marginal volatility. This scaling preserves asset indices, while C retains the correlations used to optimize the allocation jointly. For a candidate set A, define its exact face value by

$$\Phi_z(A) = \max_{u \geq \mathbf{0},\, \mathrm{supp}(u) \subseteq A} \{2\zeta^\top u - u^\top C u\} = S_A(z, \widehat{\Sigma})^2. \tag{11}$$

Here, S_A is the optimal face Sharpe ratio, set to zero without a positive-return direction. Let u^A be the zero-extended optimizer and $f(u) = 2\zeta^\top u - u^\top C u$. The residual $\alpha^A = \zeta - Cu^A = \frac{1}{2}\nabla f(u^A)$ measures the uncaptured marginal value. The KKT conditions give $\alpha_i^A = 0$ when $u_i^A > 0$, while $C_{ii} = 1$ gives each excluded asset the exact coordinate gain

$$f(u^A + te_i) - f(u^A) = 2t\alpha_i^A - t^2, \qquad \max_{t\geq 0}\{2t\alpha_i^A - t^2\} = (\alpha_i^{A,+})^2, \tag{12}$$

Here, $\alpha_i^{A,+} = \max\{\alpha_i^A, 0\}$, so a positive residual guarantees improvement before the weights are re-optimized. The following theorem gives a greedy approximation guarantee and bounds the remaining selection gain using $\gamma_{2k} = \min_{U\subseteq[N],\, 1\leq|U|\leq 2k} \lambda_{\min}(C_{UU})$, the minimum curvature over unions of two k-asset faces, used only in the analysis.

Theorem 3.1 (Greedy approximation and residual upper bound)**.** *Let $C \succ 0$. Starting at $A_0 = \emptyset$, add at most k unused indices, each maximizing $\alpha_i^{A_j,+}$, and stop early if the maximum is zero. If A^* maximizes Φ_z over $|A| \leq k$, the returned set A_g satisfies*

$$\Phi_z(A_g) \geq (1 - e^{-\gamma_{2k}})\Phi_z(A^*). \tag{13}$$

For any $|A| \leq k$ and any known $0 < \underline{\gamma} \leq \gamma_{2k}$,

$$\Phi_z(A^*) \leq U_z(A) := \Phi_z(A) + \frac{1}{\underline{\gamma}}\sum_{\ell=1}^{k} r_{(\ell)}, \qquad r_i = (\alpha_i^{A,+})^2 \mathbf{1}\{i \notin A\}, \tag{14}$$

where $r_{(1)} \geq r_{(2)} \geq \cdots$ are the ordered residual scores.

The approximation weakens as restricted curvature decreases, while $U_z(A) - \Phi_z(A)$ bounds the remaining selection gap. For $\widehat{\Sigma} = (1-\delta)\widehat{S} + \delta\tau I$, where $0 < \delta \leq 1$, $\widehat{S} \succeq 0$, and $\tau = \mathrm{tr}(\widehat{S})/N > 0$, $\underline{\gamma} = \delta\tau / \max_i \widehat{\Sigma}_{ii}$ is a valid lower bound for every principal block of C and makes $U_z(A)$ computable. Appendix B describes an independent exact search for optimality certification.

Let $G_k(z, \widehat{\Sigma})$ denote this greedy set, resolving equal scores by a fixed asset order. Combining coordinatewise clipping $z_+ = \max(z, 0)$ with exact face allocation defines

$$\mathcal{D}_k(z, \widehat{\Sigma}) = w\Big(z_+; G_k(z_+, \widehat{\Sigma}), \widehat{\Sigma}\Big). \tag{15}$$

Both G_k and w operate in asset coordinates. These operators provide sparse training allocations in Section 3.2 and the component portfolios used for deployment in Section 3.4.

3.2 Learning bounded corrections through sparse paths

We now use these sparse allocations to learn bounded corrections to a frozen predictive anchor. Proposition 2.2 establishes scale invariance for the optimal tangency decision. The following result shows that our greedy decision map inherits this property and identifies its local gradient consequence.

Theorem 3.2 (Scale invariance and radial orthogonality)**.** *For fixed $\widehat{\Sigma} \succ 0$ and index ordering, any $z \in \mathbb{R}^N$, and every $c > 0$,*

$$G_k((cz)_+, \widehat{\Sigma}) = G_k(z_+, \widehat{\Sigma}), \qquad \mathcal{D}_k(cz, \widehat{\Sigma}) = \mathcal{D}_k(z, \widehat{\Sigma}). \tag{16}$$

For locally fixed support H with $\mathbf{1}^\top v_H > 0$, $J_H = \partial w_H / \partial a_H$ and $\ell_H(a) = -y_H^\top w_H(a)$ satisfy

$$J_H a_H = 0, \qquad a_H^\top \nabla_{a_H} \ell_H(a) = 0. \tag{17}$$

An allocation-based loss therefore leaves absolute forecast scale unidentified. To make correction magnitudes explicit in return units, we use a fixed predictive reference for constrained decision-focused correction (Yang et al., 2025), with asset-wise limits determined by measured window variability. We train the predictor using squared error on standardized next-day returns and map its outputs back to return units. We write $m_t = f_\psi(X_t)$ for this complete prediction map and freeze its parameters before learning the correction.

For serially uncorrelated returns with common covariance Σ, the sample-mean covariance is Σ/T. We use the diagonal of its plug-in estimate to set the correction scale for each asset.

$$\Omega_t = \frac{1}{T}\operatorname{diag}(\widehat{\Sigma}_{t,11}, \ldots, \widehat{\Sigma}_{t,NN}), \quad e_t \sim \mathcal{Q}_t = \mathcal{N}(\mathbf{0}, \Omega_t), \quad \sigma_{ti} = \sqrt{\mathbb{E}_{\mathcal{Q}_t} e_{ti}^2} = \sqrt{\widehat{\Sigma}_{t,ii}/T}. \tag{18}$$

The decision network produces

$$a_{\theta t} = m_t + \sigma_t \odot \tanh g_\theta(X_t). \tag{19}$$

The bound $|a_{\theta t} - m_t| \le \sigma_t$ specifies how far each asset's signal may move from its predictive reference. Decision loss selects useful changes within this range, without a coefficient balancing prediction and decision losses.

Fresh exploration and sparse-path losses. Using a single asset set selected from the anchor would restrict training to that set. At the beginning of each epoch e, we draw new perturbations for every training date and replace all candidate faces:

$$\eta_{tj}^{(e)} = \sigma_t \odot \xi_{tj}^{(e)}, \qquad F_{tj}^{(e)} = G_k((m_t + \eta_{tj}^{(e)})_+, \widehat{\Sigma}_t), \qquad j = 1, \ldots, J, \tag{20}$$

where the $\xi_{tj}^{(e)}$ are independent standard Gaussian vectors. Exploration and bounded correction share a scale. The frozen anchor makes the sampling distribution independent of correction parameters, while its sampled faces are redrawn each epoch. The perturbations are used only for face selection. Allocation on each sampled face uses the clipped corrected signal $(a_{\theta t})_+$.

For each j, the sampled allocations define a return path over a minibatch $\mathcal{B}$. We evaluate each path's Sharpe ratio loss before averaging, suppressing the epoch index below:

$$\begin{aligned} r_{tj}(\theta) &= y_{t+1}^\top w((a_{\theta t})_+; F_{tj}, \widehat{\Sigma}_t), \qquad \bar{r}_{\mathcal{B},j} = \frac{1}{|\mathcal{B}|}\sum_{t\in\mathcal{B}} r_{tj}(\theta), \\ s_{\mathcal{B},j}^2 &= \frac{1}{|\mathcal{B}|-1}\sum_{t\in\mathcal{B}}\{r_{tj}(\theta) - \bar{r}_{\mathcal{B},j}\}^2, \qquad \mathcal{L}_{\mathcal{B}}(\theta) = -\frac{1}{J}\sum_{j=1}^{J}\frac{\bar{r}_{\mathcal{B},j}}{\max(s_{\mathcal{B},j}, \epsilon)}. \end{aligned} \tag{21}$$

Here, $|\mathcal{B}| \ge 2$ and $\epsilon > 0$ is a numerical safeguard. Each Sharpe ratio term evaluates a path holding at most k assets per date. Their average is a Monte Carlo estimate of the expected batch loss under the anchor-based sampling distribution. The following theorem establishes sparse feasibility and exact local differentiation of this loss.

Theorem 3.3 (Feasibility and exact batch-objective differentiation)**.** *Condition on the anchors, covariances, draws in* (20)*, and dropout masks.*

(i) For every t, j, $w((a_{\theta t})_+; F_{tj}, \widehat{\Sigma}_t) \in \mathcal{W}_k$.

(ii) For every $w \in \mathcal{W}_k$, $\left|(a_{\theta t} - m_t)^\top w\right| \le \sigma_t^\top w \le \|\sigma_t\|_\infty$.

(iii) Where the network is differentiable, clipping signs and allocation supports are locally fixed, and $s_{\mathcal{B},j} \neq \epsilon$ for every j, backpropagation through (9) *computes $\nabla_\theta \mathcal{L}_{\mathcal{B}}$ without a selection-gradient term. Locally constant fallback allocations contribute zero derivatives.*

Thus each training path remains k-sparse, corrections to predicted portfolio returns are explicitly bounded, and gradients through sampled allocations are locally exact without differentiating discrete asset selection. Algorithm 1 summarizes training. Checkpoints maximize mean validation Sharpe ratio of the unperturbed single-face portfolios $\mathcal{D}_k(a_{\theta t}, \widehat{\Sigma}_t)$.

3.3 Calibration and uncertainty-adjusted mean signals

Deployment averages the learned signals and adjusts their means for uncertainty. Their scale therefore matters even when nominal allocations agree: for fixed $b \ge \mathbf{0}$ and $c > 0$, scale invariance gives $\mathcal{D}_k((ca - b)_+, \widehat{\Sigma}) = \mathcal{D}_k((a - b/c)_+, \widehat{\Sigma})$. We first give each corrected signal an anchor-based level and dispersion, then estimate the variability of the means to be allocated.

Algorithm 1 RA-STPO training

Input: training/validation examples $(X_t, \widehat{\Sigma}_t, y_{t+1})$, k, J, M
$\sigma_t \leftarrow \sqrt{\mathrm{diag}(\widehat{\Sigma}_t)/T}$ for every example
for each member $s = 1, \ldots, M$ **do**
    train f_{ψ_s} with prediction loss, compute $m_t^{(s)}$, and freeze ψ_s
    initialize θ_s and evaluate the initial validation checkpoint
    for each epoch **do**
        replace all faces using new draws in (20)
        for each minibatch $\mathcal{B}$ **do**
            compute $a_t^{(s)}$ by (19) and each sparse return r_{tj}
            compute each path's Sharpe ratio loss, average by (21), and update θ_s
        end for
        evaluate single-face validation Sharpe ratio and retain the best checkpoint
    end for
end for
return $\{(\psi_s, \theta_s)\}_{s=1}^{M}$

For member $s = 1, \ldots, M$, let $x_t^{(s)} = \mathcal{C}_{m_t^{(s)}}(a_t^{(s)})$ be the cross-sectional moment match

$$x_t^{(s)} = \mathrm{mean}_i(m_t^{(s)})\mathbf{1} + \mathrm{sd}_i(m_t^{(s)}) \frac{a_t^{(s)} - \mathrm{mean}_i(a_t^{(s)})\mathbf{1}}{\mathrm{sd}_i(a_t^{(s)})}. \tag{22}$$

Here, mean_i and sd_i use divisor N. A zero denominator gives $x_t^{(s)} = \mathrm{mean}_i(m_t^{(s)})\mathbf{1}$. This rule satisfies $\mathcal{C}_m(ca) = \mathcal{C}_m(a)$ for $c > 0$. It takes the level and dispersion from the predictor while preserving corrected asset rankings for nonconstant cross-sections. Calibration can exceed coordinatewise training bounds, but its total displacement remains controlled (Lemma C.4).

After calibration, we form two ensemble means for deployment, one from the calibrated forecasts and one from the anchors. For either ensemble $v_t = (v_t^{(s)})_{s=1}^{M}$, write $\bar{v}_t = M^{-1} \sum_{s=1}^{M} v_t^{(s)}$ for its mean. Let $q^2(v_t) = (q_i^2(v_t))_{i=1}^{N}$, where $q_i^2(v_t)$ is the sample variance of asset i across members, using divisor $M - 1$. Set $q^2(v_t) = \mathbf{0}$ when $M = 1$.

Conditional on the data, $q^2(v_t)/M$ estimates the variance of the ensemble mean under independent training randomizations. We combine this with window variability in the Gaussian reference variance $\sigma_t^2 + q^2(v_t)/M$. The corresponding mean absolute deviations define the asset-wise uncertainty radii

$$b_t^C = \sqrt{2/\pi}\sqrt{\sigma_t^2 + q^2(x_t)/M}, \qquad b_t^A = \sqrt{2/\pi}\sqrt{\sigma_t^2 + q^2(m_t)/M}. \tag{23}$$

Powers and square roots are coordinatewise. The factor $\sqrt{2/\pi} = \mathbb{E}|Z|$ for $Z \sim \mathcal{N}(0,1)$ converts standard deviation to mean absolute deviation, without a tuned uncertainty multiplier.

To make uncertainty affect asset choice, its penalty must vary across portfolios. The following proposition identifies this penalty for an ellipsoidal uncertainty set.

Proposition 3.4. *For $V \succ 0$, $\widehat{\Sigma} \succ 0$, $\rho \geq 0$, and any nonzero portfolio w,*

$$\min_{\|V^{-1/2}(u-c)\|_2 \leq \rho} S(w; u, \widehat{\Sigma}) = S(w; c, \widehat{\Sigma}) - \rho \frac{\sqrt{w^\top V w}}{\sqrt{w^\top \widehat{\Sigma} w}}. \tag{24}$$

For $V = \widehat{\Sigma}/T$, every portfolio receives the same discount $\rho/\sqrt{T}$, leaving the optimizer unchanged. In contrast, a box uncertainty set applies asset-wise penalties that can change the selected portfolio.

For any center $c_t \geq \mathbf{0}$ and radius $b_t \geq \mathbf{0}$, the following theorem identifies the worst-case signal and bounds the nominal-center Sharpe loss for the box

$$\mathcal{U}_+(c_t, b_t) = \{u \geq \mathbf{0} : |u_i - c_{ti}| \leq b_{ti}\}, \qquad r_t = (c_t - b_t)_+. \tag{25}$$

Theorem 3.5 (Measured robust shift and nominal-center bound)**.** *For every $w \in \mathcal{W}_k$,*

$$\min_{u \in \mathcal{U}_+(c_t, b_t)} S(w; u, \widehat{\Sigma}_t) = S(w; r_t, \widehat{\Sigma}_t). \tag{26}$$

For $|F| \le k$ and $S_F(r_t, \widehat{\Sigma}_t) > 0$, define the selection gap $\varepsilon_{\mathrm{loc}} = S_k^*(r_t, \widehat{\Sigma}_t) - S_F(r_t, \widehat{\Sigma}_t)$ and its residual upper bound $\varepsilon_{\mathrm{cert}} = \sqrt{U_{r_t}(F)} - S_F(r_t, \widehat{\Sigma}_t)$. Then

$$S_k^*(c_t, \widehat{\Sigma}_t) - S(w(r_t; F, \widehat{\Sigma}_t); c_t, \widehat{\Sigma}_t) \le \varepsilon_{\mathrm{loc}} + 2\|\widehat{\Sigma}_t^{-1/2}(c_t - r_t)\|_2$$
$$\le \varepsilon_{\mathrm{cert}} + 2\sqrt{\lambda_{\max}(\widehat{\Sigma}_t^{-1})}\,\|b_t\|_2. \tag{27}$$

Theorem 3.5 therefore reduces robust optimization to nominal optimization at the lower signal r_t and bounds the nominal-center Sharpe gap using the residual bound from Section 3.1 and the size of the uncertainty adjustment. For centers $(\bar{x}_t)_+$ and $(\bar{m}_t)_+$, the robust signals are

$$r_t^C = (\bar{x}_t - b_t^C)_+, \qquad r_t^A = (\bar{m}_t - b_t^A)_+. \tag{28}$$

3.4 Performance-adaptive sparse allocation

The usefulness of the correction can change after training. We construct portfolios from the two robust signals in Section 3.3 and use their past relative performance to determine the correction's contribution to one final sparse portfolio. The component allocations are

$$w_t^C = \mathcal{D}_k(r_t^C, \widehat{\Sigma}_t), \qquad w_t^A = \mathcal{D}_k(r_t^A, \widehat{\Sigma}_t). \tag{29}$$

Credibility from realized performance. We initialize correction credibility using the risk-normalized return advantage of the corrected portfolio over the anchor portfolio during validation. The component returns are $R_{e,t} = y_{t+1}^\top w_t^e$ for $e \in \{C, A\}$. On n_0 validation dates, $s_{e,v}$ denotes the sample standard deviation of these returns, floored at $\epsilon_s > 0$. The risk-normalized advantage is $u_j^v = R_{C,j}^v / s_{C,v} - R_{A,j}^v / s_{A,v}$. We denote its mean by μ_0 and its sample variance, floored at $\epsilon_u > 0$, by s_*^2. After n_t deployment observations, we compute the risk scales from past returns. Once y_{t+1} is observed, we use these scales to evaluate the realized advantage:

$$\nu_{e,t}^2 = \frac{n_0 s_{e,v}^2 + \sum_{\tau<t} R_{e,\tau}^2}{n_0 + n_t}, \qquad u_t = \frac{R_{C,t}}{\nu_{C,t}} - \frac{R_{A,t}}{\nu_{A,t}}, \quad e \in \{C, A\}. \tag{30}$$

As deployment observations accumulate, we discount the validation prior by its share $n_0/(n_0 + n_t)$. Combining its effective count $n_{0,t}$ with observed advantages gives

$$n_{0,t} = \frac{n_0}{n_0 + n_t} n_0, \qquad \widehat{\mu}_t = \frac{n_{0,t}\mu_0 + \sum_{\tau<t} u_\tau}{n_{0,t} + n_t}, \qquad \lambda_t = \Phi_{\mathrm{std}}\left(\frac{\widehat{\mu}_t}{\sqrt{s_*^2/(n_{0,t} + n_t)}}\right). \tag{31}$$

Here, Φ_{std} is the standard normal CDF. The credibility weight λ_t is the probability of positive mean advantage under the working Gaussian distribution $\mathcal{N}(\widehat{\mu}_t, s_*^2/(n_{0,t} + n_t))$. We compute λ_t before observing y_{t+1}. After observing this return, we add $R_{e,t}^2$ and u_t to their accumulated sums and increment n_t for the next decision.

One sparse allocation. We normalize each component by its largest weight so that its maximum contribution to the selection score is λ_t or $1 - \lambda_t$:

$$s_{ti} = \lambda_t \frac{w_{ti}^C}{\max_j w_{tj}^C} + (1 - \lambda_t)\frac{w_{ti}^A}{\max_j w_{tj}^A}, \qquad F_t = \mathrm{TopK}_k(s_t). \tag{32}$$

We use the same credibility weight to blend the conservative signals and allocate to F_t:

$$r_t^\lambda = \lambda_t r_t^C + (1 - \lambda_t) r_t^A, \qquad w_t = w(r_t^\lambda; F_t, \widehat{\Sigma}_t) \in \mathcal{W}_k. \tag{33}$$

Thus credibility controls both asset selection and the signal used for exact allocation.

The signal blend also has a box-uncertainty interpretation (Remark C.5). Algorithm 2 summarizes deployment: two component allocations and one final face solve.

Regularization alignment. RA-STPO makes the form, scale, and use of regularization explicit. PFL supplies a predictive reference, while DF-STPO couples prediction and decision losses. RA-STPO bounds departures from a frozen reference and evaluates them on sparse return paths. Calibration carries the anchor's level and dispersion into signal averaging, uncertainty adjusts the deployed means, and past relative performance governs the correction's use in the final sparse allocation. These roles connect decision-focused correction and robust deployment through quantities that can be interpreted at each stage.

Algorithm 2 RA-STPO deployment

1: **Input:** window X_t, covariance $\widehat{\Sigma}_t$, trained members, k, prior and past return state
2: compute anchors $m_t^{(s)}$ and bounded corrections $a_t^{(s)}$ by (19)
3: calibrate all members by (22) and compute b_t^C, b_t^A by (23)
4: form robust signals by (28) and component allocations by (29)
5: compute λ_t from the stored state using (31)
6: select F_t from normalized component scores using (32)
7: compute r_t^λ and solve $w_t = w(r_t^\lambda; F_t, \widehat{\Sigma}_t)$ by (33)
8: **Output:** w_t for the current rebalance
9: after observing y_{t+1}, update component-return state for the next decision by (30)

4 EXPERIMENTS

We compare four sparse optimizers with historical or PFL signals, plus DF-STPO and RA-STPO, using common windows, covariances, and chronological splits. The cardinality limit is $k = 10$, and each input window contains $T = 100$ daily returns. Learned methods share architecture and inputs. OSCAR, SD-relaxation, and DF-STPO use a common long-only face solve. The other baselines use the allocations returned by their original algorithms. Appendices A and D give the protocol, with additional experiments in Appendix E.

Table 1: Annualized Sharpe ratio (SR) and terminal wealth (W from one unit). Learned ensembles use 32 networks. Best values are bold and underlined, and second-best values are bold, based on unrounded results.

	EuroStoxx50		FTSE100		KOSPI200		Nikkei225		S&P 500	
signal / rule	SR	W	SR	W	SR	W	SR	W	SR	W
Historic										
OSCAR	1.874	1.794	1.239	1.491	0.116	0.961	0.261	1.062	**2.444**	**3.315**
SD-relaxation	2.015	1.847	1.124	1.407	0.989	1.651	0.881	1.465	1.267	1.544
mSSRM-PGA	2.078	2.022	1.716	1.715	0.189	1.006	1.137	1.801	1.442	2.013
ASMP-AFBA	2.079	1.924	1.636	1.658	0.297	1.082	0.540	1.198	1.447	1.619
Prediction-focused learning										
OSCAR	2.707	2.328	2.361	**2.157**	1.772	2.463	**1.740**	**2.786**	1.807	2.207
SD-relaxation	**2.771**	**2.357**	**2.533**	**2.201**	1.911	2.483	1.124	1.894	1.965	2.209
mSSRM-PGA	2.601	2.258	2.204	2.004	1.386	2.022	1.303	2.024	1.769	**2.237**
ASMP-AFBA	2.457	2.167	2.300	2.022	2.232	2.318	0.795	1.341	1.717	1.685
Decision-focused learning										
DF-STPO	2.128	1.910	1.634	1.637	**2.558**	**3.874**	1.182	1.821	1.628	2.005
RA-STPO (Ours)	**2.837**	**2.512**	**2.402**	2.013	**2.611**	**3.498**	**1.543**	**2.322**	**2.392**	2.233

PFL and DF-STPO allocate from the mean of 32 forecasts, while RA-STPO uses 16 anchor–corrector pairs. OSCAR and DF-STPO use each dataset's fixed ticker-sorted asset order at evaluation.

RA-STPO has the highest five-market mean SR (2.357) and terminal wealth (2.515) among the methods in Table 1. Under the same deployment rule, decision-focused correction raises these means from 2.046 and 2.037, respectively (Appendix E.6). Resampling trained ensemble members yields lower variability in both metrics for RA-STPO than for DF-STPO in all five markets (Appendix E.7).

5 CONCLUSION

RA-STPO connects decision-focused forecast correction with robust sparse allocation by making the reference, scale, and use of regularization explicit. Corrections are bounded around a predictive anchor and learned from resampled sparse paths, while deployment uses uncertainty-adjusted means and observed relative performance. The evaluation uses current index constituents and excludes trading costs. Further work should incorporate historical index membership, account for turnover and trading costs, and calibrate joint forecast uncertainty.

AI USE STATEMENT

We used large language models to assist with refining the wording of the manuscript, finding and examining related literature, developing mathematical proofs, and writing experimental code. The authors reviewed the AI-assisted material, checked all references against the original sources, verified the mathematical arguments under the stated assumptions, and reviewed and validated the implementation. The authors made the final decisions on the methodology, interpretation of results, and presentation, and take full responsibility for the manuscript, mathematical claims, code, and reported results.

ETHICS STATEMENT

This work studies portfolio optimization using publicly available historical market data and does not involve human participants or personal data. The experiments provide methodological comparisons under the limitations described in the paper, including survivorship bias and the exclusion of transaction costs. The reported results do not constitute investment advice or guarantees of future performance.

REPRODUCIBILITY STATEMENT

Proofs and supplementary analysis are provided in Appendix C, and experimental details, including data preprocessing, chronological splits, model settings, and evaluation procedures, are given in Appendix D. The code is available at `https://anonymous.4open.science/r/RA-STPO-A210` and includes the proposed method, baseline implementations, data preparation scripts, dependency versions, fixed experimental settings and random seeds, and instructions for reproducing the experiments. Historical prices are obtained from Yahoo Finance, and subsequent downloads may differ from the original data snapshot.

REFERENCES

Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J. Zico Kolter. Differentiable convex optimization layers. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL `https://proceedings.neurips.cc/paper/2019/hash/9ce3c52fc54362e22053399d3181c638-Abstract.html`.

Brandon Amos and J. Zico Kolter. OptNet: Differentiable optimization as a layer in neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 136–145, 2017. URL `https://proceedings.mlr.press/v70/amos17a.html`.

David L. Applegate, William Cook, Sanjeeb Dash, and Daniel G. Espinoza. Exact solutions to linear programming problems. *Operations Research Letters*, 35(6):693–699, 2007. doi: 10.1016/j.orl.2006.12.010.

Hyunglip Bae, Haeun Jeon, Minsu Park, Yongjae Lee, and Woo Chang Kim. A Cholesky decomposition-based asset selection heuristic for sparse tangent portfolio optimization, 2025. URL `https://arxiv.org/abs/2502.11701`.

Quentin Berthet, Mathieu Blondel, Olivier Teboul, Marco Cuturi, Jean-Philippe Vert, and Francis R. Bach. Learning with differentiable perturbed optimizers. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL `https://proceedings.neurips.cc/paper/2020/hash/6bb56208f672af0dd65451f869fedfd9-Abstract.html`.

Dimitris Bertsimas and Ryan Cory-Wright. A scalable algorithm for sparse portfolio selection. *INFORMS Journal on Computing*, 34(3):1489–1511, 2022. doi: 10.1287/ijoc.2021.1127.

Michael J. Best and Robert R. Grauer. On the sensitivity of mean-variance-efficient portfolios to changes in asset means: Some analytical and computational results. *The Review of Financial Studies*, 4(2):315–342, 1991. doi: 10.1093/rfs/4.2.315.

Daniel Bienstock. Computational study of a family of mixed-integer quadratic programming problems. *Mathematical Programming*, 74(2):121–140, 1996. doi: 10.1007/BF02592208.

Joshua Brodie, Ingrid Daubechies, Christine De Mol, Domenico Giannone, and Ignace Loris. Sparse and stable Markowitz portfolios. *Proceedings of the National Academy of Sciences*, 106(30): 12267–12272, 2009. doi: 10.1073/pnas.0904287106.

Vijay K. Chopra and William T. Ziemba. The effect of errors in means, variances, and covariances on optimal portfolio choice. *The Journal of Portfolio Management*, 19(2):6–11, 1993. URL https://www.pm-research.com/content/iijpormgmt/19/2/6.

William J. Cook, Thorsten Koch, Daniel E. Steffy, and Kati Wolter. A hybrid branch-and-bound approach for exact rational mixed-integer programming. *Mathematical Programming Computation*, 5(3):305–344, 2013. doi: 10.1007/s12532-013-0055-6.

Abhimanyu Das and David Kempe. Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection. In *Proceedings of the 28th International Conference on Machine Learning*, pp. 1057–1064, 2011. URL https://arxiv.org/abs/1102.3975.

Victor DeMiguel, Lorenzo Garlappi, Francisco J. Nogales, and Raman Uppal. A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms. *Management Science*, 55(5):798–812, 2009a. doi: 10.1287/mnsc.1080.0986.

Victor DeMiguel, Lorenzo Garlappi, and Raman Uppal. Optimal versus naive diversification: How inefficient is the $1/N$ portfolio strategy? *The Review of Financial Studies*, 22(5):1915–1953, 2009b. doi: 10.1093/rfs/hhm075.

Bradley Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1): 1–26, 1979. doi: 10.1214/aos/1176344552.

Ethan R. Elenberg, Rajiv Khanna, Alexandros G. Dimakis, and Sahand Negahban. Restricted strong convexity implies weak submodularity. *The Annals of Statistics*, 46(6B):3539–3568, 2018. doi: 10.1214/17-AOS1679. URL https://doi.org/10.1214/17-AOS1679.

Adam N. Elmachtoub and Paul Grigas. Smart "predict, then optimize". *Management Science*, 68(1): 9–26, 2022. doi: 10.1287/mnsc.2020.3922.

David Gale. *The Theory of Linear Economic Models*. McGraw–Hill, 1960. URL https://books.google.com/books?id=E-C8AAAAIAAJ.

Jianjun Gao and Duan Li. Optimal cardinality constrained portfolio selection. *Operations Research*, 61(3):745–761, 2013. doi: 10.1287/opre.2013.1170.

Ambros M. Gleixner, Daniel E. Steffy, and Kati Wolter. Iterative refinement for linear programming. *INFORMS Journal on Computing*, 28(3):449–464, 2016. doi: 10.1287/ijoc.2016.0692.

David Goldberg. What every computer scientist should know about floating-point arithmetic. *ACM Computing Surveys*, 23(1):5–48, 1991. doi: 10.1145/103162.103163.

Donald Goldfarb and Garud Iyengar. Robust portfolio selection problems. *Mathematics of Operations Research*, 28(1):1–38, 2003. doi: 10.1287/moor.28.1.1.14260.

Ravi Jagannathan and Tongshu Ma. Risk reduction in large portfolios: Why imposing the wrong constraints helps. *The Journal of Finance*, 58(4):1651–1683, 2003. doi: 10.1111/1540-6261.00580.

Christian Jansson. Rigorous lower and upper bounds in linear programming. *SIAM Journal on Optimization*, 14(3):914–935, 2004. doi: 10.1137/S1052623402416839.

Haeun Jeon, Seunghoon Choi, Hyunglip Bae, Yongjae Lee, and Woo Chang Kim. Decision-focused sparse tangent portfolio optimization. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=KV7XHF0IbK.

Philippe Jorion. Bayes–Stein estimation for portfolio analysis. *Journal of Financial and Quantitative Analysis*, 21(3):279–292, 1986. doi: 10.2307/2331042.

Min Jeong Kim, Yongjae Lee, Jang Ho Kim, and Woo Chang Kim. Sparse tangent portfolio selection via semi-definite relaxation. *Operations Research Letters*, 44(4):540–543, 2016. doi: 10.1016/j.orl.2016.05.012.

Charles L. Lawson and Richard J. Hanson. *Solving Least Squares Problems*, volume 15 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics, 1995. doi: 10.1137/1.9781611971217. First published by Prentice–Hall in 1974.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004. doi: 10.1016/S0047-259X(03)00096-4.

Yizun Lin, Zhao-Rong Lai, and Cheng Li. A globally optimal portfolio for m-sparse Sharpe ratio maximization. In *Advances in Neural Information Processing Systems*, volume 37, pp. 17133–17160, 2024a. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/1eaa5146756be028ad6fff1efcc8e6bd-Abstract-Conference.html.

Yizun Lin, Yangyu Zhang, Zhao-Rong Lai, and Cheng Li. Autonomous sparse mean-CVaR portfolio optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 30440–30456. PMLR, 2024b. URL https://proceedings.mlr.press/v235/lin24w.html.

Yizun Lin, Linhui Wang, and Zhao-Rong Lai. Autonomous sparse Markowitz portfolio based on two-stage accelerated forward–backward algorithm. *Expert Systems with Applications*, 271:126598, 2025. doi: 10.1016/j.eswa.2025.126598.

Jayanta Mandi, James Kotary, Senne Berden, Maxime Mulamba, Victor Bucarey, Tias Guns, and Ferdinando Fioretto. Decision-focused learning: Foundations, state of the art, benchmark and future opportunities. *Journal of Artificial Intelligence Research*, 80:1623–1701, 2024. doi: 10.1613/jair.1.15320.

Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952. doi: 10.1111/j.1540-6261.1952.tb01525.x.

Richard O. Michaud. The Markowitz optimization enigma: Is "optimized" optimal? *Financial Analysts Journal*, 45(1):31–42, 1989. doi: 10.2469/faj.v45.n1.31.

Siegfried M. Rump. Verification methods: Rigorous results using floating-point arithmetic. *Acta Numerica*, 19:287–449, 2010. doi: 10.1017/S096249291000005X.

John Schulman, Nicolas Heess, Theophane Weber, and Pieter Abbeel. Gradient estimation using stochastic computation graphs. In *Advances in Neural Information Processing Systems*, volume 28, 2015. URL https://proceedings.neurips.cc/paper/2015/hash/de03beffeed9da5f3639a621bcab5dd4-Abstract.html.

Sanket Shah, Kai Wang, Bryan Wilder, Andrew Perrault, and Milind Tambe. Decision-focused learning without decision-making: Learning locally optimized decision losses. In *Advances in Neural Information Processing Systems*, volume 35, pp. 1320–1332, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/0904c7edde20d7134a77fc7f9cd86ea2-Abstract-Conference.html.

William F. Sharpe. Mutual fund performance. *The Journal of Business*, 39(1):119–138, 1966. doi: 10.1086/294846.

Mehrdad Yaghoobi, Di Wu, and Mike E. Davies. Fast non-negative orthogonal matching pursuit. *IEEE Signal Processing Letters*, 22(9):1229–1233, 2015. doi: 10.1109/LSP.2015.2393637. URL https://doi.org/10.1109/LSP.2015.2393637.

Jiaqi Yang, Enming Liang, Zicheng Su, Zhichao Zou, Peng Zhen, Jiecheng Guo, Wanjing Ma, and Kun An. DFF: Decision-focused fine-tuning for smarter predict-then-optimize with limited data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 26868–26876, 2025. doi: 10.1609/aaai.v39i25.34891. URL https://doi.org/10.1609/aaai.v39i25.34891.

Chee-Keng Yap. Towards exact geometric computation. *Computational Geometry*, 7(1–2):3–23, 1997. doi: 10.1016/0925-7721(95)00040-2.

A BACKGROUND: EXISTING METHODS

All rows in Table 1 ultimately return a portfolio in the same set $\mathcal{W}_k$. OSCAR, semidefinite relaxation (SD-relaxation), and DF-STPO allow short sales in their source formulations (Bae et al., 2025; Kim et al., 2016; Jeon et al., 2026). Our comparison retains their selection principles, with the implementation adaptations stated below, and re-optimizes each selected face with

$$\text{FACESOLVE}(\mu, \Sigma, F): \quad u_F \in \arg\min_{u \geq \mathbf{0}} \left\{ \tfrac{1}{2} u^\top \Sigma_{FF} u - \mu_F^\top u \right\}, \qquad w_F = \frac{u_F}{\mathbf{1}^\top u_F}, \qquad w_{F^c} = 0. \tag{34}$$

The normalization applies when the quadratic optimizer is nonzero. With no positive-return direction on a selected face, the implementation uses uniform weights on that face as a feasible fallback, not as an exact Sharpe ratio maximizer. An active-set solve identifies zero coordinates before normalization. This step enforces the same long-only constraint for all methods. mSSRM-PGA and ASMP-AFBA already satisfy this constraint, so we retain the allocations returned by their original algorithms (Lin et al., 2024a; 2025). Appendix E.2 separately reports the unrestricted-sign counterparts of the three converted methods.

A.1 SOURCES OF THE MEAN VECTOR

Historic. The historical-mean baseline uses the arithmetic mean of the rolling window used to estimate the covariance. It requires no prediction model.

Prediction-focused learning. PFL trains the network described in Appendix D.3 with cross-sectional prediction loss. Its forecast is then passed to each sparse solver below.

Decision-focused learning. DF-STPO trains through its differentiable soft selection layer. RA-STPO trains through Algorithm 1. The labels in Table 1 identify both the source of the mean and, for these two rows, the training layer.

A.2 OPTIMIZATION RULES

OSCAR. OSCAR (Bae et al., 2025) uses unrestricted tangency to construct a score in Cholesky coordinates. Algorithm 3 retains that score and applies the common long-only face solve. The ordering is explicit because the score is not permutation equivariant.

Algorithm 3 OSCAR with common long-only re-optimization

1: **Input:** (μ, Σ), cardinality limit k, asset ordering π
2: permute (μ, Σ) by π and compute $A^\top A = \Sigma$
3: $z \leftarrow \Sigma^{-1}\mu$ ▷ unrestricted tangency direction
4: $F \leftarrow \text{TopK}_k(|Az|)$ ▷ Cholesky-coordinate score
5: undo π on F
6: **return** FACESOLVE(μ, Σ, F)

SD-relaxation. Kim et al. (Kim et al., 2016) lift $ww^\top$ to a positive-semidefinite matrix. With $M = \mu\mu^\top$, the semidefinite program (SDP) used here is

$$\max_{Y \succeq 0} \ \text{tr}(MY) \quad \text{s.t.} \quad \text{tr}(\Sigma Y) = 1, \qquad \|Y\|_{1,1} \leq k \, \text{tr}(Y). \tag{35}$$

The diagonal of Y is a surrogate for squared asset weights.

Algorithm 4 SD-relaxation with common long-only re-optimization

1: **Input:** (μ, Σ), cardinality limit k
2: solve (35) for Y ▷ convex relaxation
3: $F \leftarrow \text{TopK}_k(\text{diag}\, Y)$ ▷ hard asset face
4: **return** FACESOLVE(μ, Σ, F)

mSSRM-PGA. Lin et al. (Lin et al., 2024a) optimize the homogeneous sparse quadratic with a proximal-gradient step whose proximal map retains the largest positive coordinates. We use the ridge parameter and stopping criterion from the reference implementation.

Algorithm 5 mSSRM-PGA

Input: (μ, Σ), cardinality limit k, ridge $\epsilon = 10^{-3}$
$H \leftarrow \Sigma + \epsilon I$ and $\gamma \leftarrow 0.999/\lambda_{\max}(H)$
$v \leftarrow \mu$
repeat
  $z \leftarrow [v - \gamma(Hv - \mu)]_+$
  retain the k largest entries of z and assign the result to v ▷ long-only hard threshold
until relative iterate change is at most 10^{-6}, or 3000 iterations
return $v/(\mathbf{1}^\top v)$ if its sum is positive; otherwise the single-asset portfolio with the highest Sharpe ratio

ASMP-AFBA. The autonomous sparse mean–variance procedure (Lin et al., 2025) applies an accelerated forward–backward algorithm twice. With $M_2 = \Sigma + \mu\mu^\top$, its first stage jointly updates a portfolio weight vector on the simplex and an auxiliary return parameter within the interval specified by the reference implementation. Its second stage is restricted to the largest first-stage coordinates.

Algorithm 6 ASMP-AFBA

Input: (μ, Σ), cardinality limit k, penalty $\lambda = 6$
$M_2 \leftarrow \Sigma + \mu\mu^\top$
$u \leftarrow \mathrm{AFBA}(M_2, \mu, \lambda, \rho \in [0.01, 0.10])$ ▷ first long-only stage
$F \leftarrow \mathrm{TopK}_k(u)$ ▷ $k = 10$ in this comparison
$u_F \leftarrow \mathrm{AFBA}((M_2)_{FF}, \mu_F, \lambda, \rho \in [0.01, 0.10])$ ▷ restricted second stage
normalize u_F on F, set $u_{F^c} = 0$, and **return** u

A.3 Decision-focused sparse tangent portfolio optimization

We implement DF-STPO from the convex program and Algorithm 1 of Jeon et al. (2026). With $A_t^\top A_t = \widetilde{\Sigma}_t$, its auxiliary solve is

$$(u_t^{(0)}, \tau_t^{(0)}) \in \arg\max_{u,\tau} \ \widehat{\mu}_t^\top u \quad \text{s.t.} \quad \mathbf{1}^\top u = \tau, \quad \|A_t u\|_2 \le 1, \quad \tau \ge 0. \tag{36}$$

Let $h = \widetilde{\Sigma}_t^{-1}\mu$, $e = \widetilde{\Sigma}_t^{-1}\mathbf{1}$, $s = \mathbf{1}^\top h$, and $q = \mathbf{1}^\top e$. Eliminating τ gives the analytic solution

$$d = h - \frac{\min\{s, 0\}}{q} e, \qquad U(\mu) = \frac{d}{\sqrt{d^\top \widetilde{\Sigma}_t d}}. \tag{37}$$

Set $U(\mu) = 0$ at zero value. Training uses raw Cholesky scores and the risk-constrained solution directly, with return-unit prediction error. For the mask in (2), $\partial p/\partial s = \beta\{\mathrm{diag}(v) - vv^\top/(\mathbf{1}^\top v)\}$, where $v = p \odot (1 - p)$. At evaluation, the selected indices enter the common long-only FaceSolve. Numerical settings are given in Appendix D.

Algorithm 7 DF-STPO training

Input: examples $(X_t, \widetilde{\Sigma}_t, y_{t+1})$, cardinality limit k, mixture α
compute Cholesky factors $A_t^\top A_t = \widetilde{\Sigma}_t$ for all examples
for each epoch and minibatch **do**
  $\widehat{\mu}_t \leftarrow f_\theta(X_t)$, $u_t \leftarrow U(\widehat{\mu}_t)$, and $s_t \leftarrow |A_t u_t|$
  $p_t \leftarrow \mathrm{SoftTopK}_\beta(s_t, k)$
  $\mu'_t \leftarrow p_t \odot \widehat{\mu}_t$ and $u'_t \leftarrow U(\mu'_t)$
  $L_{\mathrm{decision}} \leftarrow -y_{t+1}^\top u'_t$ and $L_{\mathrm{prediction}} \leftarrow \|\widehat{\mu}_t - y_{t+1}\|_2^2$
  $L \leftarrow \alpha L_{\mathrm{decision}} + (1 - \alpha) L_{\mathrm{prediction}}$
  update θ through both solves and the mask's implicit Jacobian
end for
return θ

Algorithm 8 DF-STPO evaluation

1: **Input:** X_t, $\widetilde{\Sigma}_t$, parameters θ, cardinality limit k
2: $\widehat{\mu}_t \leftarrow f_\theta(X_t)$ and compute $A_t^\top A_t = \widetilde{\Sigma}_t$
3: $u_t \leftarrow U(\widehat{\mu}_t)$ and $F_t \leftarrow \mathrm{TopK}_k(|A_t u_t|)$ ▷ hard transformed-coordinate support
4: **return** FACESOLVE$(\widehat{\mu}_t, \widetilde{\Sigma}_t, F_t)$ ▷ common long-only solve

B EXACT CERTIFIER FOR SPARSE PORTFOLIO OPTIMIZATION

The certifier verifies global optimality for the empirical problem in (5) when a positive-return direction exists. It is used for verification, separately from the asset-selection routine used during training. It compares bounds in exact arithmetic to distinguish a feasible incumbent from a proven optimum. This appendix also allows positive-semidefinite unshrunk covariances. In that extension, a positive-mean, zero-risk direction is assigned an infinite Sharpe ratio by convention. The positive-definite experimental estimator does not require this branch.

B.1 EXACT ARITHMETIC FOR A STORED RETURN WINDOW

The implementation interprets each stored floating-point return as its exact dyadic rational. The construction also applies to any finite return matrix with rational entries. Choose a common positive denominator Q for the return table $R \in \mathbb{Q}^{T\times N}$ such that $a_{ti} = QR_{ti} \in \mathbb{Z}$, and define

$$s_i = \sum_{t=1}^{T} a_{ti}, \qquad Q_\mu = TQ, \qquad B_{ti} = Ta_{ti} - s_i, \qquad M = B^\top B, \qquad \kappa = (T-1)Q_\mu^2. \tag{38}$$

Then $\mu = s/Q_\mu$ and the unshrunk sample covariance $\widehat{S} = M/\kappa$ hold exactly over the rationals. Covariance shrinkage is also compatible with this model whenever its coefficient and target are rational. The particular experimental estimator is specified separately in Appendix D.1. After shrinkage, M and κ are updated to represent the resulting covariance $\widehat{\Sigma}$ exactly. The original B need not factor this modified covariance. If $\max_i \mu_i \le 0$, the quadratic minimum is zero and does not identify a negative Sharpe ratio optimum. This case is reported separately, without a positive Sharpe ratio certificate. The certificate concerns the exact-arithmetic moment estimator applied to the stored returns. A covariance matrix computed separately in floating-point arithmetic may differ from this exact estimator because of rounding.

Write $q(v) = \frac{1}{2} v^\top \widehat{\Sigma} v - \mu^\top v$, the quadratic in (5) at $a = \mu$. At a branch-and-bound node, let A contain the assets that have not been excluded. Dropping cardinality gives the long-only relaxation

$$p_A = \min\{q(v) : v \ge \mathbf{0},\ \mathrm{supp}(v) \subseteq A\}. \tag{39}$$

An active-set search in rational arithmetic verifies its KKT conditions (Lawson & Hanson, 1995). Writing B_A for the columns indexed by A, the *unshrunk* sample factor gives the equivalent identity

$$p_A = -\frac{T-1}{2} \min_y \{\|y\|_2^2 : B_A^\top y \ge s_A\}, \tag{40}$$

For a modified covariance, the witness is instead checked directly through $v_A \ge 0$, $g_A = \widehat{\Sigma}_{AA} v_A - \mu_A \ge 0$, and $v_A^\top g_A = 0$, with $v_{A^c} = 0$. These conditions certify a global minimum of the convex relaxation without requiring a rational Cholesky factor. A nonnegative null direction d supported on A has positive mean and zero risk exactly when $M_{AA} d_A = 0$ and $s_A^\top d_A > 0$. The theorem of alternatives characterizes finite and improving-ray cases (Gale, 1960). The finite active-set and nullspace-search safeguards may leave a case unresolved, in which case no finite node optimum is asserted.

B.2 SEARCH AND RETURNED CERTIFICATE

A node stores required and excluded indices (I, E). Its relaxation drops the cardinality bound and the requirement that the indices in I be strictly positive. Thus p_A remains a valid lower bound. If the optimizer has at most k nonzero entries, it is feasible for the original sparse problem. Updating the

incumbent to this value allows the node to be pruned, even if the optimizer omits a required index. Otherwise, an undecided positive coordinate generates required and excluded children. The two children cover the parent, and exact lower bounds are compared with the best feasible incumbent. Each live node n retains its inherited lower bound p_n.

Algorithm 9 Exact certification for a stored return window

Input: R, cardinality limit k, rational covariance map, node and time limits
form (s, M, Q_μ, κ) by (38) and apply the covariance map exactly
if no positive-mean direction exists **then**
    return a nonpositive-mean domain status, without a positive Sharpe ratio certificate
end if
seek a rational feasible incumbent q_U on greedy supports, initially $q_U \leftarrow +\infty$
initialize $\mathcal{L} \leftarrow \{(\emptyset, \emptyset)\}$ with lower bound $-\infty$
while $\mathcal{L} \neq \emptyset$ and resources remain **do**
    pop the live node (I, E) with smallest lower bound and set $A \leftarrow [N] \setminus E$
    if $|I| > k$ **then**
        close the infeasible node and **continue**
    end if
    solve (39) exactly, obtaining $(p_A, \bar{v})$ or a null ray d
    if the node calculation is unresolved **then**
        restore the node with its inherited lower bound and **break**
    end if
    if a ray d is returned **then**
        if $|\operatorname{supp}(d)| \leq k$ **then**
            return SPARSE INFINITE CERTIFIED and d
        end if
        choose $j \in \operatorname{supp}(d) \setminus (I \cup E)$
    else
        if $|\operatorname{supp}(\bar{v})| \leq k$ **then**
            update q_U with $q(\bar{v})$, close the node, and **continue**
        else if $p_A \geq q_U$ **then**
            prune the node and **continue**
        end if
        choose $j \in \operatorname{supp}(\bar{v}) \setminus (I \cup E)$
    end if
    push both children with lower bound p_A, or $-\infty$ for a ray node
end while
if $\mathcal{L} = \emptyset$ **then**
    return FINITE CERTIFIED, $q_U = q^*$, and attaining rational weights
else
    return RESOURCE OPEN, the incumbent, and $\min\{q_U, \min_{n \in \mathcal{L}} p_n\}$
end if

A finite optimality certificate contains matching exact bounds and rational weights that attain them. A ray certificate contains the exact direction, including its coefficients. If the search terminates with unresolved nodes, it returns the best available feasible solution and a valid bound, possibly infinite, without certifying optimality. Outward-rounded interval arithmetic is used only to display the final Sharpe ratio interval. This distinction follows standard practice in verified numerical optimization (Goldberg, 1991; Yap, 1997; Applegate et al., 2007; Cook et al., 2013; Gleixner et al., 2016; Rump, 2010; Jansson, 2004). If the cardinality-free long-only solution is already k-sparse, the root relaxation certifies optimality. Otherwise, further search may be required.

C Proofs and Supplementary Analysis

C.1 Proof of Proposition 2.1

Proof. Suppose first that $AE_F = E_{\pi(F)}$ for every F. Apply the identity to a singleton $F = \{j\}$. Column Ae_j must span the coordinate line generated by $e_{\pi(j)}$, so $Ae_j = d_j e_{\pi(j)}$ for a nonzero d_j.

Nonsingularity makes π a permutation. Hence $A = P_\pi D$ for $D = \mathrm{diag}(d_1, \ldots, d_N)$. Conversely, every matrix of this form maps coordinate support subspaces to their permuted counterparts. □

C.2 Proof of Theorem 3.1

Proof. Suppress the signal subscript and write $\Phi(A) = \Phi_z(A)$. Let $f(u) = 2\zeta^\top u - u^\top C u$. The KKT conditions for the exact nonnegative solve on A give

$$\alpha_i^A = 0 \quad \text{if } u_i^A > 0, \qquad \alpha_i^A \le 0 \quad \text{if } i \in A \text{ and } u_i^A = 0. \tag{41}$$

Because $C_{ii} = 1$, the feasible ray in (12) has maximum gain $(\alpha_i^{A,+})^2$ over $t \ge 0$. Re-optimizing on $A \cup \{i\}$ therefore gives

$$\Phi(A \cup \{i\}) - \Phi(A) \ge (\alpha_i^{A,+})^2. \tag{42}$$

For $|A \cup B| \le 2k$, let $\bar{u}$ solve the problem on $A \cup B$ and put $h = \bar{u} - u^A$. Expansion around u^A gives

$$f(\bar{u}) - f(u^A) = 2(\alpha^A)^\top h - h^\top C h. \tag{43}$$

The contribution of coordinates in A is nonpositive by (41). Outside A, $h_i \ge 0$, so negative residuals may be discarded. The vector h is supported on at most $2k$ coordinates. The restricted-curvature bound (Das & Kempe, 2011; Elenberg et al., 2018) and completing the square give

$$f(\bar{u}) - f(u^A) \le 2 \sum_{i \in B \setminus A} \alpha_i^{A,+} h_i - \gamma_{2k} \|h\|_2^2 \le \frac{1}{\gamma_{2k}} \sum_{i \in B \setminus A} (\alpha_i^{A,+})^2. \tag{44}$$

Let A_j be the greedy prefix and $\Delta_j = \Phi(A^*) - \Phi(A_j)$. Monotonicity and the upper inequality applied on $A_j \cup A^*$ imply

$$\Delta_j \le \frac{k}{\gamma_{2k}} \max_{i \notin A_j} (\alpha_i^{A_j,+})^2. \tag{45}$$

The chosen coordinate attains this maximum, and (42) bounds the resulting increase in the objective. Hence $\Delta_{j+1} \le (1 - \gamma_{2k}/k)\Delta_j$. If the rule stops early, every excluded residual is nonpositive, so the KKT conditions hold on the full orthant and the current solution is globally optimal. Otherwise, iteration for k rounds and $(1 - \gamma_{2k}/k)^k \le e^{-\gamma_{2k}}$ prove (13).

For an arbitrary face A, repeat the upper argument without replacing the residual scores by their maximum. The set $A^* \setminus A$ has at most k indices, so its sum is bounded by the k largest excluded scores. Replacing γ_{2k} with a certified lower bound proves (14). □

Remark C.1 (A computable curvature bound)**.** Let $\widehat{\Sigma} = (1 - \delta)\widehat{S} + \delta\tau I$ with $\widehat{S} \succeq 0$, $0 < \delta \le 1$, and $\tau > 0$. For a principal set U and vector x, let $y = D_{UU}^{-1} x$. Then

$$x^\top C_{UU} x = y^\top \widehat{\Sigma}_{UU} y \ge \delta\tau \|y\|_2^2 \ge \frac{\delta\tau}{\max_i \widehat{\Sigma}_{ii}} \|x\|_2^2. \tag{46}$$

This proves the computable lower bound $\underline{\gamma} = \delta\tau / \max_i \widehat{\Sigma}_{ii}$ for every principal block.

C.3 Proof of Theorem 3.2

Proof. The positive-part map is positively homogeneous, so $(cz)_+ = cz_+$ for $c > 0$. Under the correlation normalization in (10), replacing z by cz replaces ζ by $c\zeta$. On any current face A, the optimizer of (11) consequently changes from u^A to cu^A. Its residual changes from $\alpha^A = \zeta - Cu^A$ to $c\alpha^A$. Positive scaling preserves residual signs and maximizing indices. The fixed index ordering therefore makes every addition and the stopping decision of G_k identical. Hence $G_k(cz_+, \widehat{\Sigma}) = G_k(z_+, \widehat{\Sigma})$.

On the returned face, the positive active subface is also unchanged under scaling. Equation (8) gives

$$v_H(ca) = c\widehat{\Sigma}_{HH}^{-1} a_H = c v_H(a), \qquad w_H(ca) = \frac{c v_H(a)}{\mathbf{1}^\top c v_H(a)} = w_H(a). \tag{47}$$

Combining the locator and valuation identities proves (16). Positive scaling also preserves zero-valuation and empty-face cases, whose uniform fallback is scale independent.

On a positive-valuation cell, $\widehat{\Sigma}_{HH}^{-1} a_H = v_H = b_H w_H$, so (9) gives

$$J_H a_H = (v_H - w_H b_H)/b_H = 0. \tag{48}$$

Since $\nabla_{a_H} \ell_H = -J_H^\top y_H$, the second identity in (17) follows by the chain rule. □

C.4 Proof of Theorem 3.3

Proof. Every F_{tj} contains at most k asset labels. The restricted solve is nonnegative and normalizes its positive solution to unit budget. Its uniform fallbacks also belong to $\mathcal{W}_k$, proving (i). For (ii), (19) gives $|a_{\theta t} - m_t| \le \sigma_t$ coordinatewise, hence

$$\left|(a_{\theta t} - m_t)^\top w\right| \le \sum_i \sigma_{ti} w_i \le \|\sigma_t\|_\infty \sum_i w_i = \|\sigma_t\|_\infty.$$

For (iii), condition on the draws for the current epoch and on the dropout masks. The faces have no dependence on θ. On the stated region, the positive-valuation allocations have Jacobian (9), and constant fallbacks have zero Jacobian. Write $B = |\mathcal{B}|$, $\bar{r}_j = \bar{r}_{\mathcal{B},j}$, and $s_j = s_{\mathcal{B},j}$. For $s_j > \epsilon$,

$$\frac{\partial \mathcal{L}_\mathcal{B}}{\partial r_{tj}} = \frac{1}{J}\left\{-\frac{1}{B s_j} + \frac{\bar{r}_j (r_{tj} - \bar{r}_j)}{(B-1) s_j^3}\right\}. \tag{49}$$

For $s_j < \epsilon$ this derivative is $-1/(JB\epsilon)$. Let J_{tj} be the zero-extended face Jacobian, or zero for a constant fallback. The chain rule gives

$$\frac{\partial r_{tj}}{\partial \theta} = y_{t+1}^\top J_{tj} \operatorname{diag}(\mathbf{1}\{a_{\theta t} > 0\}) \frac{\partial a_{\theta t}}{\partial \theta},$$
$$\nabla_\theta \mathcal{L}_\mathcal{B} = \sum_{t \in \mathcal{B}} \sum_{j=1}^{J} \frac{\partial \mathcal{L}_\mathcal{B}}{\partial r_{tj}} \left(\frac{\partial r_{tj}}{\partial \theta}\right)^\top.$$

These are the backpropagation operations. The proposal law and realized faces have no correction-parameter dependence, so no selection-score term is needed. □

Remark C.2 (Sampling interpretation)**.** For fixed data, parameters, and dropout masks, let L_j denote the batch loss of one sampled sparse path. Independent draws in (20) give

$$\mathbb{E}\left[\frac{1}{J}\sum_{j=1}^{J} L_j\right] = \mathbb{E}[L_1], \qquad \operatorname{Var}\left(\frac{1}{J}\sum_{j=1}^{J} L_j\right) = \frac{\operatorname{Var}(L_1)}{J}.$$

The variance is finite for fixed observed returns because allocations lie in the simplex and the Sharpe ratio denominator is bounded below by $\epsilon > 0$. This is conditional sampling variance, not the variance of SGD over data and parameter updates. A sampled face tuple has a finite, parameter-independent distribution, so differentiating its expected loss reduces to differentiating its finite weighted sum wherever those derivatives exist. The local derivative describes changes in allocation on the evaluated faces. It does not assign counterfactual values to discrete asset swaps. With only one active asset, the allocation is constant and its within-face derivative is zero.

C.5 Parameter-dependent exploration

Section 3.2 fixes the reference distribution and resamples its faces each epoch. If the correction instead moved the sampling distribution, the gradient would include a location-score term. The following specialization of the score-function identity (Schulman et al., 2015) states that distinction.

Proposition C.3 (Gradient under parameter-dependent asset selection)**.** *Let $\eta \sim \mathcal{N}(\mathbf{0}, \Omega)$ with fixed $\Omega \succ 0$, and define $F = G_k((a+\eta)_+, \widehat{\Sigma})$ and $\ell_F(a) = \ell\{y, w(a_+; F, \widehat{\Sigma})\}$. If each fixed-face loss is differentiable at a and the Gaussian region integrals admit dominated differentiation, then*

$$\nabla_a \mathbb{E}_\eta[\ell_F(a)] = \mathbb{E}_\eta\left[\nabla_a \ell_F(a) + \ell_F(a)\Omega^{-1}\eta\right]. \tag{50}$$

Proof. Write $\mathcal{R}_{\mathrm{mov}}(a) = \mathbb{E}_\eta[\ell_F(a)]$, with the random face defined in the proposition. Let $\mathcal{A}_F$ be the measurable region of signals mapped to F by the deterministic locator, including its positive-part operation. The finite face partition gives

$$\mathcal{R}_{\mathrm{mov}}(a) = \sum_F \int_{\mathcal{A}_F} \ell_F(a)\, \varphi_\Omega(z-a)\, dz, \tag{51}$$

where φ_Ω is the Gaussian density. Its derivative is $\nabla_a \varphi_\Omega(z-a) = \varphi_\Omega(z-a)\Omega^{-1}(z-a)$. Differentiate the integral and substitute $z = a + \eta$ to obtain (50), the Gaussian score-function identity (Schulman et al., 2015). □

C.6 Displacement under anchor calibration

Lemma C.4 (Displacement under anchor calibration)**.** *For $P = I - \mathbf{1}\mathbf{1}^\top/N$ and x defined by* (22) *with its zero-denominator convention,*

$$\|x - m\|_2 \le 2\|P(a-m)\|_2 \le 2\|a-m\|_2. \tag{52}$$

In particular, $|a-m| \le \sigma$ implies $\|x-m\|_2 \le 2\|\sigma\|_2$.

Proof. Write $u = Pm$ and $v = Pa$. When $v \neq 0$, moment matching gives $x - m = \|u\|_2 v/\|v\|_2 - u$, including the case $u = 0$. Add and subtract v to obtain

$$\begin{aligned}\|x-m\|_2 &\le |\|u\|_2 - \|v\|_2| + \|v-u\|_2 \\ &\le 2\|v-u\|_2 = 2\|P(a-m)\|_2.\end{aligned} \tag{53}$$

The second inequality is the reverse triangle inequality. When $v = 0$, the fallback gives $x - m = -u$ and $P(a-m) = -u$, so the same bound holds. Finally, the centering matrix P is an orthogonal projection and has operator norm at most one. Thus $\|P(a-m)\|_2 \le \|a-m\|_2$, and the coordinatewise bound $|a-m| \le \sigma$ implies $\|a-m\|_2 \le \|\sigma\|_2$. □

C.7 Proof of Proposition 3.4

Proof. Write $u = c + V^{1/2}h$ with $\|h\|_2 \le \rho$. Then $u^\top w = c^\top w + h^\top V^{1/2} w$. Cauchy–Schwarz bounds the second term below by $-\rho\|V^{1/2}w\|_2$. Since $V \succ 0$ and $w \neq 0$, equality is attained at $h = -\rho V^{1/2} w/\|V^{1/2}w\|_2$. Dividing by the positive portfolio risk proves (24). □

C.8 Proof of Theorem 3.5

Proof. For fixed $w \ge \mathbf{0}$, the denominator of the Sharpe ratio is constant over $\mathcal{U}_+(c_t, b_t)$. Each numerator term $u_i w_i$ is minimized at $u_i = \max\{0, c_{ti} - b_{ti}\} = r_{ti}$. This proves (26).

We next establish the regret bound. For every fixed feasible w and signals x, y,

$$|S(w; x, \widehat{\Sigma}) - S(w; y, \widehat{\Sigma})| \le \|\widehat{\Sigma}^{-1/2}(x-y)\|_2. \tag{54}$$

The result follows from Cauchy–Schwarz in the $\widehat{\Sigma}$ inner product. A pointwise maximum of one-Lipschitz functions remains one-Lipschitz, so the same inequality holds for S_k^*.

Write $w^F = w(r_t; F, \widehat{\Sigma}_t)$, which attains $S_F(r_t, \widehat{\Sigma}_t)$. Add and subtract the objectives at r_t to obtain

$$\begin{aligned}S_k^*(c_t, \widehat{\Sigma}_t) - S(w^F; c_t, \widehat{\Sigma}_t) = {} & S_k^*(c_t, \widehat{\Sigma}_t) - S_k^*(r_t, \widehat{\Sigma}_t) \\ & + S_k^*(r_t, \widehat{\Sigma}_t) - S_F(r_t, \widehat{\Sigma}_t) \\ & + S(w^F; r_t, \widehat{\Sigma}_t) - S(w^F; c_t, \widehat{\Sigma}_t).\end{aligned} \tag{55}$$

Applying (54) to the first and third lines gives the first inequality in (27).

The positive-part operation gives $|c_t - r_t| \le b_t$ coordinatewise. Therefore

$$\|\widehat{\Sigma}_t^{-1/2}(c_t - r_t)\|_2 \le \sqrt{\lambda_{\max}(\widehat{\Sigma}_t^{-1})}\|b_t\|_2. \tag{56}$$

Equation (14) gives $S_k^*(r_t, \widehat{\Sigma}_t) \le \sqrt{U_{r_t}(F)}$, proving the second inequality. □

Remark C.5 (Box interpretation of the deployed signal)**.** For the signal blend in (33), define

$$c_t^C = (\bar{x}_t)_+, \quad c_t^A = (\bar{m}_t)_+, \quad c_t^\lambda = \lambda_t c_t^C + (1-\lambda_t)c_t^A,$$
$$d_t^\lambda = \lambda_t(c_t^C - r_t^C) + (1-\lambda_t)(c_t^A - r_t^A) \geq \mathbf{0}, \qquad c_t^\lambda - d_t^\lambda = r_t^\lambda. \tag{57}$$

Since $0 \leq r_t^C \leq c_t^C$, $0 \leq r_t^A \leq c_t^A$, and $0 \leq \lambda_t \leq 1$, the blended radius is nonnegative and $(c_t^\lambda - d_t^\lambda)_+ = r_t^\lambda$. Thus (26) applies with $(c_t, b_t, r_t) = (c_t^\lambda, d_t^\lambda, r_t^\lambda)$. If $S_{F_t}(r_t^\lambda, \widehat{\Sigma}_t) > 0$, (27) also applies to the deployed allocation with $F = F_t$.

Remark C.6 (Comparison with the prediction anchor)**.** For the deployed portfolio, use $(c_t, b_t, r_t, F) = (c_t^\lambda, d_t^\lambda, r_t^\lambda, F_t)$ from (57), with $S_{F_t}(r_t^\lambda, \widehat{\Sigma}_t) > 0$, and let $h_t = (\bar{m}_t)_+$. Lemma C.4, averaging, and the nonexpansiveness of the positive part give $\|h_t - c_t^\lambda\|_2 \leq 2\lambda_t\|\sigma_t\|_2$. The decomposition above with h_t in place of c_t then yields

$$S_k^*(h_t, \widehat{\Sigma}_t) - S(w_t; h_t, \widehat{\Sigma}_t) \leq \varepsilon_{\text{cert}} + \frac{2\{2\lambda_t\|\sigma_t\|_2 + \|d_t^\lambda\|_2\}}{\sqrt{\lambda_{\min}(\widehat{\Sigma}_t)}}. \tag{58}$$

D Experimental Details

D.1 Data and walk-forward construction

We downloaded adjusted daily closing prices from Yahoo Finance for current index constituents from January 2016 through December 2025. Stocks were retained when price coverage was at least 99.5%. EuroStoxx50 used 90% because its union calendar contains exchange-specific holidays. Prices were forward-filled after this filter, and dates with any remaining missing value were dropped. Table 2 reports the resulting panels. Applying current membership to the full history introduces survivorship bias, so this is a controlled method comparison rather than an investable historical simulation.

Table 2: Equity panels after the coverage filter. Windows count one-step decisions after forming the return input.

market	stocks	price dates	windows	test windows	first date	last date
EuroStoxx50	46	2561	2460	492	2016-01-04	2025-12-31
FTSE100	92	2526	2425	485	2016-01-04	2025-12-31
KOSPI200	159	2447	2346	470	2016-01-04	2025-12-30
Nikkei225	214	2465	2364	473	2016-01-04	2025-12-30
S&P 500	465	2514	2413	483	2016-01-04	2025-12-31

Simple returns are $y_{ti} = P_{ti}/P_{t-1,i} - 1$, and the daily risk-free rate is zero. Each decision uses the preceding $T = 100$ returns. Every method receives

$$\widehat{\Sigma}_t = 0.9\widehat{S}_t + 0.1\frac{\operatorname{tr}(\widehat{S}_t)}{N} I. \tag{59}$$

The shrinkage coefficient is inherited from the DF-STPO protocol and held fixed across methods.

The last 20% of windows in each market remain the test block. Gradient-based training ends 280 windows before that boundary. Because one network is trained on data pooled across markets, checkpoint selection also uses a common date cutoff. Let d_0 be the earliest final-input date of any market's first test decision. We retain early-stopping targets only on calendar dates strictly before d_0. This strict inequality avoids using a later-closing foreign market on the same date. Here, d_0 is January 22, 2024, and every early-stopping block ends on January 19. The same cutoff applies to PFL training, including every frozen anchor, DF-STPO training, and RA-STPO fine-tuning under both input representations.

The numbers of training, early-stopping, calendar-gap, and test windows are $1688/275/5/492$, $1660/272/8/485$, $1596/279/1/470$, $1611/276/4/473$, and $1650/274/6/483$ in the order of Table 2. Dates between the shared checkpoint cutoff and a market's test start supply rolling inputs only and are excluded from training and checkpoint selection. At each test decision, the rolling input contains only returns already observed locally.

D.2 INPUT FEATURES

Every predictor input is constructed from the return window that supplies the covariance. Suppress its ending date and write y_{ui} in chronological order, $u = 1, \dots, T$. Let $y_{\ell i}^{\mathrm{lag}} = y_{T+1-\ell,i}$, where lag one is most recent. We use

$$\mathcal{H} = \big\{\{1\}, \{2, \dots, 5\}, \{6, \dots, 21\}, \{22, \dots, 63\}, \{64, \dots, T\}\big\}, \qquad T \geq 64. \tag{60}$$

The blocks separate the latest day, the remainder of a trading week, the remainder of a trading month, the following two months, and the older part of the window.

Over the full window, define $m_i = T^{-1} \sum_u y_{ui}$, $c_{ui} = y_{ui} - m_i$, $s_i^2 = T^{-1} \sum_u c_{ui}^2$, and $C_{ui} = \prod_{v=1}^{u}(1 + y_{vi})$. Let $s_{21,i}$ be the standard deviation over the latest 21 days. The equal-weight market return is $y_{u,m} = N^{-1} \sum_i y_{ui}$. We also define $\beta_i = \mathrm{Cov}_u(y_{ui}, y_{u,m}) / \mathrm{Var}_u(y_{u,m})$ and $e_{ui} = y_{ui} - \beta_i y_{u,m}$. The remaining notation is

$$\begin{aligned} R_i(B) &= |B|^{-1} \sum_{\ell \in B} y_{\ell i}^{\mathrm{lag}}, \qquad & s_i^- &= \mathrm{sd}_u(\min\{y_{ui}, 0\}), \qquad & p_i^+ &= T^{-1} \sum_u \mathbf{1}\{y_{ui} > 0\}, \\ \gamma_i &= T^{-1} \sum_u c_{ui}^3 / s_i^3, \qquad & \kappa_i &= T^{-1} \sum_u c_{ui}^4 / s_i^4 - 3, \qquad & D_i &= 1 - C_{Ti} / \max_{u \leq T} C_{ui}. \end{aligned} \tag{61}$$

Table 3: Window-derived predictor inputs. Each expression is computed for each asset before cross-sectional normalization.

group	definition
G1: return horizons	$R_i(B)$ for every $B \in \mathcal{H}$
G2: level/persistence	m_i, m_i / s_i, and $\rho_{1,i}$
G3: risk/downside	s_i, $s_{21,i}$, s_i^-, and $s_{21,i} / s_i$
G4: distribution shape	$\max_{T-20 \leq u \leq T} y_{ui}$, $\min_{T-20 \leq u \leq T} y_{ui}$, γ_i, and κ_i
G5: gains/drawdown	p_i^+ and D_i
G6: market exposure	β_i and $\mathrm{corr}_u(y_{ui}, y_{u,m})$
G7: residual return	$\mathrm{mean}_u(e_{ui})$ and $\mathrm{sd}_u(e_{ui})$

The persistence feature is a lag-one product centered using the full-window mean, $\rho_{1,i} = \{(T-1)^{-1} \sum_{u=1}^{T-1} c_{ui} c_{u+1,i}\} / \{\mathrm{sd}(c_{1:T-1,i}) \, \mathrm{sd}(c_{2:T,i}) + 10^{-14}\}$. Other zero-denominator safeguards use 10^{-14} for marginal scales and 10^{-18} for the market variance. All feature standard deviations use the population divisor.

For the window X_t, define the normalized feature matrix $\phi(X_t)$ by

$$(\phi(X_t))_{ij} = \Phi_{\mathrm{std}}^{-1}\left(\frac{\mathrm{rank}_{t,j}(i) - 1/2}{N}\right). \tag{62}$$

Here, Φ_{std} is the standard normal CDF and $\mathrm{rank}_{t,j}(i)$ ranks asset i by feature j across the market at date t. The network notation suppresses this fixed feature transformation. This transformation places features on a common marginal scale across markets. Tied feature values receive distinct ordinal ranks, so ties can make the representation depend on asset order. All learned main-table methods use this representation. Appendix E.5 compares it with raw return windows for every learned method.

D.3 PREDICTORS AND RA-STPO TRAINING

For window-derived inputs, each asset-wise network is trained on data pooled across the five markets. An eight-dimensional market embedding is concatenated to every asset's feature vector. The shared multilayer perceptron has widths 64 and 32, ReLU activations, and dropout 0.10. The raw-input architecture is specified in Appendix E.5. PFL targets cross-sectionally standardized next-day returns. DF-STPO predicts returns directly.

For each seed, PFL predicts standardized-return scores $\nu_t^{(s)} \in \mathbb{R}^N$. Let $\bar{\nu}_t^{(s)} = N^{-1} \sum_i \nu_{ti}^{(s)}$. These scores are mapped to return units around the rolling sample mean $\bar{\mu}_t$ as

$$m_{ti}^{(s)} = \bar{\mu}_{ti} + \mathrm{sd}_i(\bar{\mu}_t) \frac{\nu_{ti}^{(s)} - \bar{\nu}_t^{(s)}}{\mathrm{sd}_i(\nu_t^{(s)}) + 10^{-8}}. \tag{63}$$

The cross-sectional standard deviations in (63) use divisor $N - 1$. The notation $f_{\psi_s}(X_t)$ includes this window-dependent return-unit mapping. The trained predictor is frozen. A newly initialized network with the same architecture produces the bounded correction in (19). It is optimized only with the minibatch Sharpe ratio loss in (21), with numerical floor $\epsilon = 10^{-8}$.

PFL and DF-STPO use Adam with learning rate 10^{-3} and batches of 64. The selected correction uses learning rate 0.001 and batches of 128. All use weight decay 10^{-5}, at most 40 epochs, gradient-norm clipping at five, and patience eight. PFL and DF-STPO checkpoints maximize mean validation rank information coefficient across the five markets. Correction checkpoints maximize mean validation Sharpe ratio of the individual, central, uncalibrated signal allocated by the hard decision map. The complete configuration was chosen using validation performance before its test evaluation.

Greedy face construction batches independent draws on the GPU and updates matrix inverses incrementally while the prefix coefficients remain positive. Otherwise, it uses the nonnegative greedy reference solve. On each sampled face, an active-subface inverse is reused only when primal positivity, stationarity, and inactive-coordinate dual feasibility hold for the current corrected signal. Failed checks trigger a new nonnegative quadratic solve and update the active support.

Network outputs use single precision. Face solves, KKT checks, and portfolio-return reductions use double precision, without an additional covariance regularizer. Active-support and normalization thresholds are 10^{-12}, while KKT residual tolerances scale with floating-point precision and local magnitudes. Constant fallback allocations have zero derivatives. Training inputs and face blocks reside on the GPU, and return reductions use only selected coordinates.

D.4 Deployment implementation

For either ensemble v_t, the coordinatewise variance in (23) is

$$q_i^2(v_t) = \frac{1}{M-1} \sum_{s=1}^{M} \left(v_{ti}^{(s)} - \bar{v}_{ti}\right)^2 \quad (M > 1), \qquad q_i^2(v_t) = 0 \quad (M = 1).$$

The final score selection takes the k largest scores in (32) and solves the nonnegative allocation problem directly on those asset indices. Each robust mean signal is clipped before mixing. Every final portfolio is long-only, unit-sum, and has at most k assets; the face solver can set selected coordinates to zero. If a component holds fewer than k assets, zero-score ties can add coordinates to F_t; even at $\lambda_t = 0$ or 1, re-optimization need not reproduce the original component weights.

Section 3.4 specifies validation initialization and causal updates. We use $\epsilon_s = 10^{-12}$ and $\epsilon_u = 10^{-18}$. Initially $n_t = 0$ and the deployment sums are zero. Calendar-gap returns do not enter the state. The risk update uses uncentered squared returns, and s_*^2 stays fixed. The signal-uncertainty model in (23) does not estimate shared forecast bias or error conditional on asset selection. The credibility weight concerns positive mean risk-normalized advantage under the stated working model.

The correction ablation sets $a^{(s)} = m^{(s)}$ and retains calibration, both uncertainty estimates, credibility initialization and updates, normalized score selection, and the final face solve.

D.5 Baseline settings and reporting

DF-STPO uses 32 seeds, loss mixture $\alpha = 0.5$, soft-top-k sharpness $\beta = 10$, and 32 bisection iterations. It uses the same architecture, inputs, optimizer, split, and covariance estimate, with $\widetilde{\Sigma}_t = \widehat{\Sigma}_t$. One asset ordering is drawn for each training seed. Evaluation averages the forecasts and uses each dataset's fixed ticker-sorted asset order.

Both training solves use double precision and a 10^{-12} risk-norm floor. Prediction error is summed over assets and averaged over examples. DF-STPO is implemented from the published equations, as detailed in Appendix A.3.

The main OSCAR rows also use each dataset's fixed ticker-sorted asset order. All PFL rules, including SD-relaxation, use the same mean of 32 forecasts. Each date's SDP is solved for that mean forecast and covariance using SCS with tolerance 10^{-4} and at most 10^5 iterations. These numerical solutions are not exact optimality certificates. On 9 S&P 500 dates, prolonged or nonfinite default solves required SCS without acceleration, with the same program, tolerance, and iteration limit. mSSRM-PGA and ASMP-AFBA use the released constants and stopping rules in Algorithms 5 and 6.

Our mSSRM implementation retains exactly k indices at positive threshold ties and uses a singleton fallback if its iterate is zero. These safeguards enforce cardinality and unit budget without changing a nondegenerate native solution. The two-stage ASMP implementation uses the released second-stage size $\max\{4, k\}$, which equals k for the $k = 10$ experiments here.

In Table 1, PFL and DF-STPO allocate from 32-member mean forecasts. RA-STPO uses 16 anchor–corrector pairs, two component allocations, and a final face solve. Network counts match, while training objectives and deployment procedures differ. Appendix E.7 reports conditional member-resampling diagnostics.

Portfolios rebalance daily. The realized portfolio return is $r_{p,t} = y_{t+1}^\top w_t$, and we report the annualized Sharpe ratio (SR) and terminal wealth W:

$$\mathrm{SR} = \sqrt{252}\,\frac{\mathrm{mean}_t(r_{p,t})}{\mathrm{sd}_t(r_{p,t})}, \qquad W = \prod_t (1 + r_{p,t}). \tag{64}$$

All markets use the same rule, with no transaction costs or turnover constraints. The metrics are computed from each method's chronological portfolio-return path in each market.

E Additional Experiments

We examine asset-order dependence, the common long-only constraint, sparse allocation, and calibration. We also compare input representations and correction use, then assess sensitivity to M, J, conditional variability across members, and computational cost.

E.1 Transformed-coordinate selection is order dependent

Table 4 reports historic OSCAR at $k = 10$ under 16 random asset orderings, holding returns, moments, and optimization settings fixed. Asset-coordinate formulas commute with relabeling away from selection ties, provided any random draws are relabeled jointly. The Cholesky score has an additional dependence on the ordered factorization even without ties.

Table 4: OSCAR under asset relabeling. SR and W are reported for the ticker-sorted order and averaged over 16 random orderings. The last three columns summarize SR across those orderings.

	Original		Mean		SR across orderings		
Market	SR	W	SR	W	Min	Max	SD
EuroStoxx50	1.874	1.794	1.815	1.764	1.625	2.199	0.157
FTSE100	1.239	1.491	1.561	1.652	0.980	1.994	0.261
KOSPI200	0.116	0.961	0.103	0.982	-0.835	0.852	0.390
Nikkei225	0.261	1.062	0.596	1.336	-0.095	1.400	0.365
S&P 500	2.444	3.315	1.462	1.983	0.908	1.927	0.322

On the S&P 500, OSCAR's Sharpe ratio under alphabetical asset ordering is 2.444, outside the range observed across the sampled orderings $[0.908, 1.927]$, whose mean is 1.462. Relabeling alone therefore changes the selected strategy. This illustrates the coordinate distinction in Proposition 2.1 and makes the factorization order a material input to the selector.

E.2 The effect of the common long-only constraint

OSCAR, SD-relaxation, and DF-STPO permit unrestricted signs in their source formulations. We hold each selected face fixed and compare the common solve in (34) with unit-gross normalization of

an unrestricted-sign direction (Table 5). OSCAR and SD-relaxation use $v_F = \widehat{\Sigma}_{FF}^{-1}\mu_F$; DF-STPO retains its auxiliary-budget constraint when solving on the same selected face. mSSRM-PGA and ASMP-AFBA are omitted because they are already long-only. OSCAR and DF-STPO use the fixed ticker-sorted asset order and average their 32 learned forecasts in single precision before double-precision allocation. SD-relaxation uses the same double-precision mean of 32 forecasts as in Table 1. Each long-only/long–short pair uses an identical signal and selected face.

Table 5: Long-only unit-budget and long–short unit-gross evaluation. Each market has separate annualized SR and terminal wealth (W) columns. Native long-only methods are omitted.

	EuroStoxx50		FTSE100		KOSPI200		Nikkei225		S&P 500	
Signal / rule	SR	W	SR	W	SR	W	SR	W	SR	W
common long-only face solve										
Historic / OSCAR	1.874	1.794	1.239	1.491	0.116	0.961	0.261	1.062	2.444	3.315
Historic / SD-relaxation	2.015	1.847	1.124	1.407	0.989	1.651	0.881	1.465	1.267	1.544
PFL / OSCAR	2.707	2.328	2.361	2.157	1.776	2.469	1.740	2.786	1.807	2.207
PFL / SD-relaxation	2.771	2.357	2.533	2.201	1.911	2.483	1.124	1.894	1.965	2.209
DF-STPO	2.128	1.910	1.634	1.637	2.558	3.874	1.182	1.821	1.628	2.005
unit-gross long–short direction										
Historic / OSCAR	0.882	1.163	0.495	1.088	-0.364	0.850	0.283	1.057	2.275	1.968
Historic / SD-relaxation	0.955	1.158	0.299	1.042	0.493	1.125	0.226	1.036	1.282	1.348
PFL / OSCAR	2.096	1.363	1.741	1.355	2.177	1.907	1.087	1.314	1.075	1.334
PFL / SD-relaxation	1.481	1.213	2.324	1.395	2.436	1.795	0.282	1.051	1.462	1.339
DF-STPO	0.900	1.144	1.705	1.276	1.493	1.467	1.288	1.271	2.340	1.674

Long-only re-optimization changes both the permitted signs and exposure normalization. The long–short direction is normalized to unit gross exposure, with remaining net budget held as zero-return cash. This diagnostic does not isolate the causal effect of prohibiting short sales. The main comparison consistently uses the long-only mandate.

E.3 Residual-alpha selection and exact allocation

We audit 24 test dates per market for each of the two robust mean-signal components, covering 240 signals. Table 6 compares the batched and reference nonnegative greedy implementations, compares single- and double-precision selection, and reports the selected value relative to its residual upper bound. Minimum market-level solver agreement is 100.0%, and precision agreement is 100.0%.

Table 6: Residual-alpha selection on the deployed component signals. The last column is the median selected value divided by its residual-based upper bound.

market	solver agreement	precision agreement	fallback rate	median Φ_g/U_g
EuroStoxx50	100.0%	100.0%	0.00%	1.000
FTSE100	100.0%	100.0%	0.00%	1.000
KOSPI200	100.0%	100.0%	0.00%	0.970
Nikkei225	100.0%	100.0%	0.00%	0.999
S&P 500	100.0%	100.0%	0.00%	0.153

Table 7 reports allocation accuracy on final score-selected faces. We reconstruct the quadratic solution as $v_t = ((r_t^\lambda)^\top w_t)/(w_t^\top \widehat{\Sigma}_t w_t)\, w_t$ and set $g_{t,F_t} = (\widehat{\Sigma}_t v_t - r_t^\lambda)_{F_t}$. The face KKT residual is the maximum of $\|(g_{t,F_t})_-\|_\infty$ and $\|v_{t,F_t} \odot g_{t,F_t}\|_\infty$ over all test dates. The value ratio uses the same 24-date subset. These diagnostics separate allocation accuracy from the quality of the selected set. A low value ratio can reflect a loose upper bound.

Table 7: Final-face optimality and empirical objective bounds.

market	maximum face KKT residual	median $\Phi(F_t)/U(F_t)$
EuroStoxx50	1.09e-16	1.000
FTSE100	1.45e-16	0.988
KOSPI200	6.27e-17	0.988
Nikkei225	2.17e-16	0.990
S&P 500	1.72e-16	0.099

E.4 Checking the calibration bound

Table 8 checks all assets, 16 members, and test dates. The pre-calibration ratio $|a_i - m_i|/\sigma_i$ obeys its unit bound up to stored single-precision rounding. Calibration can exceed this coordinatewise bound. No audited case violates Lemma C.4. The maximum ratio $\|x - m\|_2/(2\|\sigma\|_2)$ is 0.481. These quantities measure displacement from the anchor, rather than forecast-error coverage.

Table 8: Calibration displacement. Pre and post ratios are $|a_i - m_i|/\sigma_i$ and $|x_i - m_i|/\sigma_i$. The last column is the maximum of $\|x - m\|_2/(2\|\sigma\|_2)$ over members and dates.

market	pre max	post 95%	post max	post > 1	norm-bound ratio
EuroStoxx50	1.000	1.329	2.707	16.5%	0.481
FTSE100	1.000	1.401	2.758	13.8%	0.441
KOSPI200	1.000	1.248	2.826	13.3%	0.475
Nikkei225	1.000	1.334	2.896	11.2%	0.443
S&P 500	1.000	1.393	2.942	16.2%	0.456

E.5 Window-derived features versus raw return inputs

Table 9 compares inputs for every learned method using the same returns, splits, covariances, and allocation rules. The raw-input model uses cross-sectionally standardized returns at each of the 100 lags, a 512–256 network, and zero dropout. Window-derived inputs use the architecture in Appendix D.3. For both representations, the market embedding dimension follows the rule $d_{\mathrm{emb}} = \max\{4, \lceil d_{\mathrm{in}}/2.75 \rceil\}$, where d_{in} is the number of input features per asset. This gives 37 dimensions for the 100 raw return lags and 8 for the 22 window-derived features. The comparison therefore includes architecture differences. Networks are trained separately for each representation, with $J = 8$ for RA-STPO.

Table 9: Window-derived features versus raw returns for every learned method. Annualized SR and terminal wealth (W) are shown in separate columns. Bold marks the better input in each matched method–market–metric comparison. Forecast aggregation and allocation match Table 1.

	EuroStoxx50		FTSE100		KOSPI200		Nikkei225		S&P 500	
Training / decision rule	SR	W	SR	W	SR	W	SR	W	SR	W
window-derived inputs										
PFL / OSCAR	**2.707**	**2.328**	**2.361**	**2.157**	**1.772**	**2.463**	**1.740**	**2.786**	1.807	2.207
PFL / SD-relaxation	**2.771**	**2.357**	**2.533**	**2.201**	**1.911**	**2.483**	**1.124**	**1.894**	1.965	2.209
PFL / mSSRM-PGA	**2.601**	**2.258**	**2.204**	**2.004**	**1.386**	**2.022**	**1.303**	**2.024**	1.769	2.237
PFL / ASMP-AFBA	**2.457**	**2.167**	**2.300**	**2.022**	**2.232**	**2.318**	0.795	1.341	1.717	1.685
DF-STPO	**2.128**	1.910	**1.634**	**1.637**	**2.558**	**3.874**	**1.182**	**1.821**	**1.628**	2.005
RA-STPO	**2.837**	**2.512**	**2.402**	**2.013**	**2.611**	**3.498**	**1.543**	**2.322**	2.392	**2.233**
raw return lags										
PFL / OSCAR	2.244	2.046	1.111	1.419	0.429	1.170	0.263	1.060	**2.177**	**2.770**
PFL / SD-relaxation	2.008	1.884	1.262	1.461	-0.023	0.906	0.535	1.253	**2.241**	**2.749**
PFL / mSSRM-PGA	1.979	1.949	1.738	1.721	0.592	1.299	1.229	1.905	**1.831**	**2.441**
PFL / ASMP-AFBA	2.024	1.911	1.872	1.759	1.502	1.788	**0.963**	**1.447**	**2.301**	**2.168**
DF-STPO	1.938	**1.998**	1.316	1.544	1.507	2.331	0.463	1.208	1.392	**2.108**
RA-STPO	2.142	1.998	1.939	1.703	0.518	1.217	1.041	1.609	**2.490**	2.174

E.6 THE CONTRIBUTION AND USE OF DECISION-FOCUSED CORRECTION

Table 10 compares three uses of the same 16 anchor–corrector pairs. Complete RA-STPO uses the full deployment rule. Without correction, $a^{(s)} = m^{(s)}$, retaining calibration, uncertainty adjustment, credibility updates, score selection, and final allocation. Corrected mean only allocates directly from r^C, omitting the comparison with the anchor and the final score-based face selection.

Table 10: Decision-focused correction and its use. Each market has separate annualized SR and terminal wealth (W) columns. All rows use the same 16 anchors.

	EuroStoxx50		FTSE100		KOSPI200		Nikkei225		S&P 500	
Method	SR	W	SR	W	SR	W	SR	W	SR	W
Complete	2.837	2.512	2.402	2.013	2.611	3.498	1.543	2.322	2.392	2.233
w.o. correction	2.651	2.220	2.583	2.097	1.923	2.319	1.153	1.790	1.921	1.756
Corrected mean only	2.869	2.569	1.664	1.659	2.670	3.622	1.592	2.381	2.505	2.360

With deployment held fixed, correction raises mean SR from 2.046 to 2.357 and mean wealth from 2.037 to 2.515. The corrected-mean-only control gives mean SR 2.260 and wealth 2.518. The complete policy improves mean SR over this control, with similar mean wealth.

E.7 CONDITIONAL VARIABILITY ACROSS ENSEMBLE MEMBERS

We assess sensitivity to member resampling with the trained model pool and market history held fixed. For each learned method and market, we draw 100 bootstrap samples of member indices with replacement (Efron, 1979). PFL and DF-STPO resample their 32 forecasts and re-optimize each resampled mean. RA-STPO resamples its 16 anchor–corrector pairs together and recomputes calibration, mean uncertainty, component allocations, the validation prior, and the causal final portfolio. PFL rules share the resampling indices. Historic rules have no trained member pool.

For SD-relaxation, each bootstrap replicate solves the same SDP at its new mean forecasts, with the original tolerance and iteration limit. A warm start reuses the numerical solver state across objectives, while every changed objective is optimized again. Table 11 reports the resulting standard deviations of SR and terminal wealth.

Table 11: Conditional member-bootstrap standard deviations over 100 replicates, with separate SR and wealth (W) columns. Each replicate re-evaluates the complete portfolio path. Dashes denote computations terminated due to excessive runtime.

	EuroStoxx50		FTSE100		KOSPI200		Nikkei225		S&P 500	
Method	SR	W	SR	W	SR	W	SR	W	SR	W
PFL / OSCAR	0.079	0.055	0.108	0.075	0.116	0.174	0.122	0.228	0.128	0.157
PFL / SD-relaxation	0.094	0.068	0.098	0.063	0.128	0.167	0.120	0.155	—	—
PFL / mSSRM-PGA	0.035	0.027	0.041	0.028	0.069	0.080	0.049	0.064	0.068	0.080
PFL / ASMP-AFBA	0.032	0.024	0.029	0.019	0.076	0.067	0.039	0.023	0.075	0.045
DF-STPO	0.199	0.110	0.208	0.112	0.209	0.426	0.224	0.232	0.204	0.184
RA-STPO	0.072	0.072	0.057	0.032	0.112	0.215	0.113	0.165	0.111	0.108

E.8 TIME AND MEMORY

We use one worker per NVIDIA A10 GPU and two CPU threads per training worker. RA-STPO has 16 anchors and 16 correctors, matching the 32 networks of each learned baseline. PFL rows share their forecasts, and RA-STPO reuses the first 16 anchors. Table 12 reports training and backtest costs. Total RA-STPO training takes $1,486.8$ seconds, including 274.3 seconds for anchors, compared with $2,037.5$ seconds for DF-STPO (27.0% less). Times sum measured stages and exclude queue delay and serialization. Measurements are shared only for identical training stages. The sensitivity totals in Table 15 also include the anchor predictions required before correction training.

Table 12: Main-table computational costs. Memory is peak allocated GPU memory and process-lifetime CPU RSS across stages. SD-relaxation backtests exclude the separately timed SDP solves.

Main-table row	networks	training (s)	forecast (s)	backtest (s)	GPU/CPU MiB
Historic / OSCAR	0	0.0	0.0	3.3	100.0/1888
Historic / SD-relaxation	0	0.0	0.0	8.9	69.2/1876
Historic / mSSRM-PGA	0	0.0	0.0	20.1	106.4/1809
Historic / ASMP-AFBA	0	0.0	0.0	94.5	161.3/1864
PFL / OSCAR	32	500.0	3.1	6.1	320.5/4278
PFL / SD-relaxation	32	500.0	3.1	8.7	320.5/4278
PFL / mSSRM-PGA	32	500.0	3.1	21.0	320.5/4278
PFL / ASMP-AFBA	32	500.0	3.1	101.7	320.5/4278
DF-STPO	32	2037.5	2.7	10.6	699.8/4770
RA-STPO (Ours)	32	1486.8	3.3	15.8	1082.1/4582

For SD-relaxation, the backtest column measures allocation from precomputed selection scores. The separate SDP computation for the PFL mean forecasts records $199,623.9$ CPU seconds, including interrupted and recovery solves.

Table 13 reports allocation latency on the first test window. Baselines start from a resident historical signal or 32-member mean forecast and covariance. RA-STPO starts from the anchor and corrected signals of 16 members, the covariance, and the validation-initialized state, all already in memory. Its timing includes calibration, uncertainty, two component allocations, credibility, score selection, and the final solve. Network inference, file I/O, and one-time prior initialization are excluded. GPU entries are medians of ten repetitions after three warm-ups. Single-thread CVXPY/SCS timings on AMD EPYC 7543 are reused for SD-relaxation.

Table 13: Resident-input latency. GPU memory is incremental working memory. CPU memory is process-lifetime peak RSS. CPU timings for SD-relaxation include the SDP solves.

method	EuroStoxx50 (ms)	S&P 500 (ms)	peak MiB	device
Historic / OSCAR	11.28	13.54	6.6	GPU
Historic / SD-relaxation	637.39	216088.20	1273.9	CPU
Historic / mSSRM-PGA	8.32	81.88	11.5	GPU
Historic / ASMP-AFBA	1774.34	1305.84	13.5	GPU
PFL / OSCAR	14.40	11.83	6.6	GPU
PFL / SD-relaxation	501.87	287104.11	1328.4	CPU
PFL / mSSRM-PGA	8.21	76.91	11.5	GPU
PFL / ASMP-AFBA	1238.74	1373.03	13.5	GPU
DF-STPO	8.44	10.47	5.0	GPU
RA-STPO	36.90	27.13	0.1	GPU

E.9 Sensitivity to ensemble size and face count

Table 14 reports results for all 25 combinations of $M \in \{1, 4, 8, 16, 32\}$ and $J \in \{1, 2, 4, 8, 16\}$. For each J, we train 32 correctors from scratch and evaluate their first M members, sharing the same seed-specific anchors across face counts. The learning rate, batch size, proposal spread, and deployment match the main setting. Every setting refreshes all faces each epoch and averages path losses. In particular, $J = 1$ uses a fresh random face, rather than the unperturbed anchor face. When $M = 1$, the disagreement term is zero and the window uncertainty remains. The main setting is $M = 16, J = 8$.

Table 14: All 25 ensemble-size and face-count combinations. Each market has separate SR and terminal wealth (W) columns. Bold indices mark the selected main setting. Member prefixes and histories are shared.

		EuroStoxx50		FTSE100		KOSPI200		Nikkei225		S&P 500	
J	M	SR	W	SR	W	SR	W	SR	W	SR	W
1	1	2.680	2.453	2.120	1.922	2.568	3.505	1.006	1.650	1.989	1.958
1	4	2.769	2.448	2.419	2.019	2.558	3.396	1.256	1.886	2.148	2.023
1	8	2.751	2.423	2.384	2.011	2.571	3.376	1.294	1.961	2.344	2.248
1	16	2.802	2.439	2.493	2.053	2.574	3.353	1.400	2.085	2.414	2.259
1	32	2.742	2.430	2.466	2.041	2.423	3.188	1.479	2.202	2.429	2.243
2	1	2.557	2.391	2.030	1.891	2.676	3.739	1.137	1.791	2.055	2.021
2	4	2.744	2.525	2.259	1.946	2.621	3.647	1.149	1.784	2.077	1.976
2	8	2.653	2.389	2.286	1.962	2.559	3.468	1.187	1.855	2.090	1.995
2	16	2.771	2.454	2.415	2.017	2.470	3.272	1.394	2.106	2.378	2.231
2	32	2.724	2.412	2.444	2.036	2.469	3.311	1.559	2.323	2.471	2.250
4	1	2.508	2.265	2.068	1.899	2.543	3.520	0.925	1.571	2.028	1.912
4	4	2.678	2.402	2.331	1.977	2.631	3.642	1.191	1.826	2.073	1.907
4	8	2.655	2.382	2.310	1.978	2.562	3.455	1.289	1.960	2.108	1.998
4	16	2.761	2.438	2.455	2.039	2.538	3.358	1.496	2.227	2.338	2.169
4	32	2.707	2.407	2.420	2.023	2.521	3.382	1.538	2.272	2.484	2.259
8	1	2.656	2.498	2.113	1.932	2.678	3.764	0.921	1.568	2.027	1.946
8	4	2.721	2.479	2.268	1.945	2.813	4.049	1.266	1.968	1.917	1.885
8	8	2.708	2.422	2.257	1.945	2.687	3.709	1.226	1.917	2.107	2.054
8	**16**	2.837	2.512	2.402	2.013	2.611	3.498	1.543	2.322	2.392	2.233
8	32	2.736	2.423	2.405	2.016	2.635	3.566	1.518	2.249	2.316	2.143
16	1	2.556	2.329	2.135	1.943	2.596	3.555	1.035	1.676	1.887	1.864
16	4	2.662	2.369	2.344	1.983	2.620	3.550	1.305	1.957	2.180	2.048
16	8	2.695	2.389	2.313	1.979	2.569	3.411	1.369	2.072	2.046	1.953
16	16	2.807	2.464	2.453	2.035	2.488	3.226	1.569	2.331	2.281	2.128
16	32	2.740	2.413	2.430	2.026	2.582	3.427	1.621	2.398	2.434	2.211

Table 15 sums anchor training, the anchor forecast pass, and correction training for each configuration. Early stopping determines the number of epochs separately for each run, so total training time need not increase monotonically with J.

Table 15: Summed training time in seconds, including anchor training, its forecast pass for correction training, and correction training. Deployment and correction export are excluded. Bold marks the main setting. Each entry is the total for its member prefix.

M	$J = 1$	$J = 2$	$J = 4$	$J = 8$	$J = 16$
1	104.4	106.1	114.2	99.0	95.3
4	273.2	320.5	296.5	377.3	314.2
8	523.0	656.5	612.5	715.0	700.8
16	1102.4	1273.2	1278.4	**1488.4**	1403.9
32	2254.3	2599.0	2652.6	3051.9	3089.1